



**Universidade Técnica de Lisboa**  
**INSTITUTO SUPERIOR TÉCNICO**

**Optimização em sistemas de processos químicos:  
generalização de modelos com planeamento e sequenciamento**

**João Luís de Miranda**

Mestre em *Engenharia Química* (IST)

Tese para obtenção do grau de Doutor em Engenharia Química

Orientador: Miguel António Soares Casquilho

**Júri**

Presidente: Reitor da Universidade Técnica de Lisboa

Vogais: Doutor João Carlos Moura Bordado

Doutor Eugénio Manuel de Faria Campos Ferreira

Doutor Rui Manuel Moura de Carvalho Oliveira

Doutor Augusto Orlando Lopes Queiroz Novais

Doutor Miguel António Soares Casquilho

**Outubro de 2007**



Apresenta-se uma proposta de técnicas e algoritmos, dirigida a casos de optimização no âmbito da Engenharia Química: planeamento de processos (*process planning*); e dimensionamento e sequenciamento de processos descontínuos (*design and scheduling of batch processes*).

A investigação aplica técnicas de optimização por Programação Matemática, especificamente de Programação Linear Mista (MILP), que conjuga a integralidade de variáveis com a flexibilidade de formulação proporcionada pelas variáveis binárias. Propõe-se a generalização de modelos existentes para os referidos casos de optimização (planeamento, dimensionamento e sequenciamento), por incorporação de vertentes estocásticas e de robustez, na solução e no modelo.

Sistematiza-se uma abordagem que, decorrendo da complexidade computacional, conduz a procedimentos heurísticos para a resolução dos modelos matemáticos tratados. Assim, constroem-se e aplicam-se estes procedimentos, elaborados segundo o problema modelizado, e efectua-se a correspondente análise probabilística. Obtiveram-se frequentemente as soluções óptimas e boas aproximações para as soluções sub-óptimas.

Aborda-se a optimalidade de cada problema equacionado, avaliando a sensibilidade à incerteza e promovendo a robustez, em atitude crítica perante o risco. Neste contexto, em detrimento da obtenção dum eventual óptimo matemático, valoriza-se a determinação de gamas de variação para os principais estimadores ou parâmetros dos casos de optimização problematizados.

## Palavras-chave

Optimização de processos químicos; Programação Linear Mista; generalizações; métodos de resolução; heurísticas.



This thesis deals with the development of tools aimed at known optimization cases in Chemical Engineering: process planning; and design and scheduling of batch processes.

The investigation applies optimization techniques of Mathematical Programming, specifically of Mixed Integer Linear Programming (MILP), that conjugates the integrality of variables and the flexibility of formulation with binary variables. It proposes the generalization of known mathematical models for the cited optimization cases (planning; design and scheduling), incorporating stochastic and robustness features, both in the solution and in the model.

Considering the computational complexity of the mathematical models, it systematizes a methodology that leads to heuristic procedures for the corresponding resolution. It thus develops and applies these procedures, worked according to each model analysed, and it performs the probability analysis associated. The heuristic procedures reach frequently the optimal solutions, and obtain good approximations for the suboptimal ones.

The optimality of each problem analysed is studied, assessing the sensitivity to uncertainty and promoting the robustness, and a risk treatment is performed. In this context, preference is given to defining variation ranges for the main estimators or parameters, instead of just obtaining the mathematical optimum for the optimization cases under scrutiny.

## Keywords

Chemical process optimization; Mixed Integer Linear Programming; generalizations; resolution methods; heuristics.



---

---

## Agradecimentos

---

Ao Prof. Miguel Casquilho, pelo incentivo e suporte, sempre presente ao longo desta investigação.

Às Prof.<sup>as</sup> Palmira Moutinho e Elisete Silva e ao Dr. Luís Miranda.

À Dr.<sup>a</sup> Helena Godinho e ao Dr. José Barrias.

Aos Colegas Cristina Dias e Luís Sertório.

Aos Colegas Isabel Borges, José Carvalho e Paulo Brito.

Ao Colega Luís Baptista.

Ao Nikolaos Sahinidis (Univ. Illinois, Urbana-Champaign, Illinois, EUA).

Ao Karl Frauendorfer (Univ. St. Gallen, St. Gallen, Suíça).

Ao *Conselho Directivo* da ESTG, Escola Superior de Tecnologia e Gestão, Instituto Politécnico de Portalegre.

Ao *Conselho Científico* do IST.

Ao *Centro de Informática* da ESTG.

Ao *Centro de Informática* do IST.

Aos funcionários da Biblioteca da ESTG.

Aos funcionários da Biblioteca de Química, da Biblioteca Central e da Biblioteca de Civil, do IST.

Ao *Deep Purple Overseas Music*.

Ao *Jack Sparrow*.

Ao Luís, João e Virgínia pelo seu permanente e incondicional encorajamento ao longo deste empreendimento.

**J. L. Miranda**

*Lisboa  
Setembro de 2007*

---

### *Aspecto gráfico*

O presente texto, designadamente em várias das convenções adoptadas para apresentação de capítulos, figuras, tabelas, etc., aproxima-se graficamente do livro de E. B. NAUMAN, “Chemical Reactor Design”, ed. John Wiley & Sons, New York, NY, 1987.

---



<b>Resumo</b>	<i>i</i>
<b>Palavras-chave</b>	<i>i</i>
<i>Summary</i>	<i>iii</i>
<i>Keywords</i>	<i>iii</i>
<b>Agradecimentos</b>	<i>v</i>
<b>Glossário de designações</b>	<i>xi</i>

---

## **Capítulo 1 Antecedentes e motivação** 1

---

<b>1.1</b> Optimização e Engenharia Química	2
1.1.1 Modelação matemática do problema,	4
1.1.2 Desenvolvimento de procedimentos heurísticos,	5
1.1.3 Estudo da optimalidade,	6
<b>1.2</b> Generalização de modelos	7
<b>1.3</b> Objectivo e Organização da <i>Tese</i>	9

---

## **Capítulo 2 Casos de Optimização por Programação Linear Mista em sistemas de processos químicos** 13

---

<b>2.1</b> O planeamento de processos ( <i>process planning</i> )	16
2.1.1 Aspectos do planeamento de processos,	16
2.1.2 Problemas de planeamento de processos dedicados,	23
2.1.3 O planeamento de processos flexíveis,	30
2.1.4 Afinação das técnicas de resolução e investigação analítica,	34
2.1.5 Incerteza no planeamento de processos,	38
<b>2.2</b> O dimensionamento e sequenciamento de processos descontínuos ( <i>design and scheduling of batch processes</i> )	43
2.2.1 Aspectos do dimensionamento e sequenciamento de processos descontínuos,	47
2.2.2 Dimensionamento e sequenciamento de sistemas descontínuos: campanhas de produto único,	50
2.2.3 Dimensionamento e sequenciamento de sistemas descontínuos: campanhas de produtos múltiplos,	54
2.2.4 Outros enquadramentos,	63
Sinopse	65

---

## **Capítulo 3 Implementação computacional** 67

---

<b>3.1</b> Aplicação a modelos de planeamento de processos (Fortran)	69
3.1.1 Modelo determinístico,	70
3.1.2 Modelo estocástico com recurso,	74
3.1.3 Tratamento dos problemas para decomposição,	78

3.2	Aplicação modular ao dimensionamento e sequenciamento de processos descontínuos (ambiente GAMS)	82
3.2.1	Estrutura modular da implementação,	83
3.2.2	Análise probabilística de heurísticas,	85
3.2.3	Generalização estocástica,	87
	Sinopse	92
<b>Capítulo 4 Análise dos modelos e heurísticas</b>		<b>95</b>
4.1	Abordagem teórica ao planeamento de processos	98
4.1.1	O problema com processos dedicados,	98
4.1.2	O problema com processos flexíveis,	101
4.1.3	O problema com esquemas de produção flexível,	103
4.1.4	O problema com reformulação de restrições,	105
4.1.5	Os problemas estocástico e robusto,	106
4.2	Análise probabilística de heurísticas para o dimensionamento e sequenciamento de processos descontínuos	109
4.2.1	O modelo com processo único e em campanha de produto único,	110
4.2.2	O modelo com processos múltiplos e em campanha de produto único,	114
4.2.3	O modelo com processo único e em campanha de produtos múltiplos,	119
4.2.4	O modelo com processos múltiplos e em campanha de produtos múltiplos,	122
	Sinopse	124
<b>Capítulo 5 Estudo numérico da optimalidade</b>		<b>127</b>
5.1	Abordagem generalizada ao planeamento de processos: flexibilidade e robustez	130
5.1.1	Aplicação numérica: sensibilidade a penalizações ( <i>spplan_ex1</i> ),	133
5.1.2	Aplicação numérica: reagente crítico ( <i>spplan_ex2</i> ),	141
5.1.3	Aplicação numérica: produto crítico ( <i>spplan_ex3</i> ),	144
5.2	Abordagem generalizada ao dimensionamento e sequenciamento de processos descontínuos: robustez e períodos de tempo	148
5.2.1	Aplicação numérica: robustez e período único ( <i>spbatch_ex1</i> ),	152
5.2.2	Aplicação numérica: robustez e períodos múltiplos ( <i>spbatch_ex2</i> ),	161
	Sinopse	173
<b>Capítulo 6 Conclusões e projecções</b>		<b>177</b>
6.1	Trabalho desenvolvido	178
6.1.1	Planeamento de processos,	178
6.1.2	Dimensionamento e sequenciamento de processos descontínuos,	179
6.2	Desenvolvimentos recentes e afins	181
6.3	Perspectivas e desenvolvimentos futuros	182

<b>Bibliografia</b>	185
<b>Apêndices</b>	
<b>Apêndice 3</b> <sup>1</sup> Ilustração de aplicações de suporte	195
<b>Apêndice 4-A</b> Heurísticas: desenvolvimento, pseudocódigo	207
<b>Apêndice 4-B</b> Ilustração dos exemplos numéricos para análise probabilística de heurísticas	233
<b>Apêndice 5-A</b> Aspectos do modelo estocástico de planeamento de processos	237
<b>Apêndice 5-B</b> Dados dos exemplos para o modelo estocástico de planeamento de processos	241
<b>Apêndice 5-C</b> Decomposição de Benders para o modelo estocástico de planeamento de processos	247
<b>Apêndice 5-D</b> Aspectos da modelação para dimensionamento e sequenciamento de processos descontínuos	251
<b>Apêndice 5-E</b> Dados dos exemplos para o dimensionamento e sequenciamento de processos descontínuos	257
<b>Anexos</b>	
<b>Anexo 1</b> Introdução ao modelo ELS	261
<b>Anexo 4-A</b> Complexidade Computacional	265
<b>Anexo 4-B</b> Heurísticas	269

---

<sup>1</sup> A designação de Apêndices e Anexos associa-se numericamente ao Capítulo a que se refere, complementando-se alfabeticamente se necessário.



---

---

## Glossário de designações

Neste glossário listam-se designações utilizadas (siglas, termos), referindo-se também expressões técnicas cujo uso principal se situa noutras áreas de investigação, sendo por isso pouco conhecidas em Engenharia Química.

- 2SSP – Programação Estocástica em Duas Fases (*Two-Stage Stochastic Programming*)
- almost surely (a.s.)* – “Quase seguramente”, de carácter assintótico
- batch\_milp\_fmz* – Modelo MILP para dimensionamento e sequenciamento de processos descontínuos (*batch*), com as características *Flowshop/ Multiple Machine/ MPC/ ZW*
- batch\_milp\_fmsz* – Modelo MILP para dimensionamento e sequenciamento de processos descontínuos (*batch*), com as características *Flowshop/ Multiple machine/ SPC/ ZW*
- batch\_milp\_fsmz* – Modelo MILP para dimensionamento e sequenciamento de processos descontínuos (*batch*), com as características *Flowshop/ Single Machine/ MPC/ ZW*
- batch\_milp\_fssi* – Modelo MILP para dimensionamento e sequenciamento de processos descontínuos (*batch*), com as características *Flowshop/ Single machine/ SPC/ Intermediate storage*
- batch\_milp\_fssz* – Modelo MILP para dimensionamento e sequenciamento de processos descontínuos (*batch*), com as características *Flowshop/ Single machine/ SPC/ ZW*
- batch\_nlp\_fsmu* – Modelo NLP para dimensionamento e sequenciamento de processos descontínuos (*batch*), com as características *Flowshop/ Single machine/ MPC/ UIS*
- batch\_nlp\_fsmz* – Modelo NLP para dimensionamento e sequenciamento de processos descontínuos (*batch*), com as características *Flowshop/ Single Machine/ MPC/ ZW*
- complete recourse* – Recurso completo (considera todo o espaço de procura de segunda fase em 2SSP)
- CPM – Método do caminho crítico (*Critical Path Method*)

- CT – Tempo de ciclo (*cycle time*)
- cyclic scheduling* – Sequência do ciclo de produções
- cycsched\_ip\_tsp* – Modelo de IP para sequenciamento do ciclo de produções (*cyclic scheduling*), formulado como um problema de TSP
- design and scheduling of batch processes* – Dimensionamento e sequenciamento de processos descontínuos
- ELS – Dimensionamento óptimo de lotes (*uncapacitated Economic Lot Size*),
- feasibility* – Admissibilidade
- FIS – Armazenagem finita (*finite intermediate storage*)
- flowshop* – Ambiente *multiproduct* para sequenciamento de produção, no qual os produtos requerem similares estágios de processamento e sequência de operações
- GAMS – *General Algebraic Modeling System*
- GO – Optimização global (*Global Optimization*)
- infeasibility* – Inadmissibilidade
- intermediate storage* – Armazenagem (intermédia)
- inventory* – Inventário (*stock*)
- IP – Programação linear Inteira (*Integer Programming*)
- jobshop* – Ambiente *multipurpose* para sequenciamento de produção, no qual nem todos os produtos requerem os mesmos estágios de processamento ou igual sequência de operações
- LHS – Termo independente do lado esquerdo (*left hand side*)
- LINDO – *Linear INteractive Discrete Optimizer*
- lower bound (LB)* – Valor de fronteira, para delimitação inferior
- LP – Programação Linear (*Linear Programming*)
- LP-rel – LP, por relaxação da integralidade em IP
- LRP – Modelo de MILP para planeamento de processos a longo prazo (*long range planning*)
- makespan* – Tempo global de duração do processamento
- MILP – Programação Linear Mista (*Mixed Integer Linear Programming*)
- MINLP – Programação Não Linear Mista (*Mixed Integer Nonlinear Programming*)
- MP – Programação Matemática (*Mathematical Programming*)

- MPC – Campanha de produtos múltiplos (*multiple product campaign*)
- multiple machine* – Processos múltiplos
- multiproduct batch plant* – Unidade de processos descontínuos multiproducto (labora num contexto de *flowsheet*)
- multipurpose batch plant* – Unidade de processos descontínuos multiobjectivo (labora num contexto de *jobshop*)
- NAG – *Numerical Algorithms Group*
- NIS – Não-armazenagem intermédia (*no-intermediate storage*)
- NLP – Programação Não Linear (*Nonlinear Programming*)
- NP-complete – NP-completo (classe de complexidade computacional)
- NP-hard – NP-difícil (classe de complexidade computacional)
- NPV – Valor líquido actual (*net present value*)
- OSL – *Optimization Subroutine Library*
- plan\_milp\_lrp* – Modelo MILP de LRP para planeamento de processos (*plan*) dedicados
- plan\_milp\_lrp\_flex* – Modelo MILP de LRP para planeamento de processos (*plan*), considerando processos flexíveis (*flex*)
- plan\_milp\_lrp\_ref* – Modelo MILP de LRP para planeamento de processos (*plan*), considerando reformulação de restrições (*ref*)
- plan\_milp\_lrp\_schem* – Modelo MILP de LRP para planeamento de processos (*plan*), considerando esquemas de produção flexíveis (*schem*)
- plan\_milp\_spp* – Modelo MILP para planeamento de processos (*plan*), e estocástico (*stochastic pp*)
- plan\_milp\_spp\_ro* – Modelo MILP para planeamento de processos (*plan*), e estocástico (*stochastic pp*) com robustez (*ro*)
- plan\_milp\_spp\_rr* – Modelo MILP para planeamento de processos (*plan*), e estocástico (*stochastic pp*) com restrição ao recurso (*restricted recourse*)
- plan\_minlp\_2ssp\_mest* – Modelo MINLP mestre (*mest*) para planeamento de processos (*plan*), em 2SSP
- plan\_nlp\_2ssp\_sub* – Submodelo (*sub*) NLP para planeamento de processos (*plan*), em 2SSP
- process planning (pp)* – Planeamento de processos
- recourse costs* – Custos de recurso, de segunda fase em 2SSP

- recourse properties* – Propriedades de recurso, de segunda fase em 2SSP
- recourse* – Recurso, relativo à segunda fase em 2SSP
- RHS – Termo independente do lado direito (*right hand side*)
- RO – Optimização robusta (*robust optimization*)
- RR – Restrição aos custos de recurso (*restricted recourse*)
- RTN – Rede de recurso-tarefa (*resource task network*)
- scheduling* – Optimização de tarefas ou operações descontínuas
- single machine* – Processo único
- SOS – Conjunto especial de variáveis binárias onde apenas uma não se anula (*special ordered set*, do tipo 1)
- SP – Programação Estocástica (*Stochastic Programming*)
- SPC – Campanha de produto único (*single product campaign*)
- STN – Rede de estado-tarefa (*state task network*)
- TSP – Problema do caixeiro viajante (*traveling salesman problem*)
- UIS – Armazenagem intermédia não-limitada (*unlimited intermediate storage*)
- UPM – Média parcial superior (*upper partial mean*)
- upper bound (UB)* – Valor de fronteira, para delimitação superior
- VLA – Valor líquido actual (*net present value*, NPV)
- with probability one (w.p. 1)* – Com “probabilidade um”, de carácter assintótico
- ZW – Política de espera nula (*zero wait*),



**Optimização em sistemas de processos químicos:  
generalização de modelos com planeamento e sequenciamento**



## Antecedentes e motivação

A optimização industrial constitui uma vertente importante da actual tendência globalizante assistindo-se, neste ambiente dinâmico e de recursos finitos, a uma evolução no sentido de uma maior complexidade e integração, verificando-se tal evolução também nas dimensões científica e tecnológica.

Os consumos materiais e energéticos duma sociedade em crescimento físico e tecnológico traduzem-se numa aproximação acelerada aos limites das suas possibilidades, com o agravamento de desequilíbrios vários, sejam sociais, económicos, ambientais ou outros. Essa limitação repercute-se também na indústria química, que terá de respeitar constrangimentos ambientais mais rigorosos, mas também deverá procurar satisfazer mercados concorrenciais e exigentes. Nomeadamente, perspectiva-se uma evolução industrial no sentido duma produção flexível, rápida e orientada para o mercado, aplicando processos observadores das normas instituídas, como as ecológicas e de qualidade.

No âmbito da indústria química, uma das vias consiste em inovar e acrescentar valor aos bens e produtos, obtendo assim uma vantagem competitiva para a selecção a efectuar pelos consumidores. Neste enquadramento, revela-se a necessidade de recorrer a técnicas matemáticas, quer para a modelação do problema, quer para a simulação da realidade, quer ainda para a optimização das decisões a implementar. Essas técnicas são bem conhecidas da comunidade científica, mas encontram-se frequentemente subestimadas —ou nem exploradas— pelos agentes que equacionam cada situação.

Este estudo de investigação apresenta uma proposta de desenvolvimento de técnicas e algoritmos, orientados para a resolução de casos problemáticos no âmbito da Engenharia Química, os quais possibilitem uma melhoria significativa relativamente ao tratamento usual dos casos associados aos processos industriais, cuja abordagem muitas vezes se encontra baseada em pressupostos empíricos. Esta investigação dirige-se a técnicas em optimização através de Programação Matemática (*Mathematical Programming*, MP), especialmente em Programação Linear Mista (*Mixed Integer Linear Programming*, MILP), que combina o rigor exigido pela integralidade de variáveis, seja com a flexibilidade de formulação associada a

variáveis binárias (ou zero, ou um), seja com a eficiência e eficácia dos métodos da Programação Linear (*Linear Programming*, LP).

Esta abordagem, baseada em estudos analíticos e de complexidade computacional, conduz a uma metodologia para produção de regras heurísticas, caracterizáveis pela elevada velocidade de resposta, relativamente, quer à resolução exacta do problema matemático, quer à estimativa de soluções subóptimas passíveis de concretização. Assim, potencia-se a disponibilidade e o processamento rápido, através da utilização em tempo real de procedimentos heurísticos, específicos e elaborados em conformidade com cada problema modelado.

Também, deve atentar-se na elaboração de um estudo de optimalidade ao problema equacionado, não só numa crítica de resultados que inviabilize a ocorrência de erros grosseiros, mas efectuando uma análise de sensibilidade que promova a utilização limite de recursos. De certa forma, valoriza-se a determinação (e implementação) de uma gama de valores para os principais indicadores ou parâmetros da situação real, englobando a vizinhança próxima do óptimo matemático em detrimento da obtenção da solução matemática “óptima” e que, mediante relaxação dalguma restrição, não apresenta as desejáveis características de aplicabilidade.

## 1.1 Optimização e Engenharia Química

Desde as últimas décadas, a optimização de base matemática tem vindo a consolidar-se como uma área de investigação autónoma, com métodos próprios, mas com um campo de aplicação transversal a diversas áreas do conhecimento, como sejam as denominadas ciências sociais, ciências exactas, além das ciências de Engenharia, nas mais diversas especialidades (Engenharia Química, Biotecnológica, e também Civil, Mecânica, etc.).

No âmbito da Engenharia Química, as aplicações da optimização matemática dirigem-se principalmente a dois objectos distintos: ao processo unitário propriamente dito (de transporte, reacção, separação), visando o dimensionamento óptimo de equipamentos e as condições operatórias; ao sistema de processos químicos, interessando tanto rentabilizar o seu investimento e funcionamento, como garantir a sua flexibilidade perante condições adversas.

Relativamente aos sistemas de processos químicos aqui visados, note-se que se entende tal como um sistema que permite estruturar os processos químicos como elementos constitutivos, mas de uma forma lata, no sentido em que se possam agrupar diferentes processos químicos (como reacção, separação) ou operações unitárias (sejam mistura, transporte de fluidos, transferência de calor). No âmbito dos sistemas de processos, Grossmann [2004] expande o conceito de *chemical supply chain*, de Grossmann e Westerberg [2000], desde os aspectos microscópicos de síntese molecular aos aspectos macroscópicos da rede global de produção e distribuição da empresa industrial. No entanto, a escala alargada deste conceito acarreta algumas dificuldades, nomeadamente, na definição de fronteiras e no âmbito difuso que implica [Sargent, 2005], pelo que o entendimento aqui seguido é análogo ao de Floudas [2005], quando entende direccionar-se a uma área de aplicação, concretamente, a biologia computacional.

A aplicação dos métodos de optimização em sistemas e em processos químicos, para além da consideração directa de princípios físicos, leis

termodinâmicas e técnicas matemáticas, constitui também um exercício empírico, pois depende dos interesses do decisor, do saber e experiência do utilizador, do cepticismo do investigador. Em termos operativos, o agente responsável pelo estudo deverá incorporar as referidas características, mas também comunicar com os outros agentes, de forma a atingir-se o melhor conjunto de resultados.

**Tabela 1.1** Breve ilustração de estudos de optimização no âmbito da Engenharia Química.

Ano	Autores	Assunto
1957	Walker e Kelly	Desenvolvimento de método CPM no âmbito de fábrica química da DuPont
1967	Erlenkotter e Manne	Localização e expansão, para unidades de produção de fertilizantes na Índia
1975	Sparrow <i>et al.</i>	Dimensionamento de equipamento por métodos heurísticos e <i>branch-and-bound</i>
1980	Himmelblau e Bickel	Expansão óptima para processo de dessulfuração de <i>fuel</i>
1982	Linnhoff <i>et al.</i>	Integração de processos visando a optimização energética ( <i>pinch method</i> )
1985	Boykin	Alocação óptima de produção na Monsanto
1986	Klingman <i>et al.</i>	Planeamento integrado de operações da Citgo Petroleum
1989	Wellons e Reklaitis	Dimensionamento sob incerteza de unidades <i>batch</i> multiproduto
1997	Daichendt e Grossmann	Integração da decomposição hierárquica com Programação Matemática
2000	Ahmed e Sahinidis	Investigação analítica e probabilística de heurísticas em planeamento de processos
2002	Lee e Chen	Optimização do planeamento da produção na BASF, com ferramentas <i>Web-based</i>
2004	Basset <i>et al.</i>	Optimização heurística do desenvolvimento de novos produtos na DOW
2006	Pacheco e Teixeira	Planeamento integrado (da produção, logístico e comercial) na CUF-QI

Na busca de economias de escala [Manne e Markowitz, 1963], verifica-se um sucessivo aumento de dimensão das empresas industriais, com uma crescente especialização dos seus elementos (divisões, departamentos, unidades), observando-se também uma maior complexidade das inter-relações (de administração, gestão, controlo). Da procura de um compromisso entre ambas as tendências resulta a problemática de otimizar esse enquadramento, onde se procuram maximizar ou minimizar diversos factores ou parâmetros, mas respeitando um conjunto de restrições ou constrangimentos, isto é, pretende-se que esses parâmetros de referência atinjam valores extremos, máximos ou mínimos.

Dado o desenvolvimento das ferramentas de cálculo, computadores e algoritmos, verifica-se que também a dimensão e complexidade dos problemas

tratáveis se encontra em significativo crescimento. Assim, possibilita-se o tratamento da situação ou problema fundamentando-o em valores e objectivos quantificáveis, conduzindo-o a índices de desempenho superiores aos da decisão empírica do responsável. A disponibilidade crescente de memória e de velocidade convida, por sua vez, à exploração de novos métodos ou de métodos já conhecidos, mas com novas abordagens: veja-se o método enumerativo do *branch-and-bound*, no âmbito de propostas de decomposição e tratamento paralelo.

Um faseamento ou guia para a resolução prática de um problema de optimização é recomendado por autores como Edgar e Himmelblau [1989]. Sintetizando diversas opiniões, pode afirmar-se que será conveniente:

- detalhar a situação real de origem, problematizando-a;
- lembrar casos e situações típicas de optimização e respectivas características;
- desenvolver o modelo matemático e implementar o algoritmo de cálculo adequados à obtenção da solução “óptima” para o problema;
- diferenciar os enquadramentos de cada solução obtida, cuja implementação deverá ser posteriormente analisada.

Visando a aprendizagem, utiliza-se com frequência o estudo de casos e a análise de projectos similares para encontrar a melhor maneira de formular e resolver os problemas. Logo, interessa focalizar cada aplicação segundo os seus objectivos, discriminar a estrutura da formulação utilizada, induzir as técnicas de resolução apropriadas e estimar a gama de resultados.

Os casos de estudo de optimização, quer em sistemas de processos químicos complexos, quer em organizações que integrem fornecimento, produção e distribuição, são já frequentes na literatura. É também devido à divulgação dos elevados benefícios aí recolhidos que se tornou possível que a optimização em ambiente industrial constitua actualmente um procedimento de rotina. Ilustrativamente, apresenta-se na *Tabela 1.1* uma breve resenha de estudos de optimização no âmbito da Engenharia Química, seleccionados pela sua reconhecida relevância, científica e económica. Em termos de resenhas, Biegler e Grossmann [2004] apresentam uma retrospectiva sobre optimização, seguida de uma perspectiva dos mesmos autores [Grossmann e Biegler, 2004], onde abordam os recentes desenvolvimentos em optimização global e de natureza lógica, referem algoritmos dirigidos a aplicações de grande dimensão, bem como descrevem as implicações dos progressos computacionais.

### 1.1.1. Modelação matemática do problema

Após a problematização da situação real, efectuada através da percepção dos efeitos a rectificar, trata-se da modelação matemática, ou seja, convirá formular uma representação —selectiva e simplificada— do problema, exprimindo matematicamente essa formulação. Tal como foi descrito, o processo de enquadramento do problema engloba, para além da observação pormenorizada da situação, a recolha de informação quantificada e validada.

A formulação do problema, ou problematização, compreende uma análise fundamental, pois envolve a construção de hipóteses e a consideração de

simplificações várias. Aqui, interessa seleccionar os aspectos relevantes da situação, equacionando-os através da definição das decisões a empreender e do(s) objectivo(s) de interesse. Então, é necessária a formulação de um modelo matemático que represente devidamente o problema equacionado, isto é: *i)* constrói-se um modelo matemático que procure abstrair a essência do problema real; *ii)* supõe-se que o modelo constitui uma representação suficiente; *iii)* pretende-se que as soluções e conclusões assim obtidas sejam válidas no contexto real.

Repare-se que a solução óptima representa a melhor solução para o problema matemático, mas pode até nem constituir uma boa solução a implementar, caso não se tenha problematizado adequadamente a situação real. No entanto, perante uma apropriada formulação do modelo e interpretando os seus resultados, obtém-se informação válida para apoiar a tomada de decisões.

A formulação da função objectivo é um passo crucial, pois requer uma medida de desempenho, bem como a expressão dessa medida como função das variáveis de decisão. Usualmente, a função objectivo deve ser desenvolvida visando aspectos económicos, dado que podem pressupor-se métodos alternativos para avaliação do lucro, envolvendo parâmetros diversos, entre outros: os custos das matérias-primas, os valores dos produtos finais, os custos de produção como funções das condições de operação, valores estimados das vendas futuras e probabilidades associadas. Note-se que, para uma mesma situação problemática, poderão ser equacionados objectivos operacionais alternativos ou conflitantes.

As acções e operações necessárias para a situação “óptima” deverão ser também incluídas no modelo matemático, pois apenas desta forma será possível reproduzir o comportamento dinâmico e estático do sistema. Ainda serão de considerar, numa abordagem integradora, as relações entre a área em análise e outras áreas contíguas, e todo aquele constrangimento de índole operacional, técnica ou económica, cuja não inclusão desvirtue os efeitos do sistema e o valor objectivo. Contudo, os detalhes a incorporar no modelo serão tais que permitam a obtenção em tempo útil de resultados de boa qualidade, atendendo à implementação computacional, resolução numérica e tratamento final.

Em conformidade com Eppen *et al.* [1993], a utilização do modelo está relacionada com a natureza dos dados e dos resultados. Considerando as finalidades gerais da utilização de modelos numa qualquer empresa industrial:

- a nível estratégico, utilizam-se modelos probabilísticos e de optimização, onde as variáveis de decisão correspondem a indicações genéricas, levando em conta a incerteza associada aos dados; os modelos são utilizados como instrumentos de planeamento com funções de previsão, de exploração de alternativas, para desenvolvimento de planos de contingência, com vista ao aumento de flexibilidade ou diminuição de tempo de resposta;
- a nível operacional, utilizam-se modelos de decisão e determinísticos, cujas soluções indicam as decisões a implementar, pois os dados são conhecidos e específicos, os objectivos são claros e explícitos, as interacções são quantificáveis.

### 1.1.2. Desenvolvimento de procedimentos heurísticos

Os procedimentos heurísticos buscam solucionar intuitivamente um dado problema, visando as características que a solução desse problema deverá apresentar. Frequentemente, desenvolvem-se modelos de optimização inteira e mista, tratáveis

quando de dimensão limitada, mas que requerem um crescimento exponencial do esforço computacional quando a dimensão do problema aumenta. Em problemas de grande dimensão, onde muitas vezes não se poderá garantir a solução óptima, a utilização de heurísticas permite obter boas soluções para as variáveis inteiras, ainda que subóptimas. Não raras vezes, apenas será possível estimar o óptimo a menos dum desvio tolerável da solução calculada; também, aplicando esquemas de relaxação linear ou lagrangeana, para delimitação inferior (superior) do valor óptimo da função objectivo, melhora-se a eficiência do cálculo para problemas de grande dimensão —as heurísticas constituem frequentemente a primeira fase de processos de cálculo de grande dificuldade. Deste modo, interessam ferramentas de cálculo simples e efectivas, que conduzam a soluções de boa qualidade, em alternativa a aproximações baseadas apenas em aspectos teóricos, de grande complexidade e exigindo elevado esforço computacional.

Havendo que conjugar características dos métodos heurísticos com as dos algoritmos exactos, utiliza-se investigação analítica para caracterizar os respectivos desempenhos. A análise das heurísticas, teórica ou probabilística, permite minorar os seus pontos fracos: por um lado, promove maior precisão, ou busca a exactidão do algoritmo eficiente; e por outro lado permite a resolução apenas aproximada dos problemas de optimização com elevada combinatoriedade, mas de modo a não prejudicar a qualidade da solução.

Os conhecimentos obtidos, de natureza analítica ou empírica, são direccionados para a adequação de métricas para optimização com robustez da solução, tratando o risco do sistema modelado, incorporando as incertezas existentes nos dados, ou considerando as aproximações efectuadas em diversos parâmetros. É possível a produção de heurísticas a partir de aproximações que emulem os efeitos pertinentes do sistema, especialmente quando se consideram cálculos mais complexos, por exemplo, de Programação Estocástica (*Stochastic Programming*, SP) em subsistemas da síntese de processos.

### 1.1.3. Estudo da optimalidade

Uma limitação reside na eventual natureza estática da solução do problema de optimização, pois, quando se determina a solução óptima, é explicitado o melhor conjunto de valores para as variáveis de decisão, o qual se encontra associado a um conjunto de parâmetros ou dados. Sendo atribuído um único valor numérico a cada variável, esse valor poderá ser válido apenas enquanto os dados da instância original do problema não se alterarem.

Os parâmetros sensíveis são aqueles que influenciam a solução do problema, mesmo a partir de pequenas variações. A sua identificação é importante, pois estes parâmetros exigem boa qualidade numérica, dado que os seus valores não podem ser modificados sem provocar flutuações. Donde, para evitar distorções sobre os resultados finais, será necessário colocar o maior rigor na estimativa dos valores para esses parâmetros.

Assim, a análise dos parâmetros sensíveis permite obter uma sequência de aproximações ao curso de acção ideal, pois uma solução inicial menos boa sugere afinações nos dados, melhorias no modelo, ou alterações no procedimento de resolução. Após determinação da solução óptima deve proceder-se a um estudo de optimalidade para determinar quais dos parâmetros do problema são críticos ou sensíveis para a implementação dessa solução. A finalidade seria obter uma solução que mantivesse o seu carácter de optimalidade perante as diversas combinações da

gama de valores expectáveis dos parâmetros sensíveis, ou seja, pretende obter-se uma solução robusta.

A robustez da solução implica que a solução óptima do modelo seja também uma solução de boa qualidade para qualquer um dos cenários. A robustez do modelo refere-se concretamente à solução óptima, que não deve apresentar menos-valias significativas em virtude da variação intrínseca ao programa estocástico. Exemplificando, para o planeamento da produção com incerteza nas previsões de procura: uma solução robusta deve apresentar valores diminutos para a não-satisfação da procura, bem como para o excesso de produção.

Uma decisão é designada como robusta caso o custo actual do cenário especificado permaneça “próximo” do valor esperado óptimo para todos os cenários. Usualmente, minimiza-se a variância dos custos dependentes dos cenários, simultaneamente com a minimização da respectiva esperança, pelo que a robustez é reforçada através da incorporação duma função de penalidade da sensibilidade da função objectivo aos parâmetros incertos. Assim a incerteza associada com a previsão de dados apresenta-se como uma função da matriz de variância-covariância para a previsão dos erros.

Apresenta-se o enquadramento de modelação para optimização robusta [Malcolm e Zenios, 1994], em dupla fase, onde se distinguem dois tipos de variáveis:

- variáveis de projecto — são as variáveis relacionadas com as restrições estruturais e determinísticas do problema, e que não dependem dos parâmetros tomados como incertos;
- variáveis de controlo — podem ser ajustadas segundo os valores dos parâmetros incertos em cada cenário, pelo que o respectivo valor óptimo dependerá tanto desses parâmetros como das próprias variáveis de projecto.

Nesta abordagem em dupla fase, trata-se da partição das variáveis do problema em duas fases: as variáveis de projecto, cujo valor é decidido *a priori*, e a partir das quais se revelam os parâmetros incertos; e as variáveis de operação e controlo, que se apresentam como variáveis aleatórias devido à incerteza associada. Do exposto, resulta um modelo de optimização estocástica onde se procura numa segunda fase correctiva, designada como de recurso (*recourse*), minimizar a soma dos custos de projecto da primeira partição com os valores esperados dos custos de recurso.

Este enquadramento de Programação Estocástica em Duas Fases (*Two-Stage Stochastic Programming*, 2SSP), com promoção da robustez, será visado oportunamente para um estudo da optimalidade, por generalização de modelos específicos de: *i*) planeamento de processos; *ii*) dimensionamento e sequenciamento de processos descontínuos.

## 1.2 Generalização de modelos

A utilização de modelos generalizados —pois decorrentes de outros— permite uma modelação mais completa, integrando os aspectos de interesse dos modelos primitivos e complementando-os de forma a melhorar a sua aplicabilidade. No entanto, verifica-se o aumento da complexidade do modelo generalizado

resultante, cuja resolução óptima poderá recorrer às características da solução do modelo primitivo, quer numa abordagem exacta quer aproximativa.

Em termos de modelação via MP, a generalização do objectivo a otimizar pode assumir diferentes formas, configurando uma optimização multicritério: uma minimização de custos integra-se com uma maximização de receitas, ou da diferença correspondente, os lucros; uma abordagem determinística complementa-se com a incerteza inerente a um programa estocástico; e num programa estocástico, inclui-se o tratamento do risco, com promoção da robustez.

Em termos do espaço de soluções admissíveis, a implementação de novas restrições permite formular outros aspectos, como sejam os de cariz operacional, considerando, por exemplo: as políticas de planeamento de produção, integradas em problemas de sequenciamento; o tratamento dos inventários multiperíodo, integrados numa rede de distribuição onde se consideram os constrangimentos do transporte.

Saliente-se a dificuldade inerente à generalização de modelos, pois se encontra limitada pela capacidade de que se dispõe para o tratamento dos subproblemas. Isto é, perante um subproblema cuja complexidade computacional impede a sua eficiente resolução exacta, não desaparece a necessidade da sua incorporação num modelo mais complexo. Devem então assumir-se metodologias de resolução aproximadas, as quais permitam a quantificação da qualidade da solução subóptima, ou duma gama de soluções aplicáveis. Nomeadamente, na síntese de processos [Daichendt e Grossmann, 1997] é possível incorporar simultaneamente os submodelos respeitantes aos processos de reacção, separação e de integração energética, adaptando a metodologia de resolução em conformidade com o detalhe da modelação: *i)* inicialmente, assumindo-se apenas perturbações lineares entre os submodelos, através duma sua formulação simplificada; *ii)* com o decorrer da resolução, integrando aspectos mais complexos e exigindo sucessivamente resoluções mais completas dos submodelos. Note-se que a consideração de uma (super) estrutura global, integradora de (sub) estruturas específicas através duma optimização simultânea, é naturalmente superior à optimização sequencial, onde os resultados duma etapa serão utilizados na etapa seguinte: é que a estrutura global considera —e incorpora— a configuração óptima obténivel através da optimização sequencial das estruturas específicas.

Como exemplo ilustrativo da abordagem usual da generalização de modelos, considere-se o problema de dimensionamento óptimo de lotes (*uncapacitated economic lot size*, ELS, descrito em Anexo 1), cuja rapidez da optimização constitui um dos principais atributos da sua estrutura matemática: a sua utilização como subproblema em cálculo intensivo é bem reconhecida. Nomeadamente, as igualdades nos balanços materiais e a imposição dos limites globais à produção configuram uma dupla restrições que se reconhece facilmente.

A generalização do modelo de ELS aos problemas tratados na Engenharia Química apresenta como fundamento a adequação deste modelo ao sequenciamento dos processos de produção descontínuos (*batch scheduling*). No entanto, através de algoritmos de decomposição ou de reformulações, que permitam desenvolver as propriedades que o caracterizam (por um lado, a unimodularidade da matriz das restrições; por outro lado, a constância da relação hierárquica entre custos), é usual também aprofundar a aplicabilidade do ELS a outros problemas. Particularmente, releva-se a aplicação deste modelo em termos da coordenação entre as funções de produção, de distribuição e inventário, na localização de unidades (*location*) ou na expansão de capacidades (*capacity expansion*), temas bem actuais e em investigação

permanente. No entanto, a simples adição de um grupo de restrições delimitador das capacidades específicas de produção torna o caso capacitado (*capacitated* ELS) num problema de complexidade computacional bem mais exigente (NP-hard) do que o caso não-capacitado (*uncapacitated*). Assim, as características de resolução não são generalizáveis caso não se mantenham os seus fundamentos (unimodularidade da matriz, relação entre custos), pelo que é curial apreciar a sua estrutura matemática.

### 1.3 Objectivo e Organização da Tese

No âmbito das metodologias de Optimização aplicadas em sistemas de processos químicos, este trabalho visa contribuir com uma abordagem de generalização a modelos de Programação Linear Mista (MILP), como acima referido.

Dada a amplitude de aplicação, bem como a necessidade de avaliar a estrutura de cada modelo matemático que se pretenda generalizar, seleccionam-se da literatura dois casos de interesse, no sentido dos modelos desenvolvidos por vários autores: o caso do planeamento de processos (*process planning*), na linha de Sahinidis *et al.* [1989]; e o do dimensionamento e sequenciamento de processos descontínuos (*design and scheduling of batch processes*), segundo Voudouris e Grossmann [1992].

Procede-se à implementação computacional de modelos de interesse, pretendendo-se inicialmente certificar tal implementação, para depois se proceder à sua generalização. Como, previsivelmente, a generalização dos modelos acarreta uma maior dificuldade de resolução, implementam-se diversas heurísticas, bem como as aplicações que suportam as ilustrações numéricas.

Aquelas heurísticas decorrem de abordagem teórica, dirigida à complexidade computacional dos modelos seleccionados. Tal permite também um melhor conhecimento dos modelos, tanto dos seus objectivos como dos seus constrangimentos, o que é essencial para a consistência da generalização a promover.

Propõem-se, assim, modelos generalizados e referentes aos casos seleccionados, abordando tanto a hipotética ocorrência de uma (super) estrutura alternativa, como um enquadramento estocástico com promoção de robustez.

Dado o fluxo de informação intrínseco ao presente trabalho, assume-se uma forma modular para a apresentação das diversas etapas, correspondendo aos Capítulos 2 a 5, os quais são finalizados com uma sinopse. Dada a diversidade de áreas focadas (de fundamentos como o matemático, numérico, computacional, probabilístico) é introduzido um glossário de designações, o que facilita a apreensão de alguns termos. Também, dado que se assume uma continuação de trabalhos da literatura, anotam-se os pontos relevantes do trabalho aqui desenvolvido.

A restante parte desta *Tese* está organizada da seguinte forma:

- No Capítulo 2, “**Casos de Optimização por Programação Linear Mista em sistemas de processos químicos**”, analisam-se relevantes aplicações deste tipo de Programação Matemática no seio da indústria química, revê-se o estado da arte neste âmbito, visando-se especificamente:
  - o planeamento de longo prazo em sistemas de processos químicos (*process planning*), com tratamento estocástico— trata-se a problematização e a qualidade da modelação efectuada em ambiente incerto;

- o sequenciamento óptimo do ciclo de produção, para desenvolvimento de sistemas de produção descontínua (*design and scheduling of batch processes*) em ambiente multiproduto (*flowshop*) — prossegue-se a generalização dos modelos, na perspectiva de incorporação do sequenciamento de processos descontínuos no dimensionamento desses mesmos processos.
- No Capítulo 3, “**Implementação computacional**”, descreve-se detalhadamente a implementação de modelos e métodos, para:
  - abordando os casos de estudo, implementar métodos de resolução específicos do tipo de problemas de MILP;
  - promovendo uma extensa experiência de implementação computacional, enquadrar cada um dos problemas equacionados;
  - promover a aplicação de técnicas de suporte.
- No Capítulo 4, “**Análise dos modelos e heurísticas**”, trata-se cada problema considerado, com base na sua estrutura matemática:
  - investiga-se teoricamente e analisa-se a complexidade computacional dos problemas em estudo;
  - através de heurísticas aqui desenvolvidas, obtêm-se conhecimentos significativos quanto à estimativa da solução inteira óptima do problema em análise;
  - desenvolvem-se estimativas de boa qualidade do valor óptimo do problema, que se verifica probabilisticamente.
- No Capítulo 5, “**Estudo numérico da optimalidade**”, promove-se a robustez de cada problema considerado, por tratamento da incerteza através de Programação Estocástica; globalmente, comprova-se a adequação das abordagens desenvolvidas, de maneira a verificar o seu interesse para a obtenção de respostas robustas, flexíveis e efectivas para as situações problematizadas.
- No Capítulo 6, “**Conclusões e projecções**”, finaliza-se e, além de discriminar os principais resultados do estudo:
  - dado o interesse que os assuntos focados (os problemas, os métodos de resolução) desde sempre têm suscitado, descrevem-se ainda alguns desenvolvimentos recentes e afins para o trabalho efectuado.
  - apontam-se, assim, possíveis vias para posteriores desenvolvimentos, e referem-se diversas limitações constatadas e a que importaria obviar;

Ainda: no Apêndice 3 <sup>1</sup>, descrevem-se e ilustram-se diversas aplicações de suporte (computacional); nos Apêndices 4-A e 4-B, relativos a modelos de dimensionamento e sequenciamento, sucessivamente, desenvolvem-se estudos analíticos e de heurísticas, incluindo-se dados numéricos representativos da análise probabilística realizada; nos Apêndices 5-A, 5-B e 5-C, respeitantes ao modelo robusto de planeamento de processos, apresentam-se sequencialmente aspectos

---

<sup>1</sup> Sobre a designação de Apêndices e Anexos, veja-se Índice.

---

marcantes do modelo generalizado, dados dos exemplos numéricos e decomposição de Benders; nos Apêndices 5-D e 5-E, associados ao modelo generalizado para dimensionamento e sequenciamento de processos descontínuos, quantificam-se os exemplos numéricos.

Finalmente, no Anexo 1, encontra-se uma introdução ao modelo ELS; nos Anexos 4-A e 4-B, respectivamente, procede-se a uma breve resenha da complexidade computacional e dos procedimentos heurísticos de aproximação.





# Casos de Optimização por Programação Linear Mista em sistemas de processos químicos

A optimização de sistemas processos químicos constitui-se como um objectivo inacabado, quer numa óptica de investigação e em termos duma abordagem global ou combinatória, quer atendendo à concretização, onde haverá que percorrer a distância entre teoria e prática.

No actual mercado aberto, a modernização industrial acompanha a dinâmica da fileira informática, sendo desafiadora dos padrões de flexibilidade e de eficiência. Busca-se a melhoria da qualidade, direccionada tanto à normalização de produtos como a melhorias nos processos de produção e de controlo ou, mesmo, num sentido lato, à fluidez de informação desde os clientes até aos fornecedores.

Em última análise, pretende-se que o valor adicionado reverta em favor da comunidade. Para tal contribui a melhoria da eficiência dos processos de produção industrial e subsequente redução de consumo de recursos materiais e energéticos, que ficarão disponíveis para outros propósitos. Também, essa mais-valia se repercutirá na redução do correspondente impacto ambiental. As inovações decorrentes da investigação devem ser aplicadas na forma tecnológica, obtendo-se não só benefícios económicos directos, mas também benefícios sociais e ecológicos, ou seja, uma melhor qualidade de vida. Ora, a minimização do consumo de recursos naturais e dos níveis de poluição são finalidades estratégicas do crescimento económico sustentável, o que se traduz na criação de um sistema eficiente de recolha de resíduos. Elevada atenção tem sido prestada a este tema, especialmente no ramo da indústria automóvel, na indústria de produção de embalagens, na metalomecânica e na electroquímica, ou nas indústrias do vidro e do papel, onde se pretende planificar todo o ciclo de vida dos materiais utilizados.

A investigação e o desenvolvimento das operações relativas aos processos têm-se generalizado, desde o estudo das actividades logísticas e de distribuição física dos produtos —numa escala macroscópica, em sentido lato— até à análise e melhoria das metodologias, na inovação de materiais e na implementação dos seus processos de fabrico —mesmo numa escala microscópica [Grossmann, 2005]. Assim, no desenvolvimento de novos produtos de especialidades químicas, farmacêuticos ou

agroquímicos, torna-se já usual o planeamento integrado e o sequenciamento desde a investigação química fundamental até ao processo de fabrico, aspectos conducentes à rápida disponibilização do novo produto.

De facto, o sentido da evolução industrial aponta para a diminuição do tempo decorrido entre a inovação tecnológica e a aplicação comercial, seja em produtos de grande utilidade, seja em produtos substituintes que incorporem alguma vantagem. São expressivos os esforços para relacionar o produto e a engenharia do processo de produção, pretendendo ligar-se o desenvolvimento do processo com o fabrico do produto. Tal melhoria permitirá a diminuição do tempo de resposta às necessidades do mercado: encurtamento do ciclo decorrido desde a ideia até ao produto final, que se traduzirá em termos de competitividade empresarial, bem como na ampliação das margens de lucro. Convirá, assim, tanto procurar a melhor metodologia para sistematizar o esforço de concepção, como planear esse esforço em termos de capacidades e recursos.

Na análise dos problemas de optimização, utilizam-se frequentemente casos ilustrativos e analogias, no intuito de encontrar a melhor forma para abordar e resolver esses problemas. É com esta finalidade que, ao longo deste Capítulo, se menciona uma variedade de casos. Neste âmbito e com carácter ilustrativo, refere-se o desenvolvimento de um caso de referência para planeamento e sequenciamento de produção na indústria química: tal caso foi desenvolvido, conjunta e internamente por especialistas da Bayer e da BASF em 1994, sendo apenas publicado alguns anos mais tarde [Kallrath, 2002a]. Também, o mesmo autor [Kallrath, 2002b] descreve a aplicação com sucesso de um modelo MILP dirigido à rede com várias instalações de produção (*multisite network*) e envolvendo o planeamento de produção, mas também a planificação da cadeia de fornecimentos (*supply chain*), pormenorizando os aspectos de transporte e financeiros.

De referir que Ribeiro e Nunes [2001] coordenaram um estudo sobre a indústria química em Portugal, onde além de situarem os principais produtos e mercados, referirem as principais tendências e as variáveis com maior impacto sobre o desenvolvimento da indústria de processos químicos, também apontam as principais linhas de acção futura.

Ora, ocorrendo casos de sucesso em temas diversificados, deve salientar-se que os problemas formulados através de MILP, à semelhança do que ocorre com a LP, abrangem diversos ramos da Engenharia e outras áreas científicas. Deste modo, os temas que se apresentam em seguida, formam a base para o presente trabalho de investigação.

- a) O problema de planeamento de processos (*process planning*), no âmbito de sistemas de processos químicos, problema cuja vertente multiperíodo, num horizonte de médio e longo prazo, envolve:
  - análise e definição de investimentos em novas fábricas ou processos;
  - definição da linha de produtos, bem como dos níveis de produção;
  - valorização de materiais;
  - promoção da flexibilidade, da robustez e da eficiência dos processos.

- b) O problema de dimensionamento e sequenciamento de processos descontínuos (*design and scheduling of batch processes*), que visa integrar no dimensionamento os aspectos de curto prazo onde se pretendem sequenciar operações *flowshop*, de forma a observar determinados níveis de produção, mas mantendo a utilização de recursos ao mínimo. Isto é, no âmbito dum horizonte também multiperíodo visa-se também considerar aspectos operacionais como:
- sequenciar os ciclos e as campanhas de produção;
  - otimizar e controlar inventários (*inventory*);
  - especificar as diferentes operações do fabrico (arranques, interrupções, manipulação, manutenção);
  - minimizar o ciclo temporal ou o período de campanha da produção;
  - determinar a política óptima, como para espera nula (*zero wait, ZW*) ou armazenagem finita (*finite intermediate storage, FIS*).

Esta súpula dos temas em estudo não é mais que uma breve introdução. De facto, inúmeras vezes são relatados problemas com características de mais do que um tipo de modelo matemático, devido a interacções não removíveis. Como por exemplo: para proceder ao sequenciamento em processos descontínuos, é necessário dispor dos parâmetros de capacidades e de taxas de produção, pelo que se pressupõe o dimensionamento desses processos; e, vice-versa, no dimensionamento desses processos descontínuos, é essencial dispor de valores óptimos para os tempos do ciclo e para a programação das campanhas de produção, de forma a proceder-se ao respectivo dimensionamento. Note-se que a remoção de tais interacções (seja dimensionando os processos no problema de sequenciamento, seja predefinindo os lotes de produção no problema de dimensionamento), corresponde matematicamente à especificação de instâncias de um problema mais lato: o da optimização simultânea por dimensionamento e sequenciamento de processos. Assim, para desenvolvimento do sistema de processos, deve-se procurar a integração ponderada de diversos subsistemas, sob pena de redução da qualidade da solução (subóptima) obtida.

Na sequência dos aspectos já referidos, procura-se aqui um tratamento aplicável aos problemas de optimização da Engenharia Química, modeláveis por MILP, reavaliando as suas finalidades, recuperando as técnicas de resolução e reanalisando os resultados. Obviamente, interessa também revalidar as modelações da literatura, verificando as especificidades do problema e assegurando assim o realismo. Neste Capítulo, embora apenas se indiquem as técnicas com melhor potencial para a resolução dos problemas em análise, não se deixam de analisar problemas ainda não completamente resolvidos, e referem-se áreas práticas onde será útil mais investigação. Especificamente, pretende-se:

- actualizar conhecimentos, analisando a problematização e a modelação dos temas de interesse;
- identificar aspectos limitativos nos métodos de resolução;
- direccionar o desenvolvimento sistemático de regras heurísticas.

## 2.1 O planeamento de processos (*process planning*)

Nesta secção, aborda-se o planeamento de médio e longo prazo na indústria de processos químicos, adoptando-se a optimização multiperíodo de investimentos e níveis de produção, com estudo da variabilidade —devida a ambiente incerto, dinâmica competitiva, efeito de aproximações, robustez da solução— inerente aos alargados horizonte temporal e ordem de grandeza.

Anotam-se aspectos do planeamento de investimentos na indústria de processos (indústrias petrolífera, de fertilizantes, outras), procurando analisar as origens de cada situação, donde se especifica o problema matemático que permite associar a solução óptima à actividade operacional. A compreensão das causas é importante, pois a efectividade da solução depende da modelação; também, o balanceamento entre os factores críticos para o problema, permitirá implementar a solução com maior robustez.

Dado o tratamento da incerteza inerente a um longo horizonte temporal, o cálculo reveste-se de um carácter probabilístico que exige um esforço computacional exponencial. Usualmente, o problema estocástico engloba o correspondente problema determinístico num ciclo interno, pelo que a velocidade da resolução deste repercute-se na resolução do programa estocástico, sendo crítica a rapidez de cálculo. Deste modo, pretende antever-se o desenvolvimento de regras heurísticas e esquemas alternativos de cálculo que permitam minorar as dificuldades numéricas.

Revêem-se trabalhos anteriores na área do planeamento de processos, decorrendo daí abordagens alternativas, cuja comparação analítica e experimental será efectuada em fases posteriores do estudo.

Na presente secção, revê-se o planeamento de longo prazo que visa a programação de investimentos em sistemas de processos e que, numa perspectiva dupla, integra aspectos de:

- planeamento operacional dos processos, quando encarado através duma discretização temporal multiperíodo, recorrente e combinatória, aproximando-se dum problema de sequenciamento (*scheduling*);
- desenvolvimento conceptual da rede de processos, constituindo também um problema de síntese (*process synthesis*), onde se procede igualmente à extracção da solução óptima a partir duma superestrutura.

Neste enquadramento, deve promover-se o conhecimento da evolução dos factores críticos para o planeamento de processos, e deve ser abordado o carácter dinâmico (multiperíodo) do sistema, ao ponto de se gerarem sistemas operacionalmente óptimos.

### 2.1.1 Aspectos do planeamento de processos

O planeamento de processos pretende visar os objectivos de longo prazo (normalmente, sobre um período superior a um ano), bem como prever os meios necessários àqueles objectivos: definir os investimentos em novas unidades fabris, expansão de unidades existentes, localização e dimensionamento de novas fábricas ou postos de distribuição, desenvolvimento de novos produtos, *etc.*. Os modelos para estes planos estratégicos não têm de ser muito detalhados, mas devem possuir características de simplicidade, flexibilidade, rapidez, e acessibilidade aos dados. Para suporte à decisão, importa o horizonte temporal onde as previsões são

necessárias, de forma a definir-se o intervalo de tempo onde a decisão se mantém óptima. Note-se que o número de períodos que constitui o horizonte de planeamento deve possibilitar a comparação entre decisões com diferentes implicações económicas, e a imprecisão da previsão deve ser tratada de forma conveniente, conduzindo a resultados cuja fiabilidade acompanhe a grandeza das decisões a tomar.

No planeamento de investimentos, os efeitos financeiros decorrentes da incerteza futura são incluídos no cálculo —usualmente, do valor líquido actual, VLA (*net present value*, NPV)— através da utilização da função exponencial decrescente. Assim, os valores futuros têm menor ponderação para a tomada de decisões, sendo usual a partição do horizonte em períodos desiguais: na literatura, pode encontrar-se um horizonte de dez anos dividido em três períodos de tempo, ordenadamente, com dois, três e cinco anos. A função objectivo habitualmente focaliza aspectos económicos, e podem supor-se métricas alternativas para avaliação do plano ao longo de todo os períodos, como seja: maximizar lucros, minimizar custos, otimizar valor esperado, diminuir variabilidade, penalizar incumprimento de restrições “fracas”.

Como critério económico de avaliação de projectos de investimento é habitualmente utilizado o critério do VLA, que numa perspectiva anual considera os seguintes parâmetros [Biegler *et al.*, 1997]:

$CI$  – custo de investimento;

$R$  – montante da venda dos produtos;

$CO$  – custo operacional;

$C_{inv}$  – custo de inventário;

$intx$  – taxa de actualização (*interest rate*);

$Ny$  – número dos períodos de tempo de vida do projecto;

$tx$  – taxa de impostos;

$restx$  – valor residual, em fracção (taxa) do custo de investimento;

e que apresenta forma geral,

$$NPV = (R - CO - C_{inv}) \cdot (1 - tx) \cdot \frac{1 - (1 + intx)^{Ny}}{intx} + \frac{CI}{Ny} \cdot tx \cdot \frac{1 - (1 + intx)^{Ny}}{intx} + \frac{restx \cdot CI}{(1 + intx)^{Ny}} - CI$$

Para uma correcta avaliação do planeamento, é curial dispor de medidas fiáveis para avaliação dos custos. Usualmente, em cada processo de produção, supõe-se que as actividades de transformação são representadas por funções contínuas e suaves, considerando que se mantêm a implementação e o equipamento associado. No entanto, repare-se que as estimativas dos custos correspondentes devem representar os custos económicos de oportunidade, pelo que tais custos devem ser conformes com o modelo e as decisões em análise.

No respeitante à decisão sobre operações de produção, os modelos de custos são de difícil construção, pois custos que numa perspectiva de curto prazo são considerados de natureza fixa, podem assumir carácter variável quando considerados num prazo mais longo. Em termos de MILP, uma das utilizações frequentes das variáveis binárias tem a ver com os parâmetros de custo fixo (*fixed charge*), ou seja,

referem-se aos custos em que se incorre independentemente do nível de actividade. Podem enquadrar-se nesta categoria os custos de implantação, arranque e manutenção duma unidade industrial, enquanto que os custos directos de produção, referentes às matérias e utilidades necessárias e que variam com as quantidades, se definem como custos variáveis. Neste contexto de ajuste dos valores económicos aos níveis produtivos, Williams [1985] alude à partição da função dos custos através de LP, para simular o efeito de economias de escala, o que também pode ser concretizado utilizando variáveis binárias.

As primeiras aplicações industriais de LP e de MILP datam dos anos cinquenta (do século passado), e referem-se a problemas em refinação de petróleo, processamento de produtos alimentares, planeamento de fabrico de aços, programação de projectos. Obviamente, esses métodos generalizaram-se a outras aplicações em processos industriais, na produção de materiais químicos, papel, detergentes, gases industriais.

Desde o advento da LP, que se tratou um elevado numero de problemas, mesmo que não se coadunassem com este tipo de Programação Matemática. Num estudo prévio sobre a expansão da capacidade de produção de fertilizantes na Índia, onde se desejava estabelecer um conjunto de novas unidades produtivas bem como fasear os respectivos aumentos de capacidade, Erlenkotter e Manne [1967] aplicam o conhecido modelo de transporte (*transportation*) para minimizar o custo, tanto dos fluxos das matérias-primas desde as suas fontes para as unidades de produção de fertilizantes, como dos fluxos destes produtos até aos mercados. Desta forma, a localização das unidades decorre como uma minimização global dos custos de transporte, em alternativa a utilizar um modelo matemático de Programação linear Inteira (*Integer Programming*, IP) específico para a localização de unidades, cuja resolução seria sem dúvida bem mais complexa. Para a definição dos períodos de tempo onde proceder à expansão da capacidade das unidades implementadas, aqueles autores utilizam um ciclo iterativo de cálculo. No entanto, após esse estudo de natureza matemática haveria ainda que se proceder a um plano detalhado para implantação das unidades produtoras de fertilizantes. Aí, verificou-se a impossibilidade técnica de uma das localizações seleccionadas, devido à incapacidade de crescimento futuro dos fornecimentos de matérias-primas, a partir das fontes que lhe estavam alocadas. Então, atendendo à previsão de aumento das necessidades em matérias para acompanhar os sucessivos aumentos de produção, acabou por ser implementada uma unidade numa localização que, aquando do procedimento inicial, tinha sido posta de lado.

Rudd e Watson [1968] referem um caso algo semelhante, mas cuja acção decorreu na Islândia: no ajustamento da capacidade de produção às necessidades de fertilizante, as recomendações da optimização matemática (implementar uma unidade com capacidade para produzir 5400 toneladas por ano, seguindo-se uma expansão da capacidade em 1900 toneladas anuais em cada período de 12 anos) apenas foram tomadas como sugestões, dada a omissão de factores relevantes no cálculo efectuado. O que aconteceu foi a implementação de uma unidade para produção de fertilizantes com uma capacidade de 8400 toneladas anuais, com previsível duplicação posterior: ocorreu um crescimento da procura superior ao previsto, traduzindo um efeito autocatalítico da disponibilidade do fertilizante, facilitando os ajustes às encomendas e o cumprimento dos prazos de entrega, permitindo diminuir os custos de transporte globais e acomodar variações sazonais.

Logo, é essencial uma compreensão das circunstâncias envolvendo o problema, prevendo efeitos não visíveis mas prováveis, o que exigirá uma análise mais detalhada do que a efectuada para um estudo matemático. Por exemplo, repare-se na localização de uma unidade cimenteira em plena zona natural da Arrábida e na proximidade de zonas densamente povoadas, que uma análise rápida consideraria desaconselhável. Porém, atendendo ao elevado custo de transporte dos materiais, à distância (curta) aos mercados de consumo de cimento, à disponibilidade de transportes (via rodoviária, marítima), aquela localização parece justificar economicamente a menor qualidade ambiental [Coulson e Richardson, 1989].

Na indústria da pasta de papel, é conveniente garantir a sustentabilidade das produções florestais. Dado os elevados períodos de tempo necessários ao crescimento do plantio, e para que se atinjam os níveis de corte desejados, é conveniente minimizar o risco associado aos elevados custos de investimento, por exemplo, através de métodos estocásticos que permitam simular mudanças futuras, promover as reavaliações necessárias, e otimizar a política de investimentos florestais. Por exemplo, González *et al.* [2005] investigam os efeitos do risco de incêndios sobre o planeamento florestal em Espanha, enquanto Penttinen [2006] analisa o impacto dos preços sobre a rotação na floresta finlandesa.

**Tabela 2.1** Sinopse dos estudos no âmbito do problema de planeamento de processos.

Ano	Autores	Assunto
1989	Sahinidis <i>et al.</i>	Modelo LRP, para planeamento de longo prazo na indústria química (processos dedicados)
1991	Sahinidis e Grossmann	Reformulação do modelo LRP através de subproblemas ELS
1991	Sahinidis e Grossmann	Planeamento de processos flexíveis
1994	Norton e Grossmann	Planeamento de processos flexíveis (Dow Chemical) com esquemas de produção
1995	Liu e Sahinidis	Algoritmo <i>branch-and-cut</i> , com planos de corte e métodos heurísticos
1998	Bok <i>et al.</i>	Planeamento de processos robusto para a indústria petroquímica coreana
2000	Ahmed e Sahinidis	Investigação analítica e probabilística de heurísticas
2000	Bok <i>et al.</i>	Planeamento de produção para a rede de processos flexíveis (Dow Chemical)

Himmelblau e Bickel [1980] combinaram um procedimento *branch-and-bound* com um esquema de Programação Não Linear (*Nonlinear Programming*, NLP) para planificar as expansões da capacidade para um processo de dessulfuração de *fuel*. Desenvolveram uma formulação detalhada do sistema de processos, configurando correntes de passagem (*bypass*) e de reciclo, modelando aspectos da implementação operacional das expansões, e propondo um algoritmo de cálculo numérico exigente em termos computacionais. De facto, é no sector energético e na

indústria petrolífera que se tem verificado maior aplicação dos métodos de MP a sistemas de larga escala, designadamente, quando se pretende a integração do planeamento ao longo da hierarquia de operações, visando uma resposta adequada ao nível global da empresa.

Jimenez e Rudd [1987] utilizaram um modelo recursivo de MILP para optimizar a expansão de capacidades para a indústria petroquímica mexicana, no contexto do planeamento dos investimentos para promover a substituição de importações por produção interna, obtendo-se um desenvolvimento integrado. Assim, o início da produção de determinados polímeros provoca a necessidade dos monómeros associados, bem como se sugere a implementação de unidades de compostos aromáticos e de outros polímeros, logo que as quantidades exigíveis destes materiais fundamentem a sua viabilidade económica.

Van Roy [1989] relata a implementação de um modelo de MILP direccionado a complexos problemas de produção e distribuição que ocorrem numa unidade de petroquímica e considerando uma abordagem multiestágio, buscando a minimização dos custos envolvidos. A problemática desenvolvida enquadra-se na localização de unidades de produção —para engarrafamento de gases, butano, propano—, e localização de unidades de armazenagem — sejam depósitos ou entrepostos. São também definidos os níveis óptimos de actividade nas refinarias e unidades de engarrafamento, os níveis de inventário nestas unidades e nos depósitos, a afectação de clientes às diferentes unidades, tratando dos fluxos de transporte e suas especificidades (agregação, tipologia, alocação de condutores).

Note-se que são relatados outros casos na indústria petroquímica, mas que não se dirigem ao planeamento das capacidades dos processos, mas antes ao planeamento da produção e distribuição, coordenando os fluxos materiais e financeiros desde os fornecedores aos consumidores. Nomeadamente, o trabalho de Klingman *et al.* [1986] constitui um bom exemplo das interacções que ocorrem nos complexos sistemas de produção e distribuição, pois consideram-se os fluxos materiais desde a aquisição das matérias, passando pelas unidades de refinação, até à entrega dos produtos aos clientes, e atende-se ao ambiente económico e à estratégia da empresa. A coordenação dos planos de produção e de distribuição é de importância crítica para uma empresa industrial com instalações em diferentes localizações. Klingman *et al.* [1987] descrevem o projecto de implementação de modelos de LP de grande dimensão, bem como a sua integração num sistema informático cujo plano mestre se baseava em optimização de redes (grafos). Este sistema de planeamento constituiu uma ferramenta útil para a avaliação de decisões relativas a distribuição, a orçamentação e definição de preços, a controlo de inventários, a acordos comerciais, e sustentou o desenvolvimento de políticas de vendas e um crescimento significativo de lucros. Neste âmbito, do planeamento de produção e da planificação dos fluxos da cadeia de fornecimentos, têm sido relatados outros casos da indústria petrolífera.

Num trabalho de continuidade realizado numa petrolífera brasileira: Moro *et al.* [1998] tratam o planeamento de produção de *diesel* através de NLP, aplicando-se o método do gradiente reduzido, o que permite substituir os cálculos manuais então utilizados; em sequência, Pinto *et al.* [2000] melhoram tal planeamento, abordando também o sequenciamento de curto prazo das operações de descarga de *crude*, de transferência em *pipelines*, de armazenagem e mistura, da afectação às unidades de destilação; Moro e Pinto [2004] aprofundam o tratamento das operações de curto prazo (4 dias) relativas ao *crude*, transformando o modelo original de Programação Não Linear Mista (*Mixed Integer Nonlinear Programming*, MINLP) em MILP;

Neiro e Pinto [2004] apresentam um enquadramento que visa modelar toda a cadeia de fornecimentos, conjugando os diversos planeamentos sobre as unidades de processo, armazenagem e *pipelines*, através de um modelo de MINLP de grande dimensão.

Este tipo de modelação, dirigido à cadeia de fornecimentos, visa integrar como ferramenta de cálculo um sistema de suporte à decisão (*Decision Support System*). Similares desenvolvimentos têm sido relatados, nomeadamente: Julka *et al.* [2002] apresentam um enquadramento para modelação, supervisão e gestão de toda a cadeia de fluxos, tanto dirigido internamente à rede de produção, como externamente ao fornecimento de *crude*; Lasschuit e Thijssen [2004] desenvolvem também um enquadramento para suporte à decisão numa petrolífera global, onde se visa a integração horizontal dos fluxos (desde os fornecedores aos consumidores), a integração vertical do planeamento (de médio e curto prazo), a integração modular dos componentes informáticos (promovendo o aumento de escala, o fluxo da informação entre instalações e o seu tratamento em tempo real).

Dada a complexidade inerente a problemas de grande dimensão, os trabalhos anteriormente referidos são, usualmente, de MINLP e de natureza determinística. No entanto, Gupta e Maranas [2003] tratam o efeito da incerteza das procuras dos produtos sobre o planeamento dos fluxos ao longo de toda a cadeia de fornecimentos, aplicando um 2SSP onde tratam em primeira fase as decisões de produção, integrando as decisões de distribuição em segunda fase. Então, em fase de recurso, são tratados os aspectos de logística, dos inventários, de abordagem aos mercados. Ainda, Oh e Karimi [2004] abordam, primeiro de uma forma determinística e depois estocástica, o planeamento das expansões de capacidades no âmbito da cadeia de fornecimentos global; no entanto, não prevêm o tratamento do risco nem a promoção da robustez, dada a complexidade da tarefa.

Tal complexidade é reconhecida por Papageorgiou *et al.* [2001], quando abordam a cadeia de fornecimentos na indústria farmacêutica, integrando num modelo determinístico de MILP o desenvolvimento de novos produtos, o planeamento da produção dos ingredientes activos, bem como os investimentos em novas unidades de produção. A situação de desenvolvimento de novos produtos farmacêuticos é tratada de forma probabilística por Blau *et al.* [2000], para o que se aplicam procedimentos heurísticos e de simulação no âmbito dum planeamento de projecto, no intuito de se diminuir a exposição ao risco. Têm sido relatados variados trabalhos na área da indústria farmacêutica, tendo Shah [2004] revisto os aspectos críticos que afectam estas cadeias de fornecimento, como por exemplo: a implementação de novas unidades (para produção e distribuição), quer no aspecto de localização ou de expansão de capacidades; o planeamento da produção, quer envolvendo os longos ciclos de produção dos ingredientes activos e dos produtos secundários, quer o sequenciamento de curto prazo; o desenvolvimento de novos produtos e os inventários existentes; a integração dos fluxos de fornecimento e distribuição. O mesmo autor [Shah, 2005], expande a sua revisão às cadeias de fornecimento das indústrias de processo, apontando desenvolvimentos e desafios que permanecem, que supõem a combinação de aspectos de: implementação da rede logística; planeamento de produção, em diversos horizontes; gestão e controlo dos fluxos, em tempo real; longo prazo, como a relação com fornecedores e a evolução da própria cadeia industrial.

Na optimização da cadeia de fornecimentos e duma rede de processos químicos, Bok *et al.* [2000] aplicam um enquadramento determinístico que recorre à

modelação de processos flexíveis [Norton e Grossmann, 1994]: o pormenorizado (considera aspectos de distribuição, inventários, intermitência nos fornecimentos, transições de produção e suas quebras) problema de MILP é decomposto num subproblema que origina a *lower bound*, considerando-se também o problema relaxado que fornece a *upper bound*. Na mesma empresa industrial, Jackson e Grossmann [2003] também aplicam um enquadramento determinístico para tratar da rede *multisite*, considerando os fluxos com os fornecedores e os mercados finais; o problema NLP de grande dimensão é abordado por decomposição lagrangeana, em duas vertentes: espacial, discriminando as redes de produção e distribuição; temporal, optimizando as redes em função das condições de mercado, segundo cada período de tempo.

A dificuldade destes problemas é evidente, tendo Bose e Pekny [2000] abordado uma cadeia de fornecimento de bens de consumo através de um modelo de controlo preditivo: em alternativa aos esquemas de decomposição, preferem uma abordagem de simulação onde se incorpora a incerteza. Através de um modelo de previsão, estabelecem-se as necessidades de inventário, que são optimizadas de acordo com os fluxos produzidos, sendo considerada a dinâmica da cadeia e a sensibilidade aos parâmetros de controlo, como seja o serviço ao consumidor. Outra aplicação alternativa é a de D'Alessandro e Baveja [2000] que, no âmbito duma cadeia de produção e distribuição de polímeros e resinas, procederam a um planeamento direccionado, quer para a satisfação de encomendas, quer para inventário, bem como desenvolveram canais de distribuição e segmentaram o mercado, obtendo assim ganhos significativos.

Assim, constata-se a importância específica de certos factores, como o financeiro. Suh e Lee [2001] consideram um enquadramento economicamente robusto para planeamento de processos, onde promovem uma medida probabilística do desempenho e a robustez; o pior cenário possível conduz à solução robusta de referência, é desenvolvida uma análise Pareto para definir um conjunto de soluções robustas, entre as quais se selecciona a melhor através de um procedimento específico. Roderer *et al.* [2002] aprofundam um modelo estocástico [Liu e Sahinidis, 1997] para tratamento do risco económico do investimento na indústria de processos, utilizam uma formulação multicritério, e aplicam uma decomposição multiparamétrica de forma a definir, não só a melhor solução, mas também a sua vizinhança onde o decisor se possa basear. Barbaro e Bagajewicz [2004a], também no âmbito da optimização robusta do planeamento de processos, abordam o risco financeiro através de diferentes estimadores probabilísticos e ilustram numericamente a sua implementação computacional. Estes autores [Barbaro e Bagajewicz, 2004b] prosseguem a sua investigação, aprofundando o tratamento do risco financeiro através da utilização de inventários e de instrumentos financeiros, que possibilita o recurso ao mercado bolsista de Derivados (Futuros, Opções) para manipular as curvas de risco.

Além do factor financeiro, também o factor ambiental tem sido abordado, em termos de fecho da cadeia de fornecimentos ou de reciclagem de materiais. Realff *et al.* [2000] promovem uma optimização robusta do planeamento do sistema de produção reversivo, isto é, considerando a reciclagem dos materiais e a sua reutilização como matéria prima; a abordagem é ilustrada com a aplicação à reutilização de materiais sintéticos (de tapetes e alcatifas), o que constituía à época um aspecto crítico nos Estados Unidos. Também Hugo e Pistikopoulos [2005] consideram o fecho ou reversão de materiais no âmbito da cadeia de fornecimentos, pois através de um modelo multicritério e determinístico que considera a localização

de unidades, a expansão de capacidades, também avalia e pondera o ciclo de vida dos produtos (*life cycle assessment*).

### 2.1.2 Problemas de planeamento de processos dedicados

Bok *et al.* [1998] apresentam um caso de optimização, ilustrativo do plano de investimentos para expansão da indústria petroquímica coreana, temporizando e dimensionando as expansões de capacidade em conformidade com os cenários de procura. A desregulação e o crescimento rápido dos mercados de produtos químicos, coreano, chinês e do sudoeste asiático, ao longo dessa década, provocaram um súbito aumento de procura de produtos petrolíferos:

- A procura interna do mercado coreano encontra-se relacionada com a das outras indústrias dessa fileira industrial, nomeadamente, com a indústria de especialidades químicas, cujos produtos derivam para as indústrias electrónicas e do automóvel, de tecidos e vestuário, embalagens e alimentação;
- convém antecipar a procura explosiva do mercado chinês, pois cerca de 50% da produção coreana é dirigida para a China, pese embora os problemas de natureza burocrática usualmente relatados;
- os países do sudoeste asiático apresentam taxas de crescimento de procura dos produtos petroquímicos de cerca de 9%, mas têm efectuado também investimentos na indústria petrolífera doméstica.

Dada a complexidade dos aspectos equacionados e embora seja previsível uma tendência crescente para a procura, o cenário de saturação de procura não é excluído, nem sequer se coloca de lado a hipótese de decrescimento da procura. Consideraram-se taxas de crescimento diferenciadas para cada gama de produtos (olefinas, aromáticos), em função dos cenários traçados para a evolução da procura nos mercados finais de produtos petrolíferos, de resinas sintéticas, de termoplásticos e outros derivados. Aproveitou-se alguma experiência retirada de um estudo anterior, realizado no âmbito do planeamento dos investimentos na indústria petroquímica e de refinação da Coreia do Sul, que balanceava múltiplas localizações de unidades e fluxos de transporte respeitantes, impondo constrangimentos tanto a importações como a exportações de materiais.

Considerem-se os seguintes parâmetros para formulação do problema de planeamento de processos, e utilizados no âmbito de SP por discretização de cenários:

$NC, NR$ , números de componentes ou materiais  $j$ , de cenários  
 $NT, NP, NM$  – previsíveis ou aleatórios (*random*)  $r$ , de períodos de tempo  
 $t$ , de processos de produção  $i$ , de mercados dos materiais  
químicos  $l$ ;

$prob_r$  – probabilidade de ocorrência do cenário aleatório, ou do  
“ponto” de amostragem aleatória  $r$ ;

$\xi$  – valor líquido actual (VLA);

$\lambda, w$  – parâmetros de penalidades;

$Z_p$  – erros ou desvios de impossibilidade, em cada cenário;

- $u$  – multiplicadores de Lagrange, associados à relaxação de restrições;
- $\gamma, \Gamma$  – preços unitários actualizados, de venda e de compra, dos produtos químicos;
- $\delta$  – custos unitários actualizados de produção, relativos ao componente principal de cada processo;
- $\alpha, \beta$  – custos actualizados para expansão de capacidades, de natureza variável e fixa;
- $a, d$  – disponibilidades (*availabilities*) e procuras (*demands*) dos produtos químicos;
- $\mu, \eta$  – constantes positivas características, associando os fluxos de entrada  $WI$  e de saída  $WO$  de cada componente, a um fluxo genérico  $W$  do processo.

A função objectivo do modelo de optimização robusta pretende maximizar a esperança da métrica  $\xi$  (o VLA), a menos duma parcela associada à respectiva variância (quadrado dos desvios ao valor esperado em cada cenário  $r$ ), bem como se introduz uma penalidade  $w$  pelos desvios  $Zp$  inerentes à modelação efectuada (por exemplo, excesso de produção ou de capacidade de produção a implementar):

$$\text{p.2} \quad [\max] \Phi = \sum_{r=1}^{NR} \text{prob}_r \xi_r - \lambda \sum_{r=1}^{NR} \text{prob}_r \left( \xi_r - \sum_{r'=1}^{NR} \text{prob}_{r'} \xi_{r'} \right)^2 - \omega \sum_{r=1}^{NR} \sum_{t=1}^{NT} \sum_{i=1}^{NP} (Zp_{rit})^2$$

A função objectivo determinística considera como métrica o VLA,  $\xi$ , definido através da mais valia realizada entre as quantidades compradas  $Pur$  e as quantidades vendidas  $Sal$  de materiais químicos, descontada dos custos associados aos fluxos de produção  $W$ . Consideram-se também os custos devidos à expansão de capacidade dos processos: de natureza variável, sendo os custos proporcionais às quantidades  $QE$ ; de natureza fixa, se associados à variável binária  $y$ . Apresentando os custos unitários respeitantes através de letras do alfabeto grego, o VLA define-se:

$$\text{p.3} \quad \xi_r = \sum_{j=1}^{NC} \sum_{t=1}^{NM} \sum_{t=1}^{NT} (\gamma_{jlt} Sal_{rjlt} - \Gamma_{jlt} Pur_{rjlt}) - \sum_{i=1}^{NP} \sum_{t=1}^{NT} \delta_{it} W_{sit} - \sum_{i=1}^{NP} \sum_{t=1}^{NT} (\alpha_{it} QE_{it} + \beta_{it} y_{it}), \quad \forall r$$

A gama de valores para expansão de capacidades de produção e de armazenagem encontra-se delimitada logicamente, pela inclusão das variáveis binárias  $y$ , entre valores mínimo ( $QE$  lower) e máximo ( $QE$  upper):

$$\text{p.4} \quad QE_{it}^{Low} y_{it} \leq QE_{it} \leq QE_{it}^{Upp} y_{it}, \quad \forall i, t$$

Em cada período  $t$  e para cada processo  $i$ , é necessário proceder a balanços à capacidade de produção instalada,

$$\text{p.5} \quad Q_{it} = Q_{i,t-1} + QE_{it}, \quad \forall i, t$$

pois assim poder-se-á delimitar os níveis de produção  $W$  e determinar os desvios (folgas)  $Zp$  associados à subutilização da capacidade instalada, para cada cenário  $r$ :

$$p.6 \quad W_{rit} + Zp_{rit} = Q_{it} \quad , \quad \forall r, i, t$$

São definidos valores de limite superior para as disponibilidades de matérias,  $a$ , e para as procuras de produtos,  $d$  (para promoção da resolução numérica possível do programa estocástico, relaxam-se os valores mínimos das gamas de variação destas quantidades para o valor zero):

$$p.7 \quad Pur_{rjlt} \leq a_{rjlt}^{Upp} \quad , \quad \forall r, j, l, t$$

$$p.8 \quad Sal_{rjlt} \leq d_{rjlt}^{Upp} \quad , \quad \forall r, j, l, t$$

Para cada espécie química e ao longo da rede de processos, estabelecem-se equações de balanços materiais, quer atendendo ao principal fluxo de alimentação dos processos,  $WI$ , ao principal fluxo efluente dos processos,  $WO$ , ou mesmo definindo balanços materiais globais:

$$p.9 \quad WI_{rijt} \leq \mu_{it} W_{rit} \quad , \quad \forall r, i, j, t$$

$$p.10 \quad WO_{rijt} \leq \eta_{it} W_{rit} \quad , \quad \forall r, i, j, t$$

$$p.11 \quad \sum_{l=1}^{NM} Sal_{rjlt} + \sum_{i=1}^{NP} WI_{rijt} = \sum_{l=1}^{NM} Pur_{rjlt} + \sum_{i=1}^{NP} WO_{rijt} \quad , \quad \forall r, j, t$$

Da conjugação das relações descritas,  $p.2$  a  $p.11$  com as condições de não-negatividade e de integralidade das variáveis, obtém-se um modelo não linear devido aos termos quadráticos da função objectivo estocástica, onde ainda ocorrem variáveis binárias para modelação das decisões de expansão de capacidades: encara-se um problema de MINLP, de difícil resolução. Saliente-se, não só o elevado número de variáveis e restrições associado ao cálculo determinístico do VLA, mas também o numeroso grupo de amostras que seria necessário avaliar para obtenção de uma solução fiável, apresentando uma variância possível.

Todavia, Bok *et al.* consideraram o processo alternativo de cálculo da medida probabilística através de um esquema de 2SSP. Por conseguinte, neste problema da expansão de capacidades numa rede de processos:

- em primeira fase, a capacidade de cada processo individual, em cada período do horizonte temporal, pertence ao conjunto das variáveis de projecto, pois os valores disponíveis para selecção não se encontram dependentes da incerteza associada às procuras de cada produto; também para cada período, as variáveis associadas às vendas desses produtos, aos fluxos materiais ao longo da rede de produção, às compras de materiais, constituem as variáveis de controlo;
- em segunda fase, é introduzida a incerteza através da definição de um conjunto de cenários, e da probabilidade associada; as previsões de procura de materiais e dos respectivos preços, ao longo do horizonte de planeamento, permitem construir os cenários que influenciam os contornos da situação.

Para obviar às dificuldades computacionais inerentes aos termos não lineares da função objectivo estocástica, Bok *et al.* aplicam um algoritmo de decomposição, desenvolvido inicialmente por Liu e Sahinidis [1996] e adaptado ao problema de

planeamento de processos com robustez. No problema mestre da primeira fase, esse algoritmo considera as variáveis referentes às expansões de capacidades ( $y$ ,  $QE$ ,  $Q$ ), às quais se associam as variáveis de projecto do programa estocástico em cada iteração  $k$ .

**Modelo *plan\_minlp\_2ssp\_mest*:**

**p.12-a**  $[max] \alpha$

sujeito a

**p.12-b** 
$$\alpha \leq \sum_{r=1}^{NR} prob_r \xi_r - \lambda \sum_{r=1}^{NR} prob_r \left( \xi_r - \sum_{r=1}^{NR} prob_r \xi_r \right)^2$$

$$- \omega \sum_{r=1}^{NR} \sum_{t=1}^{NT} \sum_{i=1}^{NP} (Zp_{rit})^2 + \sum_{r=1}^{NR} \sum_{t=1}^{NT} \sum_{i=1}^{NP} u_{rit}^k (W_{rit}^k - Q_{it})$$

**p.12-c** 
$$\xi_r + \sum_{i=1}^{NP} \sum_{t=1}^{NT} (\alpha_{it} QE_{it} + \beta_{it} y_{it}) = \sum_{j=1}^{NC} \sum_{l=1}^{NM} \sum_{t=1}^{NT} (\gamma_{jlt} Sal_{jlt}^k - \Gamma_{jlt} Pur_{jlt}^k)$$

$$- \sum_{i=1}^{NP} \sum_{t=1}^{NT} \delta_{it} W_{rit}^k, \quad \forall r$$

**p.12-d**  $Q_{it-1} + QE_{it} - Q_{it} = 0, \quad \forall i, t$

**p.12-e**  $QE_{it}^{Low} y_{it} - QE_{it} \leq 0, \quad \forall i, t$

**p.12-f**  $QE_{it} - QE_{it}^{Upp} y_{it} \leq 0, \quad \forall i, t$

**p.12-g, h**  $\xi_r, Zp_{rit}, QE_{it}, Q_{it} \geq 0, \quad \forall i, t; \quad y_{it} \in \{0,1\}, \quad \forall i, t$

O modelo *plan\_minlp\_2ssp\_mest* apresenta apenas:  $(NR + NR.NP.NT + 2 NP.NT)$  variáveis contínuas, correspondentes às capacidades dos processos e respectivas expansões de capacidade;  $(NP.NT)$  variáveis discretas, respeitantes às decisões binárias quanto às expansões em cada período de tempo;  $(3 NP.NT + 2)$  restrições, respectivamente, associadas aos balanços de capacidades e à dupla de restrições lógicas sobre as quantidades a expandir, bem como à definição dos objectivos determinístico e estocástico a otimizar (respectivamente, a função determinística da métrica do VLA, a função estocástica sob a forma constringida de Benders).

Encontrando-se assim definidas as decisões quanto à expansão de capacidades ( $y$ ,  $QE$ ,  $Q$ ) e configurada a rede de processos químicos, considera-se então —em cada iteração  $k$  da segunda fase— o subproblema de planeamento operacional com as variáveis de controlo ( $Sal$ ,  $Pur$ ,  $W$ ):

**Modelo *plan\_nlp\_2ssp\_sub*:**

**p.13-a**  $[max] \Phi = \sum_{r=1}^{NR} prob_r \xi_r - \lambda \sum_{r=1}^{NR} prob_r \left( \xi_r - \sum_{r=1}^{NR} prob_r \xi_r \right)^2 - \omega \sum_{r=1}^{NR} \sum_{t=1}^{NT} \sum_{i=1}^{NP} (Zp_{rit})^2$

sujeito a

$$\text{p.13-b} \quad \xi_r - \sum_{j=1}^{NC} \sum_{l=1}^{NM} \sum_{t=1}^{NT} (\gamma_{jlt} Sal_{rjlt} - \Gamma_{jlt} Pur_{rjlt}) + \sum_{i=1}^{NP} \sum_{t=1}^{NT} \delta_{it} W_{rit} = - \sum_{i=1}^{NP} \sum_{t=1}^{NT} (\alpha_{it} QE_{it}^k + \beta_{it} y_{it}^k) , \quad \forall r$$

$$\text{p.13-c} \quad W_{rit} + Zp_{rit} = Q_{it}^k , \quad \forall r, i, t$$

$$\text{p.13-d} \quad WI_{rijt} - \mu_{it} W_{rit} \leq 0 , \quad \forall r, i, j, t$$

$$\text{p.13-e} \quad WO_{rijt} - \eta_{it} W_{rit} \leq 0 , \quad \forall r, i, j, t$$

$$\text{p.13-f} \quad \sum_{l=1}^{NM} Sal_{rjlt} + \sum_{i=1}^{NP} WI_{rijt} - \sum_{l=1}^{NM} Pur_{rjlt} - \sum_{i=1}^{NP} WO_{rijt} = 0 , \quad \forall r, j, t$$

$$\text{p.13-g} \quad Pur_{rjlt} \leq a_{rjlt}^{Upp} , \quad \forall r, j, l, t$$

$$\text{p.13-h} \quad Sal_{rjlt} \leq d_{rjlt}^{Upp} , \quad \forall r, j, l, t$$

$$\text{p.13-i} \quad \xi_r, Sal_{rjlt}, Pur_{rjlt}, W_{rit}, WI_{rijt}, WO_{rijt}, Zp_{rit} \geq 0 , \quad \forall r, i, j, l, t$$

Este modelo *plan\_nlp\_2ssp\_sub* apresenta:  $(NR + 2 NR.NC.NM.NT + 2 NR.NP.NT + 2 NR.NP.NC.NT)$  variáveis contínuas, correspondentes a quantidades vendidas e compradas, aos níveis de fluxo dos processos, aos desvios aceitáveis pela formulação; note-se que as variáveis binárias do problema original, respeitantes às decisões de expansão, e as capacidades dos processos são tratadas como parâmetros neste subproblema;  $(NR.NP.NT + 2 NR.NP.NC.NT + NC.NT)$  restrições, associadas às restrições para definição dos desvios, dos fluxos de cada componente em cada processo, dos balanços globais aos componentes, entendendo-se que os constrangimentos às quantidades de compra e de venda, *p.13-g* e *p.13-h*, são tratados como valores limites daquelas variáveis.

Ora, o modelo determinístico —de Sahinidis *et al.* [1989]— seleccionado como suporte do problema estocástico de planeamento de processos, para além do conjunto de características comuns que apresenta relativamente aos problemas de sequenciamento de operações e de síntese de sistemas de processos, é correntemente utilizado como referência na experimentação de propostas pertinentes para investigação. Tal deve-se ao carácter combinatório que o modelo revela, com a conjugação simultânea de variáveis binárias e contínuas, à flexibilidade da formulação do problema em termos lineares ou não-lineares, o que se traduz em problemas de PM com elevado grau de dificuldade e onde interessa testar novas técnicas de resolução. Embora o modelo original tenha sofrido sucessivos aperfeiçoamentos —que se apresentam seguidamente—, a essência ou objectivo da planificação óptima do sistema de processos tem-se mantido: calendarização das expansões e dimensionamento de cada processo; determinação dos níveis produtivos dos processos; definição das quantidades compradas e vendidas dos materiais químicos.

Sahinidis *et al.* [1989] desenvolveram um modelo de MILP para planeamento de longo prazo (*long range planning*, LRP) na indústria química, utilizando uma

estrutura de períodos temporais múltiplos, modelo esse que se apresenta em seguida com similar nomenclatura à do descrito modelo com robustez, mas considerando também os parâmetros:

$NEXP(i)$  – número admissível máximo de expansões, para cada processo  $i$ , ao longo de todo o horizonte temporal;

$CI(t)$  – valor máximo para o investimento em expansão de capacidades, para cada período  $t$ .

Todavia, considera-se aqui que os níveis de produção  $W$  para cada processo se encontram constrangidos pelo valor da capacidade instalada respeitante:

$$\text{p.14} \quad Q_{it} \geq W_{m,t} \quad , \quad \forall i, t$$

Para cada espécie química e ao longo da rede de processos, estabelecem-se equações de balanços materiais supondo a existência de uma relação linear do fluxo do componente para com o fluxo principal de cada processo:

$$\text{p.15} \quad W_{kt} \leq \mu_{ik} W_{m,t} \quad , \quad \forall k, i, t$$

São definidas gamas de variação para as disponibilidades de matérias,  $a$ , e para as procuras dos produtos,  $d$ , especificando ambos os valores para limite superior e inferior:

$$\text{p.16} \quad a_{jlt}^{Low} \leq Pur_{jlt} \leq a_{jlt}^{Upp} \quad , \quad \forall j, l, t$$

$$\text{p.17} \quad d_{jlt}^{Low} \leq Sal_{jlt} \leq d_{jlt}^{Upp} \quad , \quad \forall j, l, t$$

Obtém-se o modelo determinístico *plan\_milp\_lrp*, para planeamento de longo prazo e que utiliza uma formulação de MILP, com variáveis contínuas e inteiras (binárias).

**Modelo *plan\_milp\_lrp*:**

$$\text{p.18-a} \quad \begin{aligned} [max]_{\xi} = & \sum_{j=1}^{NC} \sum_{l=1}^{NM} \sum_{t=1}^{NT} (\gamma_{jlt} Sal_{jlt} - \Gamma_{jlt} Pur_{jlt}) \\ & - \sum_{i=1}^{NP} \sum_{t=1}^{NT} \delta_{it} W_{m,t} - \sum_{i=1}^{NP} \sum_{t=1}^{NT} (\alpha_{it} QE_{it} + \beta_{it} y_{it}) \end{aligned}$$

sujeito a

$$\text{p.18-b} \quad Q_{it-1} + QE_{it} - Q_{it} = 0 \quad , \quad \forall i, t$$

$$\text{p.18-c} \quad QE_{it}^{Low} y_{it} - QE_{it} \leq 0 \quad , \quad \forall i, t$$

$$\text{p.18-d} \quad QE_{it} - QE_{it}^{Upp} y_{it} \leq 0 \quad , \quad \forall i, t$$

$$\text{p.18-e} \quad Q_{it} - W_{m,t} \geq 0 \quad , \quad \forall i, t$$

$$\text{p.18-f} \quad W_{kt} - \mu_{ik} W_{m,t} \leq 0 \quad , \quad \forall k, i, t$$

$$\text{p.18-g} \quad \sum_{l=1}^{NM} Sal_{jlt} + \sum_{k \in I(j)} W_{kt} - \sum_{l=1}^{NM} Pur_{jlt} - \sum_{k \in O(j)} W_{kt} = 0, \quad \forall j, t$$

$$\text{p.18-h} \quad a_{jlt}^{Low} \leq Pur_{jlt} \leq a_{jlt}^{Upp}, \quad \forall j, l, t$$

$$\text{p.18-i} \quad d_{jlt}^{Low} \leq Sal_{jlt} \leq d_{jlt}^{Upp}, \quad \forall j, l, t$$

$$\text{p.18-j, k} \quad Sal_{jlt}, Pur_{jlt}, W_{it}, QE_{it}, Q_{it} \geq 0; \quad y_{it} \in \{0,1\}, \quad \forall i, j, l, t$$

Este modelo *plan\_milp\_lrp* apresenta: (4 *NP.NT* + 2 *NM.NC.NT*) variáveis, sendo (*NP.NT*) as variáveis discretas correspondentes às decisões binárias (0/1) sobre expansão de capacidades de cada processo em todos os períodos de tempo, e as restantes variáveis, dos fluxos de produção, das quantidades expandidas e das capacidades dos processos, das quantidades compradas e vendidas nos diversos mercados, são contínuas; (*NC.NT* + 5 *NP.NT*) constrangimentos, associados às (*NC.NT*) equações de balanço global, para cada componente e período temporal, bem como (*NP.NT*) relações para cada um dos cinco tipo de restrições — respectivamente, equações de balanço às capacidades, a dupla de restrições lógicas da gama de expansão de capacidades, a capacidade instalada como limite superior do fluxo em cada processo, o fluxo específico de cada componente. É aqui implícito que as quantidades das compras e vendas de produtos são incorporadas como variáveis limitadas, não existindo assim a necessidade de considerar novas restrições.

Eventualmente, podem ser adicionadas outras restrições, umas limitando o número de expansões que os processos possam sofrer,

$$\text{p.19} \quad \sum_{t=1}^{NT} y_{it} \leq NEXP(i), \quad \forall i$$

enquanto outras constroem o capital disponível para investimento, em alguns dos períodos de tempo (aqui, consideram-se coeficientes não actualizados dos custos):

$$\text{p.20} \quad \sum_{i=1}^{NP} (\overline{\alpha}_{it} QE_{it} + \overline{\beta}_{it} y_{it}) \leq CI(t), \quad \forall t$$

Sahinidis *et al.* consideraram que o modelo LRP (aqui designado por *plan\_milp\_lrp*) pode também assumir explicitamente: a paragem temporária de unidades de produção, mas suportando os respectivos encargos de manutenção; o funcionamento dum processo com nível ou caudal mínimo; o fecho definitivo da unidade de produção.

São investigadas várias estratégias de resolução — como o *branch-and-bound*, a aplicação de diversos procedimentos heurísticos e de planos de corte, o método de decomposição de Benders —, para determinar os momentos óptimos de instalar e expandir diversos processos químicos, numa complexidade crescente até à exemplificação com um complexo petroquímico. Verifica-se que a introdução de planos de corte e a aplicação de regras heurísticas constituíram as estratégias mais eficientes para complementar o *branch-and-bound*, revelando-se como pouco promissora a metodologia de decomposição efectuada, pois os limites produzidos pelo problema mestre eram bastante alargados e a exigência de resolução de subproblemas foi elevada.

### 2.1.3 O planeamento de processos flexíveis

Com o intuito de flexibilizar o processo de produção, requerida quando diferentes produtos de alto valor acrescentado são produzidos em pequena quantidade numa rede interligada de processos químicos, Sahinidis e Grossmann [1991b] generalizam o anterior modelo, para programação multiperíodo dos investimentos em processos contínuos dedicados [Sahinidis *et al.*, 1989], permitindo a consideração de operações descontínuas em ambiente flexível de produção.

Introduzem-se os seguintes parâmetros, que permitem formular os balanços mássicos e definir as exigências de capacidades produtivas:

$\rho_{ij}$  – taxa de produção adimensional relativa, definida a partir da taxa de produção e do factor de capacidade  $Q$ , referentes ao produto principal  $j$  no esquema de produção utilizado pelo processo  $i$ ;

$\theta_{ijt}$  – quantidade produzida do produto principal  $j$ , em produção no processo  $i$  e no período  $t$ ;

$H_{ij}$  – valor máximo do tempo de produção, disponível pelo processo  $i$  ao longo do período  $t$ ;

$M_i$  – conjunto referente ao produto principal  $j$ , conforme o esquema de produção no processo  $i$ .

Ora, o caso da produção contínua numa unidade dedicada pode ser considerado como um caso específico no enquadramento de produção contínua flexível, através de uma representação matematicamente unificada que recorre ao valor atribuível ao parâmetro  $\rho$ :

$\rho_{ij} = 1$  – aplicável ao processo contínuo  $i$ , se dedicado à produção do componente  $j$ ;

$\rho_{ij} = \text{const} (\neq 1)$  – aplicável ao processo contínuo  $i$ , em produção flexível;

$\rho_{ij} = 1/(\sigma_{ij}\tau_{ij})$  – aplicável ao processo descontínuo  $i$ , em produção dedicada ou flexível do componente  $j$ .

No caso de produção descontínua, a taxa de produção relativa não se apresenta como um parâmetro adimensional, dependendo inversamente do tempo de produção *batch*  $\tau$  (horas) e do factor de dimensão  $\sigma$  (litros/kg). Nesta situação, deve também ser cuidada a especificação do valor para o factor de capacidade  $Q$  (litros), de forma a permitir a fiabilidade da mesma formulação matemática para descrever os diferentes tipos de processos, tanto de produção contínua como descontínua.

A quantidade máxima produzida pelo processo  $i$ , dos produtos principais  $j$  no período  $t$ , depende do tempo máximo  $H$  alocável ao esquema de produção do componente  $j$  e do factor de capacidade do processo  $i$ :

$$\text{p.21} \quad \sum_{j \in M_i} \theta_{ijt} \leq H_{ij} Q_{it}, \quad \forall i, j, t$$

A quantidade produzida do produto principal  $j$ , no processo  $i$  e no período  $t$ , é definida linearmente utilizando a taxa de produção relativa (adimensional para processos contínuos):

p.22 
$$W_{ijt} = \rho_{ij}\theta_{ijt} \quad , \quad \forall i, j, t$$

Designando aqui por  $j'$  o produto principal do esquema alocado ao processo  $i$  no período  $t$ , a quantidade produzida de cada subproduto ou produto secundário  $j$  nesse processo  $i$  é determinada utilizando os índices  $\mu$  característicos do processo:

p.23 
$$W_{ijt} = \sum_{j' \in M_i} \mu_{ijj'} W_{ij't} \quad , \quad \forall i, j, t$$

Introduzindo estas (3 NP.NC.NT) restrições, que possibilitam um tratamento similar para os processos contínuos e descontínuos, obtém-se o modelo *plan\_milp\_lrp\_flex* para planeamento flexível de produção em horizonte de longo prazo, mas onde deve atender-se às diferenças na definição das taxas de produção relativas  $\rho$  e dos factores de capacidade  $Q$ , para se obter devidamente as (NP.NC.NT) quantidades adicionais associadas a cada esquema de produção,  $\theta_{ijt}$ .

**Modelo *plan\_milp\_lrp\_flex*:**

p.24-a 
$$[max]_{\xi} = \sum_{j=1}^{NC} \sum_{l=1}^{NM} \sum_{t=1}^{NT} (\gamma_{jlt} Sal_{jlt} - \Gamma_{jlt} Pur_{jlt})$$

$$- \sum_{i=1}^{NP} \sum_{j \in M_i} \sum_{t=1}^{NT} \delta_{ijt} W_{ijt} - \sum_{i=1}^{NP} \sum_{t=1}^{NT} (\alpha_{it} QE_{it} + \beta_{it} y_{it})$$

sujeito a

Relações p.18-b a p.18-d

Relações p.18-h a p.18-i

Relações p.19 e p.20

p.24-b 
$$\sum_{l=1}^{NM} Sal_{jlt} + \sum_{k \in I(j)} W_{ijkt} - \sum_{l=1}^{NM} Pur_{jlt} - \sum_{k \in O(j)} W_{ijkt} = 0 \quad , \quad \forall j, t$$

p.24-c 
$$\sum_{j \in M_i} \theta_{ijt} - H_{ij} Q_{it} \leq 0 \quad , \quad \forall i, j, t$$

p.24-d 
$$W_{ijt} - \rho_{ij}\theta_{ijt} = 0 \quad , \quad \forall i, j, t$$

p.24-e 
$$W_{ijt} - \sum_{j' \in M_i} \mu_{ijj'} W_{ij't} = 0 \quad , \quad \forall i, j, t$$

p.24-f, g 
$$Sal_{jlt}, Pur_{jlt}, W_{ijt}, QE_{it}, Q_{it}, \theta_{ijt} \geq 0 \quad ; \quad y_{it} \in \{0,1\} \quad , \quad \forall i, j, l, t$$

A principal vantagem desta modelação revela-se através da conformação do sistema de produção aos requisitos do mercado de produtos finais, através da flexibilização dos processos utilizados: a diminuição da procura de um produto, que poderia levar à subutilização ou mesmo à paragem de produção do processo

dedicado, poderá assim disponibilizar capacidade para satisfazer a procura crescente de outros produtos.

Contudo, subsistem dificuldades na formulação descrita, nomeadamente, quanto à flexibilidade nos esquemas de produção, de maneira a equacionar a obtenção de um mesmo produto a partir de materiais diferentes, ou mesmo para diferenciar esses esquemas, caso seja exigível perante a transformação de um mesmo material mas oriundo de fonte diversa. Em conformidade com o modelo anterior, cada diferente esquema de produção deverá de ser formulado como uma nova unidade de processo, provocando o crescimento do número de variáveis, mas também se terão de adicionar restrições respeitantes à delimitação global das capacidades processuais.

Este conceito da flexibilização da rede de processos será expandido por Norton e Grossmann [1994] na concretização de um estudo elaborado para a Dow Chemical, que não só ponderam a utilização de diferentes esquemas de produção para um dado processo, como também consideram a flexibilidade de utilização de diferentes tipos de matéria-prima na produção dos diversos produtos finais, perspectivando-se ainda as múltiplas combinações de esquemas de produção entre as matérias e os produtos.

Para tal, individualizam-se os esquemas de produção disponíveis para cada transformação, pelo que as quantidades produzidas nos processos terão de observar a agregação dos referidos esquemas. Sejam:

$NS$  – número dos esquemas de produção  $s$  alternativos para os processo flexíveis;

$PS(i)$  – conjunto dos esquemas de produção  $s$ , alocáveis ao processo flexível  $i$ ;

$JM(i,s)$  – conjunto referente a cada produto principal  $j$ , segundo o esquema de produção  $s$  no processo  $i$ .

A quantidade máxima de produção (agregada) do processo  $i$ , englobando cada um dos esquemas  $s$  de destinados à produção dos produtos principais  $j$  no período  $t$ , depende do tempo máximo alocável  $H$  ao esquema de produção do componente  $j$  e do factor de capacidade do processo  $i$ :

$$\text{p.25} \quad \sum_{j \in JM(i,s)} \sum_{s \in PS(i)} \theta_{ijst} \leq H_{ij} Q_{it}, \quad \forall i, j, t$$

Para o produto principal  $j$ , a agregação das quantidades produzidas pela totalidade dos esquemas  $s$ , alocados ao processo  $i$  no período  $t$ , é obtida linearmente através da taxa de produção relativa:

$$\text{p.26} \quad W_{ijt} = \sum_{s \in PS(i)} \rho_{ijs} \theta_{ijst}, \quad \forall i, j, t$$

A quantidade agregada de cada produto secundário  $j$ , em cada processo  $i$  no período  $t$ , considera a soma das contribuições de todos os esquemas  $s$  alocados; cada contribuição particular é obtida utilizando os índices característicos do processo, que relacionam de forma linear as quantidades dos produtos secundários com a quantidade referente ao produto principal  $j'$  do esquema de produção:

p.27 
$$W_{ijt} = \sum_{s \in PS(i)} \mu_{ijs} W_{ij'st} \quad , \quad \forall i, j, t$$

O modelo com aperfeiçoamento de flexibilidade por Norton e Grossmann [1994], aqui denominado *plan\_milp\_lrp\_schem*, consiste num modelo MILP para investimentos num horizonte de longo prazo em redes de processos químicos, que considera unidades de produção dedicadas ou flexíveis, contínuas ou descontínuas. Este modelo trata a implementação de novos processos e a produção de novos produtos, equaciona de forma flexível a utilização de diferentes materiais provenientes de diversas fontes ou assume mesmo a evolução temporal de uma rede de processos preexistente. Mantém-se o objectivo de maximizar o VLA ao longo de um horizonte de planeamento, definindo-se os valores óptimos para:

- a política de expansões, quantificando e calendarizando a adição de capacidades aos processos, novos ou já existentes;
- níveis de operação dos processos, em cada período de tempo;
- compras de materiais e vendas de produtos, ao longo do horizonte de planeamento.

**Modelo *plan\_milp\_lrp\_schem*:**

p.28-a 
$$[max] \xi = \sum_{j=1}^{NC} \sum_{l=1}^{NM} \sum_{t=1}^{NT} (\gamma_{jlt} Sal_{jlt} - \Gamma_{jlt} Pur_{jlt}) - \sum_{i=1}^{NP} \sum_{s \in PS(i)} \sum_{t=1}^{NT} \left( \delta_{ijt} \sum_{j \in JM(i,s)} \rho_{ist} \theta_{ijst} \right) - \sum_{i=1}^{NP} \sum_{t=1}^{NT} (\alpha_{it} QE_{it} + \beta_{it} y_{it})$$

sujeito a

Relações *p.18-b* a *p.18-d*

Relações *p.18-h* a *p.18-i*

Relações *p.19* e *p.20*

p.28-b 
$$\sum_{l=1}^{NM} Sal_{jlt} + \sum_{k \in I(j)} W_{ijt} - \sum_{l=1}^{NM} Pur_{jlt} - \sum_{k \in O(j)} W_{ijkt} = 0 \quad , \quad \forall j, t$$

p.28-c 
$$\sum_{j \in JM(i,s)} \sum_{s \in PS(i)} \theta_{ijst} - H_{ij} Q_{it} \leq 0 \quad , \quad \forall i, j, t$$

p.28-d 
$$W_{ijt} - \sum_{s \in PS(i)} \rho_{ijs} \theta_{ijst} = 0 \quad , \quad \forall i, j, t$$

p.28-e 
$$W_{ijt} - \sum_{s \in PS(i)} \mu_{ijs} W_{ij'st} = 0 \quad , \quad \forall i, j, t$$

p.28-f, g 
$$Sal_{jlt}, Pur_{jlt}, W_{ijt}, W_{ijst}, QE_{it}, Q_{it}, \theta_{ijst} \geq 0 \quad ; \quad y_{it} \in \{0,1\} \quad , \quad \forall i, j, l, t$$

Neste planeamento multiperíodo de longo prazo, são necessários os dados referentes às previsões das disponibilidades de materiais, de procuras de produtos, dos preços destas matérias químicas, bem como valores previsionais para os custos de operação e de investimento de todos os processos. A expansão das capacidades, exigível perante o crescimento das procuras de um produto, ocorrerá: *i*) apenas depois da utilização total da capacidade de produção existente; *ii*) apenas depois da flexibilização dos esquemas de produção, preferindo os produtos mais lucrativos e diminuindo a utilização da capacidade instalada pelos produtos de menor interesse económico; *iii*) quando as perdas, por vendas possíveis e não concretizadas dos produtos menos lucrativos, suplantarem os custos associados com a expansão de capacidades; *iv*) quando atingido o limite de disponibilidade de alguma matéria-prima, for justificável a expansão de um processo que utilize esquemas de produção e componentes materiais alternativos.

Implementando este modelo *plan\_milp\_lrp\_schem*, é então possível considerar a flexibilidade decorrente dos diferentes esquemas de produção em cada processo, sem recorrer a reformulações específicas ou à adição de processos *dummy*. Dado que os referidos esquemas de produção foram explicitados aquando do desenvolvimento do modelo, verifica-se que esta formulação é mais eficiente do que a do modelo *plan\_milp\_lrp\_flex*, necessitando de menor número de variáveis binárias e contínuas, de menor número de restrições, o que se traduz numa substancial redução do esforço computacional. Esta característica torna-se mais relevante quando o número de processos, de esquemas de produção alternativos, de componentes (produtos finais e intermédios, de matérias-primas), atinja dimensões significativas.

#### 2.1.4 Afição das técnicas de resolução e investigação analítica

Para redes de processos de grande complexidade, concretamente quando o número de períodos do horizonte temporal for elevado, obtêm-se modelos de planeamento com numerosas variáveis e restrições, mas onde o nível de dificuldade vem associado às variáveis binárias. Aplicando um método enumerativo, como o *branch-and-bound*, o esforço computacional aumenta exponencialmente, dado ser necessário examinar um crescente número de nós, o que poderá até inviabilizar a confirmação da optimalidade de uma solução de boa qualidade.

A estratégia de reformulação aplicada por Sahinidis e Grossmann [1991a, 1992] permite obter modelos de MILP que apresentam relaxações lineares de melhor qualidade, especialmente para problemas de grande dimensão onde se verificaram significativas reduções do tempo de cálculo (até uma ordem de grandeza), ou foi mesmo possível a resolução de algum problema não finalizado em tempo útil pelo *branch-and-bound*.

Essa estratégia baseia-se na observação de que o modelo *plan\_milp\_lrp* para planeamento e expansão de capacidades, quando especificados os valores dos fluxos de produção  $W$ , das quantidades compradas  $Pur$  e vendidas  $Sal$ , poderá ser resolvido como uma série de subproblemas independentes de ELS. Aplicando, ou uma técnica de desagregação de variáveis [Krarup e Bilde, 1978] ou uma de redefinição de variáveis [Martin, 1987], a relaxação linear do subproblema ELS de MILP resulta exacta e óptima. Esta reformulação da subestrutura matemática de ELS, a qual no modelo de planeamento representará o subproblema de expansão das capacidade, permite então obter desempenhos numéricos comprovadamente superiores.

Supõe-se que exista a necessidade de expandir a capacidade dos processos, que corresponde à “procura da capacidade de produção” decorrente do sucessivo crescimento da procura de produtos. Tal necessidade ou procura deverá ser satisfeita pela expansão de capacidade  $QE$ , a variável de “produção da capacidade” a ser tratada. Seja:

$\varphi_{it\tau}$  – variável (de “produção de capacidade”), corresponde à produção num período  $t$  para satisfazer a procura acumulada (de capacidade) até ao período  $\tau$  posterior;

$QE_{it\tau}^{accum}$  – limite superior (*upper bound*) para a expansão de capacidades, respeitante à “procura de capacidade” acumulada desde o período  $t$  até ao período  $\tau$ .

Adicionam-se as restrições adequadas à reformulação de Martin [1987], descrevendo que a “produção de capacidade” destinada a satisfazer a “procura de capacidade” num período posterior, não poderá obviamente ultrapassar o valor da expansão (“produção”) de capacidade no período em que a própria é “produzida”,

**p.29** 
$$\varphi_{it\tau} \leq QE_{it} \quad , \quad \forall i, t, \tau \geq t$$

exprimindo uma *upper bound* de cariz lógico para as variáveis de “produção de capacidade”,

**p.30** 
$$\varphi_{it\tau} \leq QE_{it\tau}^{accum} y_{it} \quad , \quad \forall i, t, \tau \geq t$$

e delimitando de forma exacta a ocorrência consecutiva de “produção de capacidade”,

**p.31** 
$$\sum_{\tau=1}^t \varphi_{i\tau} \geq (W_{m,t} - Q_{i0}) \quad , \quad \forall i, t, \tau \leq t$$

Introduzindo estas  $\{3 NP.NT [NT.(NT+1)/2]\}$  restrições, obtém-se o modelo *plan\_milp\_lrp\_ref* para planeamento de processos em horizonte de longo prazo, que apresenta como vantagem a diminuição do desvio da relaxação linear, o que se traduzirá por um menor esforço em termos de procura ao longo da “árvore” do algoritmo de *branch-and-bound*.

**Modelo *plan\_milp\_lrp\_ref*:**

**p.32-a** 
$$[max] \xi = \sum_{j=1}^{NC} \sum_{l=1}^{NM} \sum_{t=1}^{NT} (\gamma_{jlt} Sal_{jlt} - \Gamma_{jlt} Pur_{jlt}) - \sum_{i=1}^{NP} \sum_{t=1}^{NT} \delta_{it} W_{sit} - \sum_{i=1}^{NP} \sum_{t=1}^{NT} (\alpha_{it} QE_{it} + \beta_{it} y_{it})$$

sujeito a

Relações *p.18-b* a *p.18-i*

**p.32-b** 
$$\varphi_{it\tau} - QE_{it} \leq 0 \quad , \quad \forall i, t, \tau \geq t$$

**p.32-c** 
$$\varphi_{it\tau} - QE_{it\tau}^{accum} y_{it} \leq 0 \quad , \quad \forall i, t, \tau \geq t$$

$$\text{p.32-d} \quad \sum_{\tau=1}^t \varphi_{i\tau} - (W_{m,t} - Q_{i0}) \geq 0, \quad \forall i, t, \tau \leq t$$

$$\text{p.32-e, f} \quad \text{Sal}_{jt}, \text{Pur}_{jt}, W_{it}, QE_{it}, Q_{it}, \varphi_{it\tau} \geq 0; \quad y_{it} \in \{0,1\}, \quad \forall i, j, l, t$$

Este modelo *plan\_milp\_lrp\_ref* para planeamento de longo prazo apresenta um número de variáveis e de restrições que é obviamente superior ao do modelo original, *plan\_milp\_lrp*, mas a introdução dos descritos planos de corte (*cutting planes*) no espaço de procura do problema original [Sahinidis e Grossmann, 1991a e 1992] possibilita uma redução significativa do tempo de cálculo para problemas de grandes dimensões. No entanto, para aproveitamento deste tipo de técnica, a respectiva metodologia deverá encontrar-se já a um nível de sistematização adequado.

Também Liu e Sahinidis [1995] analisam diversos aspectos da implementação computacional do problema de LRP, como sejam os efeitos da discretização dos períodos de tempo e da incerteza inerente aos dados previsionais sobre a qualidade da solução obtida. Essencialmente, verificam que:

- à medida que aumenta o número de períodos de tempo no horizonte considerado, o desvio da relaxação linear do MILP tende a reduzir-se — mas quanto maior a discretização temporal, maior será a necessidade de exactidão nas previsões dos dados;
- aparentemente, a incerteza nos valores dos preços e das procuras não apresenta impacto significativo sobre a perda de qualidade da solução, caso se possibilite o ajustamento às flutuações observadas nos dados através do controlo das variáveis de decisão numa segunda fase, ou seja, permitindo o ajustando dos níveis de produção, das compra de materiais, de venda de produtos— desvios de até 20% nos dados do problema revelam-se toleráveis, originando uma perda inferior a 2% do VLA.

Em termos de metodologias de resolução, Liu e Sahinidis apresentam um algoritmo de *branch-and-cut* específico para este problema de planeamento, onde se aprofunda a reformulação ELS descrita anteriormente [Sahinidis e Grossmann, 1991 e 1992], pois adiciona iterativamente planos de corte em vez de introduzir as variáveis de desagregação. Então, o modelo matemático utilizado corresponde ao modelo original *plan\_milp\_lrp*, ao qual se adiciona a seguinte restrição de corte:

$$\text{p.33} \quad W_{m,t} - \sum_{\tau=1}^t [\sigma_{i\tau} QE_{i\tau} + (1 - \sigma_{i\tau}) QE_{i\tau}^{accum} y_{i\tau}] \leq Q_{i0}, \quad \forall i, t$$

O plano de corte engloba justamente o trio de restrições a incluir no modelo quando se procede à desagregação das variáveis do subproblema de ELS, para tal considerando o parâmetro disjuntivo  $\sigma$  (para incorporação exclusiva de uma ou de outra parcela no somatório):  $\sigma_{i\tau} \in \{0,1\}, \quad \forall i, t$ .

Além da incorporação dos referidos planos de corte, a implementação do algoritmo de *branch-and-cut* prevê a utilização de heurísticas para a obtenção de uma solução admissível inicial de boa qualidade, mas que representam perspectivas algo antagónicas: por um lado, quando o número de períodos de tempo é diminuto, pretende estimar-se quais os processos a expandir no período inicial, ou estimar o

período de tempo para o qual se deverá efectivar a expansão inicial de um processo; por outro lado, quando o número de períodos é elevado, procede-se a uma agregação desses períodos aos pares, resolve-se o problema de MILP com uma redução para metade do número de variáveis binárias, e obtém-se a estimativa admissível para o problema original atribuindo cada expansão prescrita ao primeiro período de cada par de períodos.

Em seguida, e como ferramenta teórica para caracterização do comportamento de heurísticas e de algoritmos de optimização, Liu e Sahinidis [1997] propõem a utilização de investigação (por via) analítica, de forma a desenvolver um enquadramento que conjugue os aspectos mais valiosos de ambas as abordagens (respectivamente, resolução rápida *vs.* rigor e exactidão) e que elimine os pontos mais débeis (garantia de optimalidade *vs.* combinatoriedade). Por um lado, interessará introduzir rigor na aplicação de heurísticas; por outro lado, convém proceder à resolução aproximada de problemas de optimização combinatoria, mas sem comprometer grandemente a optimalidade da solução. Contudo, a análise do desempenho de heurísticas baseia-se geralmente em evidências empíricas, desligadas do suporte teórico dos métodos exactos. Daí a necessidade de desenvolver-se uma análise teórica das heurísticas para problemas de planeamento de processos, similarmente ao que tem acontecido no contexto de outros problemas de optimização combinatoria.

Liu e Sahinidis [1997] desenvolvem uma heurística baseada em LP, caracterizando-se o seu desempenho não só por análise do pior caso (*worst case analysis*), mas também através de uma mais completa análise do comportamento probabilístico médio. Desta forma, será possível a resolução deste difícil problema de MILP através de uma heurística cujo comportamento provável será próximo do óptimo, ou seja, combinar-se-á a rapidez característica dos métodos heurísticos com a qualidade de solução providenciada pela optimização matemática.

Os resultados principais assim obtidos, associados com o aumento do número de períodos de tempo e com a dificuldade combinatoria correspondente:

- o desvio do programa linear por relaxação das restrições de integralidade (LP-rel) não piora, dissipando-se com “probabilidade um” (*with probability one, w.p. 1*) quando o horizonte temporal é infinito;
- a heurística desenvolvida, tendo por base uma resolução mais simples de LP, providencia “quase seguramente” (*almost surely, a.s.*) soluções óptimas para problemas de grande dimensão;

Como conclusão principal, confirma-se a necessidade de investigação analítica dos algoritmos exactos e heurísticos, de forma a complementar as usuais investigações empíricas dos problemas da engenharia de sistemas de processo.

Estes investigadores sugerem alguns desenvolvimentos possíveis no contexto de expansão de capacidades de processos, em problemas (de formulação temporal discreta) de sequenciamento e de planeamento de processos, em problemas de síntese de processos e de operações, como sejam:

- integração da heurística com um esquema de relaxação lagrangeana, para problemas com restrições adicionais;
- desenvolvimento de resultados probabilísticos assintóticos quando o número de processos em rede aumenta;

- melhorar a alocação óptima de capacidade, tendo em conta os fluxos óptimos obtidos do LP-rel;
- verificar em que medida a discretização fina de formulações multiperíodo (discretização multiestágio, em síntese de processos) permitirá obter melhores valores do LP-rel;
- a investigação analítica de outros problemas (síntese de processos, de operações) aparenta ser prometedora; aí, o estudo deve iniciar-se por uma análise formal da complexidade computacional, pois embora o número de soluções candidatas aumente exponencialmente com a dimensão do problema, tal não significa que o problema não possa ser resolvido em tempo polinomial; também, as provas e demonstrações de complexidade baseiam-se na redutibilidade entre problemas, o que fornece informações úteis sobre as inter-relações entre os problemas de optimização e de projecto, e que podem servir de aproximações;
- finalmente, podem obter-se conhecimentos significativos em termos algorítmicos, heurísticos e da estrutura dos problemas, de forma a potenciar o desenvolvimento de novas heurísticas (algoritmos aproximados) com bom desempenho, em termos médios.

Também no contexto do problema de planeamento de processos, Ahmed e Sahinidis [2000] pretendem saber se existe a possibilidade de desenvolver algoritmos exactos e eficientes. Desenvolveram investigação analítica para verificar que são da classe NP-hard, tanto a versão estática, como a versão dinâmica do seu modelo de planeamento de processos, o que foi determinado através de uma análise da complexidade computacional.

Tendo verificado teoricamente que não é possível desenvolver algoritmos polinomiais exactos, Ahmed e Sahinidis desejam promover a sistematização de abordagens de qualidade, para buscar métodos que permitam boas soluções, obtidas em tempo razoável e comprovadas estatisticamente de uma forma bem provável: verificaram ser possível construir uma heurística cujo erro diminui assimptoticamente quando a dimensão do problema aumenta.

O método heurístico desenvolvido baseia-se na perturbação da solução obtida do LP-rel, para se proceder à construção das soluções admissíveis. Os problemas assim tratados pela heurística permitem aproximar o valor óptimo da função objectivo a menos de 1%, utilizando uma só resolução do LP-rel. Também através da análise probabilística, explana-se o fenómeno da aproximação assimptótica ao óptimo com o número de períodos, onde o desvio do LP-rel se vai dissipando.

### 2.1.5 Incerteza no planeamento de processos

Ierapetritou e Pistikopoulos [1994] apresentam um algoritmo baseado no método de decomposição de Benders para optimização estocástica de problemas de planeamento de processos sob incerteza, considerando como variáveis complicativas as variáveis referentes às decisões sobre as capacidades dos processos e as respectivas decisões de expansão, ao longo do horizonte temporal. Assim, as variáveis binárias são tratadas em primeira fase, enquanto que em segunda fase se obtém um subproblema onde se procura aproximar o cálculo do integral multivariável conducente ao valor esperado dos custos de recurso (recourse costs).

O algoritmo proposto requer elevado esforço computacional, dado recorrer a quadratura gaussiana para avaliar o valor esperado dos custos de recurso, ou seja, da função objectivo em segunda fase. Os pontos da quadratura não são previamente calculados, mas serão obtidos ao longo do processo de optimização através da resolução sequencial de subproblemas que definem os contornos da região possível, mas cujo número aumenta exponencialmente face ao número de parâmetros incertos. Então, evitam-se instâncias de inadmissibilidade ao colocar estes pontos em plena região possível, pelo que se assegura a fiabilidade do plano de investimentos simultaneamente com a respectiva optimização.

Liu e Sahinidis [1996] abordam o problema estocástico de planeamento de processos sob incerteza, desenvolvendo e aplicando um algoritmo de decomposição (de Benders) que não requer a discretização antecipada do espaço probabilístico, e que permitirá: por um lado, obter efectivamente a solução exacta do problema de 2SSP de grande dimensão; por outro lado, obtê-la em tempo reduzido, ou mesmo em tempo real.

Para tal, este algoritmo recorre a uma amostragem de Monte Carlo para estimar o valor esperado da função objectivo, no intuito de que a amostragem providencie estimativas consistentes da função objectivo, sem a necessidade da resolução explícita de um excessivo número de subproblemas de LP. Outros aspectos que beneficiam essa resolução numérica:

- apenas se requer a resolução subóptima do problema mestre de MILP durante as primeiras iterações do algoritmo, o que permitirá a resolução de problemas de grande dimensão e facilitará o tratamento das variáveis binárias associadas às expansões de capacidade; de facto, em cada iteração, apenas interessará garantir a obtenção de uma solução inteira admissível que verifique os limites (*lower bound* e *upper bound*) actuais do método de decomposição de Benders, evoluindo-se logo em seguida para a próxima iteração, para a resolução dos subproblemas separáveis (por período de tempo e por cenários) de LP;
- a incorporação de restrições sobre o espaço de procura admissível (*feasibility restrictions*), é efectuada de forma a garantir as condições para recurso completo (*complete recourse*) e também de maneira a acelerar a convergência do método de decomposição; para tal, são introduzidas no problema mestre as restrições do “pior cenário possível” (considerando para cada restrição de *lower bound*, o máximo dos valores limite para todos os cenários; para cada restrição de *upper bound*, seleccionando o mínimo de todos os valores-limites associados).

A modelação do problema de optimização MILP para planeamento de processos, com incerteza nos termos independentes (*right hand side*, RHS), considera na função objectivo o valor esperado respeitante às variáveis de recurso, enquanto que o espaço de procura probabilístico é definido através da discretização das relações onde se apresentam as variáveis de carácter aleatório.

**Modelo *plan\_milp\_spp*:**

$$\text{p.34-a} \quad \min \xi = \sum_{i=1}^{NP} \sum_{t=1}^{NT} (\alpha_{it} QE_{it} + \beta_{it} y_{it}) + \sum_{r=1}^{NR} \text{prob}_r \left[ \sum_{i=1}^{NP} \sum_{t=1}^{NT} (\delta_{it} W_{rit}) + \sum_{j=1}^{NC} \sum_{t=1}^{NT} (\Gamma_{jt} \text{Pur}_{rjt} - \gamma_{lt} \text{Sal}_{rjt}) \right]$$

sujeito a

Relações *p.18-b* a *p.18-d*Relações *p.19* e *p.20*

$$\text{p.34-b} \quad W_{rit} - Q_{it} \leq 0, \quad \forall r, i, t$$

$$\text{p.34-c} \quad \sum_{l=1}^{NM} \text{Sal}_{rjlt} + \sum_{i=1}^{NP} \eta_{ij} W_{rit} - \sum_{l=1}^{NM} \text{Pur}_{rjlt} - \sum_{i=1}^{NP} \mu_{ij} W_{rit} = 0, \quad \forall r, j, t$$

$$\text{p.34-d} \quad a_{rjlt}^{Low} \leq \text{Pur}_{rjlt} \leq a_{rjlt}^{Upp}, \quad \forall r, j, l, t$$

$$\text{p.34-e} \quad d_{rjlt}^{Low} \leq \text{Sal}_{rjlt} \leq d_{rjlt}^{Upp}, \quad \forall r, j, l, t$$

$$\text{p.34-f, g} \quad \text{Pur}_{rjlt}, \text{Sal}_{rjlt}, W_{rit}, QE_{it}, Q_{it} \geq 0; y_{it} \in \{0,1\}, \quad \forall i, j, l, r, t$$

Clay e Grossmann [1997] utilizam também o enquadramento do 2SSP, postulando funções de distribuição de probabilidades gerais, para o tratamento do problema de planeamento de processos em ambiente incerto. Propõem um algoritmo de desagregação sucessiva baseado em análise de sensibilidade e que, não atendendo às variáveis binárias de expansão de capacidades, efectua iterativamente partições no espaço probabilístico para assim assegurar a convergência para a solução exacta.

Dado que o problema determinístico equivalente apresenta um número extremamente elevado de variáveis e restrições, a implementação em arquitectura computacional paralela aproveita variantes do método de decomposição de Benders, especificamente desenvolvidas para a estrutura do problema de planeamento de processos. Então, trata-se de:

- agregar as incertezas nos coeficientes dos custos e em RHS segundo partições do espaço probabilístico de segunda fase, onde se reaplicam sucessivamente aproximações aos valores (aritméticos) médios; obtém-se um modelo agregado, construído por combinação linear das restrições e dos termos da função objectivo com específico interesse;
- desenvolver e aplicar propriedades teóricas do algoritmo de desagregação sucessiva para 2SSP, quanto à delimitação do espaço de procura para o modelo agregado e no respeitante à solução exacta a obter, e atendendo à convexidade do problema de recurso;
- caso necessário, introduzir variáveis de folga nas restrições estocásticas, associando-lhes parâmetros penalizadores na função objectivo, pois assim é

possível assegurar a verificação numérica de todas as restrições estocásticas, e evitar que algum subconjunto de valores incertos com probabilidade diminuta venha impedir a efectivação de uma solução de boa qualidade.

Ahmed e Sahinidis [1998] desenvolvem optimização robusta para o problema de longo prazo em indústrias de processo, que permite a obtenção de um plano adverso ao risco mesmo num ambiente altamente variável, pois trata-se a variabilidade dos custos de recurso ao longo de cenários para os parâmetros incertos, onde:

- para evitar formulações não-lineares, bem como para evitar soluções estocasticamente dominadas, propõe-se a utilização da média parcial superior (*upper partial mean*, UPM) como valor de referência na medida de variabilidade, o desvio positivo *dsvp* relativamente à média; assim, evita-se a utilização da variância que, além de introduzir não linearidades no modelo, constitui também uma medida simétrica do risco, penalizadora tanto das soluções conducentes a valores de custo superiores ao custo esperado como das soluções de boa qualidade, as conducentes a custos inferiores;
- os modelos desenvolvidos, a partir do espectro de soluções obtidas e dos respectivos graus de robustez nos parâmetros incertos de recurso, providenciam uma útil ferramenta de análise do lucro esperado e da sua variabilidade;
- se desenvolve um procedimento heurístico para a formulação de recurso restringida, que força iterativamente em cada passo a robustez de recurso enquanto trata o programa estocástico; obtêm-se aqui resultados ligeiramente mais conservativos para os valores limites do lucro e para a respeitante média parcial superior.

A formulação correspondente ao modelo para optimização robusta (*robust optimization*, RO), com incerteza em RHS, considera:

**Modelo *plan\_milp\_spp\_ro*:**

$$\begin{aligned}
 \text{p.35-a} \quad \min \xi = & \sum_{i=1}^{NP} \sum_{t=1}^{NT} (\alpha_{it} QE_{it} + \beta_{it} y_{it}) \\
 & + \sum_{r=1}^{NR} \text{prob}_r \left[ \sum_{i=1}^{NP} \sum_{t=1}^{NT} (\delta_{it} W_{rit}) + \sum_{j=1}^{NC} \sum_{t=1}^{NT} (\Gamma_{jt} \text{Pur}_{rjt} - \gamma_{lt} \text{Sal}_{rjt}) \right] \\
 & + \omega \sum_{r=1}^{NR} \text{prob}_r .dsvp_r
 \end{aligned}$$

sujeito a

$$\begin{aligned}
 \text{p.35-b} \quad dsvp_r \geq & \left[ \sum_{i=1}^{NP} \sum_{t=1}^{NT} (\delta_{it} W_{rit}) + \sum_{j=1}^{NC} \sum_{t=1}^{NT} (\Gamma_{jt} \text{Pur}_{rjt} - \gamma_{lt} \text{Sal}_{rjt}) \right] \\
 & - \sum_{r=1}^{NR} \text{prob}_r \left[ \sum_{i=1}^{NP} \sum_{t=1}^{NT} (\delta_{it} W_{rit}) + \sum_{j=1}^{NC} \sum_{t=1}^{NT} (\Gamma_{jt} \text{Pur}_{rjt} - \gamma_{lt} \text{Sal}_{rjt}) \right], \quad \forall r
 \end{aligned}$$

Relações *p.18-b* a *p.18-d*

Relações *p.19* e *p.20*

Relações *p.34-b* a *p.34-e*

**p.35-c, d**  $dsvp_r, Pur_{rjt}, Sal_{rjt}, W_{rit}, QE_{it}, Q_{it} \geq 0; y_{it} \in \{0,1\}, \forall i, j, l, r, t$

Com ligeiras diferenças relativamente à formulação robusta em *plan\_milp\_spp\_ro*, desenvolve-se a formulação do problema com restrição aos custos de recurso *plan\_milp\_spp\_rr* (*restricted recourse*, RR) considerando o parâmetro de tolerância,

$\varepsilon$  – nível prescrito para tolerância do valor da média parcial superior, *UPM*;

apresentando como função objectivo,

$$\min \xi = \sum_{i=1}^{NP} \sum_{t=1}^{NT} (\alpha_{it} QE_{it} + \beta_{it} y_{it})$$

**p.36**

$$+ \sum_{r=1}^{NR} prob_r \left[ \sum_{i=1}^{NP} \sum_{t=1}^{NT} (\delta_{it} W_{rit}) + \sum_{j=1}^{NC} \sum_{t=1}^{NT} (\Gamma_{jt} Pur_{rjt} - \gamma_{lt} Sal_{rjt}) \right]$$

e supondo a adição da seguinte restrição:

$$\text{p.37} \quad \sum_{r=1}^{NR} prob_r \cdot dsvp_r \leq \varepsilon$$

As propostas [Ahmed e Sahinidis, 1998] para futura investigação prendem-se com a dificuldade destes problemas originada pela inseparabilidade dos termos de variabilidade, como se verifica na métrica em *dsvp*, o que não permite a aplicação directa dos métodos de amostragem e desagregação do modelo estocástico, prevendo-se:

- desenvolvimento de algoritmos exactos e eficientes direccionados à resolução de problemas de optimização com medida de robustez;
- desenvolvimento de algoritmos exactos e eficientes direccionados à resolução de modelos de recurso restringidos;
- analisar a generalização da média parcial como critério de variabilidade em outros problemas, como sejam os problemas de sequenciamento de operações e de síntese de sistemas de processos químicos.

Essencialmente, no âmbito da optimização de processos sob incerteza, interessa obter um plano que optimize o objectivo estocástico, seja o valor esperado do custo, lucro ou flexibilidade do plano. Assim, Ahmed *et al.* [2000] propõem uma modificação ao algoritmo de decomposição de Ierapetritou e Pistikopoulos [1994], baseado no método de decomposição de Benders, de forma a evitar a imposição de constrangimentos aos parâmetros incertos, o que permite uma modelação mais realística da incerteza. Por análise teórica desse algoritmo, é possível desenvolver um algoritmo melhorado que evita mínimos locais e permite equacionar mais rigorosamente a relação entre custo e flexibilidade. O algoritmo melhorado é mais rápido (3 a 6 vezes) do que o original de Ierapetritou e Pistikopoulos, o que é conseguido evitando a resolução de subproblemas para avaliação da região

admissível, pois o número dos referidos subproblemas aumenta exponencialmente face ao número de parâmetros incertos. Por conseguinte, e neste contexto do planeamento de processos sob incerteza [Ahmed *et al.*, 2000]:

- são desenvolvidas as propriedades de recurso (*recourse properties*), completas e relativamente completas, neste tipo de 2SSP; discute-se o problema do planeamento de processos sob incerteza; explica-se o algoritmo FR (*feasible region*) que avalia sucessivamente a região admissível em cada subproblema e, caso necessário, introduz cortes de admissibilidade para impossibilitar uma solução;
- verifica-se, por via numérica e através da referida abordagem de decomposição (baseada na integração apenas da região admissível), que a não satisfação dessas propriedades de recurso se pode traduzir em soluções subóptimas ou em não obtenção do grau de flexibilidade desejado; teoricamente, comprova-se que a formulação subjacente é não-convexa, donde a possível convergência subóptima do algoritmo;
- tomam-se medidas analíticas reforçadoras das propriedades de recurso, modificando o conjunto de restrições da primeira fase (de forma a que a solução assim obtida seja admissível segundo o algoritmo FR); essas medidas levam ao desenvolvimento do algoritmo melhorado, o qual não tratará dos diversos mínimos locais, bem como relacionará de forma mais rigorosa os parâmetros de custo e de flexibilidade — sendo o grau óptimo de flexibilidade aquele que maximiza a diferença entre o lucro esperado e o custo esperado de reprojecto, combinando assim os dois objectivos conflitantes;
- apresentam-se exemplos ilustrativos da resolução numérica e implementação computacional do novo algoritmo, exemplos que são obtidos da literatura.

## 2.2 O dimensionamento e sequenciamento de processos descontínuos (*design and scheduling of batch processes*)

Dada a tendência de aproximação da produção industrial ao mercado de consumo, o fabrico de produtos de elevado valor acrescentado traduz-se num renovado interesse pelos processos descontínuos, visando-se a construção de unidades de fabrico com superior flexibilidade, bem como a diminuição do período de desenvolvimento destes processos.

Nesta secção descrevem-se as principais alternativas a ser avaliadas no domínio do dimensionamento e sequenciamento para processos descontínuos de produção. Atendendo a que o número de alternativas possíveis é muito elevado, contendo factores económicos complexos e de difícil comparação, torna-se necessário sistematizar as abordagens efectuadas, que se baseiam sobretudo em métodos matemáticos de optimização. Privilegia-se o tratamento por MP do dimensionamento e sequenciamento de processos descontínuos, considerando unidades multiproducto (ambiente de produção *flowshop*) e tratando o dimensionamento das unidades em

cada estágio em modelação contínua ou discreta. O sequenciamento das operações em cada ciclo é englobado na fase de dimensionamento do sistema, o que dificulta a resolução do problema, em especial quando se consideram campanhas de produtos múltiplos (*multiple product campaign*, MPC).

Os processos descontínuos são típicos na indústria farmacêutica, alimentar, de especialidades químicas, de plásticos e tintas, ou outras empresas industriais com processos químicos de fabrico onde, de acordo com Douglas [1988]:

- as quantidades necessárias do produto são diminutas, para tal bastando uma pequena fracção do tempo de produção em cada ano;
- o processo de produção é sensível a variações, a informação sobre o seu desenvolvimento é escassa;
- a vantagem competitiva, com vista a lançar o produto no mercado, exige rapidez na implementação e operacionalidade do processo produtivo;
- o ciclo de vida comercial do produto ou especialidade química é curto, sendo expectável a aparição a breve trecho de um produto substituinte (mais barato, menos tóxico, de melhor manipulação e transporte);
- o valor do produto ultrapassa largamente o custo da sua produção, que acaba por não ser significativo em termos dos custos globais.

Dado que as quantidades produzidas são usualmente diminutas, as unidades de fabrico de pequena volumetria são utilizadas para uma variedade de produtos, que partilham recursos escassos, já que requerem similares equipamentos. A problemática encontra-se associada tanto ao sequenciamento, como à alocação das ordens de produção, pois as tarefas ou operações necessárias devem ocorrer em cada um dos equipamentos específicos: a homogeneização dos materiais no misturador; a transformação química no reactor; a separação de sólidos na centrífuga ou ciclone; a separação de líquidos em coluna de extracção ou destilação; a operação de secagem no secador, por exemplo, de pratos. Em numerosas situações trata-se de uma alocação específica, onde existe uma relação única entre a tarefa e o equipamento. Os tempos de processamento podem ser representados em gráfico de Gantt, sendo habitual não se representar os tempos das operações de transferência (esvaziamento, transferência, enchimento), pois tais períodos são frequentemente negligenciáveis.

Usualmente, o fabrico de especialidades químicas, de produtos alimentares e farmacêuticos, segue um procedimento (receita galénica) específico, desde a composição das misturas reagentes até a um conjunto de fases, de preparação e de transformação química, com tempos de processamento e condições de operação bem definidos. As condições operacionais e as quantidades tratadas variam substancialmente, além de que os processos contínuos dedicados acabam por se revelar globalmente menos onerosos, mesmo os de diminuta dimensão, em termos de custos, de equipamento e operacionais. Os equipamentos de transferência de calor (arrefecedores, aquecedores) e de massa (bombas centrífugas) operam em modo semicontínuo, dado que os equipamentos principais do processo *batch* operam igualmente em modo descontínuo (misturador, reactor, separador, secador), compondo ciclos de operação. O tempo de ciclo (*cycle time*, *CT*), em cada estágio de produção e equipamento associado, estabelece-se através da diferença entre os tempos final e inicial correspondentes a um dado ciclo de operações. O tempo global

de duração do processamento (*makespan*) corresponde ao período de tempo necessário para completar a produção de um dado número de lotes, ou de todos os lotes da campanha.

Sparrow *et al.* [1975] comparam a utilização de heurísticas com a aplicação de um método *branch-and-bound* para o dimensionamento de equipamento no âmbito de unidades multiproduto em produção descontínua. O procedimento heurístico é um processo iterativo que se desenrola até que o custo global do equipamento cesse de diminuir, consistindo: *i*) na estimativa das dimensões exigíveis para os equipamentos ao longo dos diversos estágios de produção; *ii*) na correspondente conversão para as dimensões normalizadas mínimas; *iii*) na determinação do número de unidades de cada tipo a adquirir. O método *branch-and-bound* consiste no procedimento enumerativo habitual, considerando a diversidade combinatória na dimensão e número dos aparelhos produtivos necessários em cada estágio, visando o custo de investimento mínimo. Sparrow *et al.* preferem a modelação através da enumeração exaustiva, pois tal permitirá incorporar novos constrangimentos, formular a produção de diferentes produtos ou tratar situações de maior complexidade. Todavia, salientam a utilidade da regra heurística para efectuar estimativas rápidas, como para obter a estimativa inicial do método *branch-and-bound*.

**Tabela 2.2** Sinopse dos estudos de dimensionamento e sequenciamento de processos descontínuos multiproduto (*flowshop*).

Ano	Autores	Assunto
1988	Kocis e Grossmann	Formulação MINLP não-convexa para dimensionamento de processos <i>batch</i>
(1991)	Pekny e Miller	Tratamento de TSP assimétrico
1989	Birewar e Grossmann	Sequenciamento ZW utilizando formulação TSP
1989	Birewar e Grossmann	Incorporação do sequenciamento no âmbito do dimensionamento
1992	Voudouris e Grossmann	Reformulação MILP e discretização das dimensões dos equipamentos
1993	Voudouris e Grossmann	Dimensionamento com sequenciamento dos ciclos e consideração de inventários
1996	Voudouris e Grossmann	Dimensionamento e sequenciamento de unidades <i>batch</i> multiobjectivo ( <i>jobshop</i> )

Em alternativa à implementação de novas unidades de produção, com o elevado investimento assim imobilizado, pode pretender-se um aumento de eficiência, com melhoria das taxas de produção, de utilização da capacidade instalada, ou outros parâmetros que se repercutam sobre o lucro. Nomeadamente, Boykin [1985] tratou as operações de produção de anidrido maleico da companhia norte-americana Monsanto, através da minimização de custos decorrente da afinação óptima dos reactores de oxidação catalítica. Desenvolveu-se um modelo matemático de IP, que se poderá classificar como um problema de alocação de produção, visto que se pretende alocar a cada reactor a respectiva afinação (segundo valores discretos de velocidade do reactor, caudal de alimentação, pressão), de forma a satisfazer os níveis requeridos de produção mensal.

Os benefícios económicos são evidentes quando se considera a enorme dimensão desta empresa industrial, que representava uma quota de mercado de anidrido maleico de cerca de 45% da produção total dos E.U.A., destinada principalmente à indústria automóvel e à produção de aditivos, detergentes e agroquímicos. Assim, mesmo obtendo melhorias unitárias ligeiras, foi possível conseguir ganhos globais consideráveis, nomeadamente através de:

- poupança energética, por operação óptima de equipamentos (reactores, compressores) e respectivo programa de manutenção;
- menor consumo de matérias-primas, por maximização dos indicadores de reacção;
- avaliação de diferentes políticas de produção, adaptando-as às condições de mercado;
- maior eficiência económica dos diferentes fabricos, por modelação das suas funções de custo.

Repare-se que a referida técnica para discretização de variáveis permitiu igualmente a delimitação implícita das condições de operação (caudais, pressões, velocidades de produção, utilidades, factores de eficiência), sem se acrescentar as correspondentes restrições ao modelo de LP, pois apenas foi considerado o universo discreto de afinações possíveis. Em consequência, o cálculo numérico será ainda mais eficiente pois evita *a priori* toda e qualquer situação de impossibilidade e de soluções inadmissíveis.

Repare-se que é limitado o número de casos práticos publicados no âmbito de dimensionamento e sequenciamento de processos descontínuos em ambiente *flowshop*, como é nítido na revisão efectuada por Floudas e Lin [2004], ou noutra mais recente. Assim, referem-se casos de estudo mas direccionados a temáticas próximas, contribuindo assim para uma visão mais alargada.

Zhu e Majozzi [2001] relatam um caso industrial, onde integram o planeamento numa rede de produção *multisite* com o sequenciamento de operações em ambiente *jobshop*. Para tal decompõem o problema de grande dimensão, segundo a sua estrutura angular: o problema de planeamento trata da coordenação geral da rede de produção; através da optimalidade da alocação das matérias-primas, que os autores demonstram, efectua-se a interligação aos subproblemas de sequenciamento. A resolução deste problema determinístico pode requerer várias iterações, para assegurar a convergência entre o plano de coordenação e os planos operacionais de sequenciamento.

Romero *et al.* [2003], visando também a integração do planeamento de produção com o sequenciamento em ambiente *jobshop*, incorporam a evolução determinística dos fluxos financeiros em paralelo com a dos fluxos materiais, de forma a maximizar os lucros da empresa industrial. Este desenvolvimento insere-se na linha seguida por Barbaro e Bagajewicz [2004a, 2004b], no âmbito do planeamento de processos (Sec. 2.1).

Grunow *et al.* [2003] tratam um caso da indústria farmacêutica, onde pretendem a coordenação do sequenciamento (longo) de operações, entre fábricas em diferentes localizações ou mesmo diferentes países. Abordam a logística global da cadeia de fornecimentos através dum modelo MILP, determinístico e de grandes

dimensões, ao qual aplicam diversos esquemas de agregação. O procedimento iterativo conduz a soluções subótimas de boa qualidade.

Também no âmbito da otimização de fluxos numa cadeia de fornecimento, mas para a indústria automóvel italiana, Gnoni *et al.* [2003] relatam o caso dirigido aos sistemas de travões numa abordagem ao fabrico *multisite*. Para tal integram um modelo de simulação com um modelo MILP, incorporando subproblemas de dimensionamento e sequenciamento de lotes. Similar abordagem, integrando subproblemas de localização de instalações com os de transporte, conduziu a resultados significativos [Camm *et al.*, 1997] no âmbito da cadeia de fornecimentos de um empresa industrial norte-americana que se dedica a especialidades químicas.

### 2.2.1 Aspectos do dimensionamento e sequenciamento de processos descontínuos

A tipologia de operação dos sistemas de processos descontínuos depende, fundamentalmente, da estrutura e do conjunto de transformações a processar para obter cada produto, a partir dos diversos materiais e conforme cada esquema de tarefas. Designa-se por unidade multiproducto (*multiproduct batch plant*) aquela unidade de produção que labora num contexto de *flowshop*, onde todos os produtos requerem similares estágios de processamento e seguem a mesma sequência de operações. A unidade multiobjectivo (*multipurpose batch plant*) é aquela que assume um contexto de *jobshop*, onde: *i*) ou nem todos os produtos requerem os mesmos estágios de processamento; *ii*) ou nem todos os produtos seguem igual sequência de operações. Desta maneira, se a estrutura de processamento for similar para todos os produtos, apresentando igual sequência de estágios e operações, então o ambiente de produção designar-se-á por *flowshop* em unidade multiproducto, enquanto que se houver grande discrepância nos tratamentos a aplicar, encara-se um ambiente de produção *jobshop* na unidade multiobjectivo. Porém, nesta tese, não se tratará o sequenciamento em unidades de produção descontínua multiobjectivo (*jobshop*).

Quando é possível estabelecer os ciclos de sequenciamento ou de regularidade da produção, as campanhas de produto único (*single product campaigns*, SPC) ocorrem quando se processa: primeiro, todos os lotes de um dado produto; então, depois, é que se permuta a produção para o fabrico de um outro produto. As campanhas de produtos múltiplos (*multiple product campaign*, MPC) ocorrem quando lotes de diferentes produtos são processados simultaneamente. Geralmente, e caso os tempos de limpeza ou de preparação não sejam significativos quando se procede a mudanças do produto em fabrico, a opção pela campanha de produtos múltiplos aparenta ser mais eficiente, quer no âmbito do tempo de ciclo, quer no âmbito do tempo de duração do processamento.

Supondo que se opera em regime de não-sobreposição (*no-overlapping*), os períodos de não-utilização de equipamento serão elevados, pois o processamento de um novo lote apenas se inicia depois da produção do anterior terminar. Então, optando-se pela sobreposição de operações (*overlapping*), o processamento em simultâneo dos diferentes lotes permitirá: diminuir o tempo global e o tempo de ciclo; aumentar o nível de utilização do equipamento. De acordo com Birewar e Grossmann [1989a], para a produção de um só produto e assumindo sobreposição de operações, o tempo de ciclo  $CT$  corresponderá ao valor máximo dos tempos  $\tau$  de processamento de cada um dos  $M$  estágios:

$$\text{b.1} \quad CT = \max_{j=1..M} \{ \tau_j \}$$

No dimensionamento de processos descontínuos para unidades multiproducto é usual a implementação de modelos complexos de Programação Matemática (MILP, MINLP) cujas finalidades correspondem ao dimensionamento e enumeração de equipamento, simples ou distribuído em paralelo, em ambiente de produção com campanhas de um só produto ou de diversos produtos. Todavia, verifica-se que o dimensionamento de processos descontínuos é afectado pelo sequenciamento e pela utilização de equipamento, pois para melhor decidir sobre qual o equipamento a adquirir deve também enquadrar-se a política de produção. De facto, o tipo de sequenciamento deve ser antecipado na fase de dimensionamento, pois o impacto económico pode ser significativo.

Birewar e Grossmann [1989b] tratam o sequenciamento de produção descontínua em unidades multiproducto, aplicando uma política de espera nula, para a minimização do tempo total necessário para satisfazer as procuras dos produtos. Estes investigadores indicam que a consideração de estratégias eficientes de sequenciamento, aquando do dimensionamento, poderá traduzir-se numa diminuição dos custos de investimento. Dada a natureza combinatoria do problema, pois formulado para uma multiplicidade de produtos, os autores utilizam uma técnica específica [Birewar e Grossmann 1989a], que possibilita a resolução de um problema de MILP com diminutas variáveis binárias, sendo possível uma resolução mais rápida. Esta modelação pretende minimizar os tempos de ciclo através da agregação das múltiplas partidas de produtos, obtendo-se uma boa aproximação à solução original.

A implementação de unidades de armazenagem flexibiliza o sequenciamento das operações, e a eficiência do sistema será beneficiada; logo, a integração na mesma formulação dos níveis de decisão estrutural e operacional, permite enquadrar tanto o sequenciamento como o dimensionamento. São vários os autores — Voudouris e Grossmann [1993], por exemplo — que consideram importante a incorporação de vários níveis de planeamento no mesmo modelo matemático, como o sequenciamento, o dimensionamento e a síntese de processos, no intuito de que se possua uma melhor informação no momento da decisão. Todavia, o estado actual da tecnologia não nos permite considerar simultaneamente e com pormenor todos os níveis de decisão. Se, em termos industriais, não se encontram exploradas muitas das ferramentas matemáticas já desenvolvidas, é certo que tais ferramentas não estão preparadas para os novos problemas, necessitando de progressos. Do exposto, resulta a diferenciação prática dos vários tipos de planeamento.

Ora, considerando-se o ambiente *flowshop* com unidades em paralelo e seguindo uma política de transferência espera nula, o ciclo de tempo  $CT$  específico de cada produto  $i$  (diferença entre os tempos final e inicial do ciclo de produção) depende do número  $NP(j)$  de unidades de processo implementadas em paralelo em cada estágio  $j$ , bem como do respeitante tempo de processamento  $\tau$  de cada produto  $i$ , pelo que:

$$\text{b.2} \quad CT_{ZW}^p = \max_{\substack{j=1..M \\ i=1..N}} \left\{ \frac{\tau_{ij}}{NP(j)} \right\}$$

A opção de processo único em cada estágio é directamente descrita por esta equação, como um caso específico onde será unitário todo o parâmetro  $NP_j$ .

Supondo campanhas de produto único, uma solução deriva da regra heurística que iguala a dimensão para os lotes de todos os  $i$  produtos. Obviamente, o tempo de

campanha  $t_i$  para cada produto terá de ser determinado, mas o dimensionamento  $B_i$  do lote do produto  $i$  resulta directamente da quantidade a produzir,  $Q_i$ :

$$\text{b.3} \quad B_i = \frac{Q_i}{(t_i/CT_i)}$$

O volume requerido em cada estágio  $V_j$  corresponde ao valor máximo dos volumes necessários à produção dos lotes de dimensão  $B_i$ , atendendo aos respectivos factores de dimensão  $S_{ij}$  (volume do equipamento  $j$  requerido para produzir uma unidade de massa do produto  $i$ ), de forma que,

$$\text{b.4} \quad V_j = \max_{i=1..N} \{V_{ij}\} = \max_{i=1..N} \{S_{ij}B_i\}$$

Note-se, que tal heurística providencia uma aproximação grosseira à solução, pois não leva em conta qualquer factor económico, seja relativamente a custos de investimento, seja a custos operacionais, nem sequer considera a utilização plena da capacidade instalada. Bem pelo contrário, o equipamento será subutilizado caso aqueles factores de dimensão  $S_{ij}$  apresentem valores díspares — para lotes com iguais quantidades, apenas no caso destes factores de dimensão serem iguais é que a utilização de capacidade seria plena.

A política de transferência mais restritiva é a de espera nula, que supõe que cada lote é transferido de imediato para o estágio seguinte. Por exemplo, quando não se encontra disponível qualquer recipiente de armazenagem, ou quando possa ocorrer alguma reacção química indesejável. Por contraponto, a política menos restritiva é a de armazenagem intermédia não-limitada (*unlimited intermediate storage*, UIS), que assume que qualquer lote possa ser armazenado, sem qualquer delimitação à capacidade de armazenagem. Uma política de transferência de meio-termo é a de não-armazenagem intermédia (*no-intermediate storage*, NIS), que apenas permite guardar o lote em fabrico dentro do recipiente onde se processou a última operação.

Embora, em situação real possam coexistir as diferentes políticas de transferência, é previsível que a uma política menos constrangedora possam corresponder menores valores para o tempo de ciclo e para o tempo de duração do processamento. Donde, para uma política de transferência de armazenagem intermédia não-limitada, UIS, o tempo de ciclo  $CT$  corresponderá ao valor máximo, comparando todos os  $M$  estágios, da soma dos tempos de processamento  $\tau$  de todos os  $n$  lotes de produção, para todos os produtos  $i$  [Birewar e Grossmann, 1989b],

$$\text{b.5} \quad CT_{UIS} = \max_{j=1..M} \left\{ \sum_{i=1}^N n_i \tau_{ij} \right\}$$

Ou seja, em *flowshop*, UIS e com máquina única em cada estágio, o tempo de ciclo equivalerá ao tempo que o estágio mais extensivamente utilizado necessita para processar todos os  $N$  produtos (ou ao estágio globalmente mais longo), constituindo-se tal estágio como um ponto de estrangulamento do sistema de produção.

Caso se considere a armazenagem intermédia, então o custo total de inventário  $C_{inv}$  de todos os  $i$  produtos [Voudouris e Grossmann, 1993], supondo um elevado número de lotes de produção, depende directamente do custo de inventário,  $h_i$  (*holding cost*, por unidade de massa do produto  $i$  por unidade de tempo), da quantidade procurada no âmbito do horizonte temporal,  $Q_i$ , da dimensão do horizonte de ciclo,  $H_{cyc}$ , do tempo de campanha de produção afectado ao produto  $i$  em cada ciclo,  $t_i$ :

$$\text{b.6} \quad C_{inv} = \sum_{i=1}^N h_i \left( \frac{Q_i}{2} \right) (H_{cyc} - t_i)$$

### 2.2.2 Dimensionamento e sequenciamento de sistemas descontínuos: campanhas de produto único

Para tratamento do problema de dimensionamento e sequenciamento de processos descontínuos, consideram-se os seguintes parâmetros, alguns já descritos:

$M, N$ , números de estágios  $j$ , de produtos  $i$ , de equipamentos por  
 $NP(j), NS(j)$  – estágio  $p(j)$ , de dimensões discretas dos equipamentos em  
 cada estágio  $s(j)$ ;

$V, dv$  – volumes em cada estágio, respectivamente, contínuos e  
 discretos;

$\alpha, \beta, c$  – parâmetros dos custos em cada estágio, coeficiente e  
 expoente, e os respeitantes custos;

$S, \tau$  – factores de dimensão (*litros/kg*) e tempos de processamento  
 (*horas*), relativos a cada produto  $i$  no estágio  $j$ ;

$H$  – horizonte temporal considerado;

$Q$  – quantidades procuradas, para cada produto  $i$ ;

$CL, SL$  – tempos de limpeza e folgas, para cada transição de  
 produção entre os produtos  $i$  e  $k$ , em cada estágio  $j$ .

A consideração do constrangimento referente ao horizonte temporal  $H$  realiza-se de forma directa, com o tempo de produção  $t$  (para cada produto  $i$ ) decorrente dos números de lotes,  $n$ , e do tempo de ciclo (note-se a coincidência entre campanha e produto),  $CT$ :

$$\text{b.7} \quad \begin{cases} \sum_{i=1}^N t_i \leq H \\ t_i = n_i CT_i \end{cases}$$

Para aplicação deste constrangimento a um sistema com processos múltiplos, devem actualizar-se os tempos de ciclo em conformidade com o número de processos em paralelo. Repare-se que os tempos relativos à "cabeça" (*head*) e "cauda" (*tail*) são usualmente negligenciados, o que é aceitável perante um elevado número  $n_i$  de lotes produzidos ao longo do horizonte temporal, bem como se poderá descontar os tempos de limpeza e transição para a definição do tempo total disponível.

Kocis e Grossmann [1988] apresentaram um modelo MINLP onde consideram processos múltiplos em cada estágio, para satisfazer as quantidades procuradas  $Q_i$  dos diversos produtos. O problema consiste em determinar o número de unidades a implementar em paralelo, proceder ao dimensionamento dessas unidades, bem como definir as quantidades em cada lote de produção e os tempos de ciclo, de forma a minimizar os custos de investimento. Para definir as dimensões dos equipamentos aplicam um modelo contínuo, assumindo-se campanhas de produto único onde o tempo de processamento do *batch* é independente da quantidade produzida.

Porém, dada a não convexidade da função objectivo inicialmente apresentada, o que não permitiria garantir a obtenção do óptimo global, Kocis e Grossmann desenvolveram um conjunto de transformações logarítmicas das variáveis, contínuas e discretas, que permitiu que todas as funções não lineares, na função objectivo ou nas restrições, apresentem uma forma convexa.

No entanto, verifica-se que apenas se encontram disponíveis algumas dimensões, normalizadas, pelo que a adaptação à situação real trata-se rigorosamente por discretização das dimensões dos equipamentos [Voudouris e Grossmann, 1992]. Ou seja, em cada estágio  $j$ , o conjunto discreto  $SV$  descreve as  $NS$  dimensões volumétricas discretas  $dv$  disponíveis para implementação:

$$\text{b.8} \quad SV_j = \{dv_{js}, s = 1..NS(j)\}, \quad \forall j$$

Por introdução das variáveis de decisão binárias,

$$\text{b.9} \quad y_{js} = \begin{cases} 1, & \text{se a dimensão } s \text{ é seleccionada no estágio } j \\ 0, & \text{caso contrário} \end{cases}$$

a alocação de um e um só valor discreto em  $s$  para capacidade da unidade processual, das  $NS(j)$  alternativas possíveis em cada estágio  $j$ , é requerida por:

$$\text{b.10} \quad \sum_{s=1}^{NS(j)} y_{js} = 1, \quad \forall j$$

Então, o inverso do volume da unidade em cada estágio  $j$ , na forma discreta,

$$\text{b.11} \quad \frac{1}{V_j} = \sum_{s=1}^{NS(j)} \frac{y_{js}}{dv_{js}}, \quad \forall j$$

Além do constrangimento ao horizonte de sequenciamento, anteriormente descrito, é necessário garantir que os processos tratem as quantidades  $Q_i$  requeridas para cada produto no tempo de campanha  $t_i$ , e atendendo às especificidades do produto e equipamento:

$$\text{b.12} \quad S_{ij} Q_i CT_i \sum_{s=1}^{NS(j)} \frac{y_{js}}{dv_{js}} \leq t_i, \quad \forall i, j$$

Por conseguinte, o dimensionamento considerando estágios com processo único, operando em campanha de produto único e política de espera nula, pode obter-se do modelo MILP seguinte, *batch\_milp\_fssz* (*Flowshop/ Single machine/ SPC/ ZW*), que trata as dimensões dos equipamentos de forma discreta.

**Modelo *batch\_milp\_fssz*:**

$$\text{b.13-a} \quad [min] \quad z = \sum_{j=1}^M \sum_{s=1}^{NS(j)} c_{js} y_{js}$$

sujeito a

$$\text{b.13-b} \quad S_{ij} Q_i CT_i \sum_{s=1}^{NS(j)} \frac{y_{js}}{dv_{js}} - t_i \leq 0, \quad \forall i, j$$

$$\text{b.13-c} \quad \sum_{s=1}^{NS(j)} y_{js} = 1, \quad \forall j$$

$$\text{b.13-d} \quad \sum_{i=1}^N t_i \leq H$$

$$\text{b.13-e, f} \quad t_i \geq 0, \quad \forall i; \quad y_{js} \in \{0,1\}, \quad \forall j, s$$

O modelo *batch\_milp\_fssz* apresenta:  $N$  variáveis contínuas, correspondentes aos tempos de campanhas  $t_i$  dos  $N$  produtos; até  $(NS.M)$  variáveis discretas, respeitantes às  $NS(j)$  dimensões a seleccionar, para cada um dos  $M$  estágios;  $(N.M + M + 1)$  restrições, respectivamente, associadas às restrições de desempenho da capacidade instalada, de unicidade das capacidades em cada estágio, de observância pelo horizonte de sequenciamento.

Neste problema, pretende-se satisfazer as procuras ao longo de um horizonte, mas ao menor custo de equipamento; atendendo às políticas de produção SPC e de transferência ZW, o tempo de ciclo para cada produto fica associado ao máximo tempo de processamento desse produto nos diferentes estágios, pelo que o número de ciclos de produção fica limitado pelo horizonte; então, para que as quantidades produzidas satisfaçam simultaneamente as procuras de produtos e o constrangimento temporal, é necessário que o sistema de produção *batch* possua capacidade suficiente. O balanceamento económico decorre entre os tempos disponibilizados para cada produto,  $t_i$ , e a definição volumétrica discreta dos estágios, que se encontra associada às variáveis binárias  $y_{js}$ ; para satisfazer o horizonte e não aumentar a dimensão dos lotes, a afectação de maior tempo para um dado produto traduz-se em menos tempo para os outros produtos, pelo que haverá que aumentar a capacidade; assim, interessará implementar uma maior capacidade nos estágios limitantes, definidores dos ciclos de tempo  $CT$ , em função dum menor investimento.

Esta formulação explicita as dimensões discretas disponíveis, o que possibilita melhores soluções do que utilizando o modelo de dimensões contínuas, no qual se procede posteriormente ao ajustamento às capacidades normalizadas: o custo de investimento total diminui, relatando-se [Voudouris e Grossmann, 1992] diminuições de cerca de 5%. Contudo, nesta situação, não se equacionam as implicações decorrentes de um maior número de ciclos mas com menores quantidades de produção, nomeadamente, em termos de paragens e arranques, transições de operações, armazenagem de materiais, ou mesmo o inerente consumo de utilidades.

Voudouris e Grossmann [1992] generalizaram esta modelação considerando a introdução de múltiplas unidades em paralelo  $NP(j)$  nos diferentes estágios, de forma a diminuir os tempos de ciclo  $CT_i$  de cada produto. Recorde-se que o tempo de ciclo para cada produto, em política de espera nula e campanhas de produto único, corresponde ao tempo máximo de processamento do produto ao longo dos estágios. Introduzindo uma segunda (terceira,...) unidade processual no estágio mais longo para o produto e operando desfasadamente, o tempo de ciclo desse produto diminuirá até metade (terça parte,...) do valor inicial.

Introduz-se uma nova variável, o número desagregado de ciclos  $nc_{ijsp}$  (correspondente ao produto cruzado  $n_i \cdot y_{jsp}$ , onde  $y_{jsp}$  apresenta valor unitário apenas caso o estágio  $j$  tenha  $p$  unidades em paralelo de igual dimensão  $s$ ) produzido nas diferentes máquinas de cada estágio  $j$ , que deve observar a igualdade de agregação:

$$\text{b.14} \quad \sum_{s=1}^{NS(j)} \sum_{p=1}^{NP(j)} nc_{ijsp} = n_i, \quad \forall i, j$$

Dada a sua natureza de variável semicontínua, o número desagregado de ciclos respeitará o respectivo limite superior com carácter lógico, pois o número de ciclos será nulo se a correspondente variável binária  $y_{jsn}$  também se anular:

$$\text{b.15} \quad nc_{ijsn} \leq nc_{ijsn}^{Upp} y_{jsn}, \quad \forall i, j, s, n$$

Para todos os produtos  $i$  e em cada estágio  $j$ , o tempo consumido nos ciclos de produção não ultrapassará o tempo disponível para a campanha do produto:

$$\text{b.16} \quad \sum_{s=1}^{NS(j)} \sum_{p=1}^{NP(j)} \left( \frac{\tau_{ij}}{p(j)} nc_{ijsp} \right) \leq t_i, \quad \forall i, j$$

A restrição linear relativa aos volumes discretos expressa-se agora em termos do número de ciclos de produção,  $n_i$ :

$$\text{b.17} \quad n_i \geq S_{ij} Q_i \sum_{s=1}^{NS(j)} \sum_{p=1}^{NP(j)} \frac{y_{jsp}}{dv_{js}}, \quad \forall i, j$$

Então, o dimensionamento da unidade multiproducto, de processos múltiplos em cada estágio, operando em campanha de produto único e seguindo uma política de espera nula, assumindo modelação discreta e normalizada das dimensões dos equipamentos, pode obter-se por resolução do seguinte modelo MILP, *batch\_milp\_fmsz* (*Flowshop/ Multiple machine/ SPC/ ZW*).

**Modelo *batch\_milp\_fmsz*:**

$$\text{b.18-a} \quad [\min] \quad z = \sum_{j=1}^M \sum_{s=1}^{NS(j)} \sum_{p=1}^{NP(j)} c_{jsp} y_{jsp}$$

sujeito a

$$\text{b.18-b} \quad S_{ij} Q_i \sum_{s=1}^{NS(j)} \sum_{p=1}^{NP(j)} \frac{y_{jsp}}{dv_{js}} - n_i \leq 0, \quad \forall i, j$$

$$\text{b.18-c} \quad \sum_{s=1}^{NS(j)} \sum_{p=1}^{NP(j)} y_{jsp} = 1, \quad \forall j$$

$$\text{b.18-d} \quad \sum_{i=1}^N t_i \leq H$$

$$\text{b.18-e} \quad \sum_{s=1}^{NS(j)} \sum_{p=1}^{NP(j)} \left( \frac{\tau_{ij}}{p} nc_{ijsp} \right) - t_i \leq 0, \quad \forall i, j$$

$$\text{b.18-f} \quad nc_{ijsp} - nc_{ijsp}^{Upp} y_{jsp} \leq 0, \quad \forall i, j, s, p$$

$$\text{b.18-g} \quad \sum_{s=1}^{NS(j)} \sum_{p=1}^{NP(j)} nc_{ijsp} - n_i = 0, \quad \forall i, j$$

$$\text{b.18-h} \quad nc_{ijsp}, n_i, t_i \geq 0, \quad \forall i, j, s, p; \quad y_{jsp} \in \{0,1\}, \quad \forall j, s, p$$

O modelo *batch\_milp\_fmsh*, com processos em paralelo, apresenta-se bem mais complicado do que o correspondente modelo com um processo em cada estágio: o número de variáveis contínuas aumenta para  $(2.N + N.M.NS.NP)$ , devido aos números de ciclos, agregados  $n_i$  e desagregados  $nc_{ijsp}$ ; as variáveis discretas são agora  $(M.NS.NP)$ , aumentando no respeitante ao número  $NP(j)$  de processos a seleccionar, para cada estágio; além das  $(N.M + M + I)$  restrições anteriormente consideradas, introduziram-se mais três grupos de restrições, associadas à agregação do número de ciclos desagregado, à restrição lógica de semicontinuidade, à conformação do número de ciclos pelo tempo de campanha, aumentando assim o número de restrições em  $(2.N.M + N.M.NS.NP)$ .

As dificuldades computacionais são evidentes, dado que o número de variáveis binárias aumenta com o número de dimensões alternativas e o número de processos permitidos em paralelo; também o número de variáveis semicontínuas é significativo, bem como são numerosas as restrições aqui introduzidas. O desempenho do método de resolução melhora quando se considera que apenas uma das variáveis binárias não se anula (*special ordered set* do tipo 1, *SOS1*), quando se introduzem cortes ao valor da função objectivo, quando se procede à redução do domínio admissível.

O problema em *batch\_milp\_fmsh* equaciona a implementação de processos em paralelo, logo que se entenda que os tempos de ciclo dos produtos possam beneficiar significativamente com a redução dos tempos de processamento; uma configuração possível para o sistema é a de processo único por estágio, desde que as capacidades máximas disponibilizadas assegurem as quantidades procuradas; porém, numa abordagem de médio e longo prazo, o crescimento sucessivo das procuras pode inviabilizar essa possibilidade, requerendo-se a implementação de processos nos estágios sujeitos a maior constrangimento temporal.

No domínio das configurações possíveis, este modelo traduz o balanceamento económico entre a implementação de um só processo com elevada capacidade, *versus* a implementação de dois ou mais processos de menor dimensão.

### 2.2.3 Dimensionamento e sequenciamento de sistemas descontínuos: campanhas de produtos múltiplos

A adopção de campanhas de produção com produtos múltiplos envolve o sequenciamento dos lotes dos diferentes produtos, no intuito de reduzir tempos de não-utilização do equipamento, o que poderá ser vantajoso caso os tempos das transições sejam diminutos. Então, em ambiente *flowshop* com diversos estágios a processo único, adoptando campanhas de produtos múltiplos e considerando armazenagem intermédia não-limitada, tem-se o seguinte modelo de NLP [Birewar e Grossmann, 1989b], *batch\_nlp\_fsmu* (*Flowshop/ Single machine/ MPC/ UIS*), com dimensões contínuas para os equipamentos,

**Modelo *batch\_nlp\_fsmu*:**

**b.19-a** 
$$[min] \quad z = \sum_{j=1}^M \alpha_j V_j^{\beta_j}$$

sujeito a

**b.19-b** 
$$V_j - S_{ij} B_i \geq 0 \quad , \quad \forall i, j$$

**b.19-c** 
$$\sum_{i=1}^N \frac{Q_i}{B_i} \tau_{ij} \leq H \quad , \quad \forall j$$

**b.19-d** 
$$V_j^{Low} \leq V_j \leq V_j^{Upp} \quad , \quad \forall j$$

**b.19-e** 
$$B_i \geq 0 \quad , \quad \forall i$$

Através de transformação exponencial de variáveis, o modelo reformulado pode apresentar as condições de convexidade que garantam a optimalidade da solução [Kocis e Grossmann, 1988]; de igual modo, também se poderá optar pela discretização das dimensões de equipamento [Voudouris e Grossmann, 1992]. Note-se que: este modelo permitirá obter um valor limite inferior utilizável na resolução de problemas mais complexos, ou com adopção de políticas mais restritivas (de espera nula, sem armazenagem intermédia); ainda, foram negligenciados os custos dos tanques de armazenagem, o que pode constituir uma opção discutível.

**Algoritmo *Folgas\_do\_Ciclo*:**

- |  |
|--|
| <p>1) Definição dos tempos de <i>início</i>, <math>Tini_j</math>, para o produto <math>k</math>:</p> $Tini_1 = \tau_{i1} + CL_{ik1}$ $Tini_j = Tini_{j-1} + \tau_{k,j-1} \quad , \quad j = 2..M$ <p>2) Cálculo das folgas e da folga mínima:</p> $folg_j = Tini_j - \sum_{l=1}^j \tau_{il} - CL_{ikj} \quad , \quad j = 1..M$ $\delta = \min_j \{ folg_j \}$ <p>3) Actualização das folgas:</p> $SL_{ikj} = folg_j - \delta$ |
|--|

A adopção da política de armazenagem UIS permite um tratamento numérico simplificado, devido ao cálculo directo do tempo de ciclo do sistema — neste caso, corresponde ao tempo do estágio globalmente mais demorado, ou seja, à soma dos tempos de processamento de todos os produtos no referido estágio. No entanto, a dificuldade do cálculo será bastante superior quando se considera a restritiva política de espera nula, especialmente se os tempos de transição não forem negligenciáveis. O fundamento para tal aumento de dificuldade reside na determinação do tempo de ciclo, pois para uma política UIS esse valor é obtido directamente, mas noutras

situações (NIS, ZW) requer também a determinação da sequência de produção no ciclo (*cyclic scheduling*).

No entanto, sendo conhecido o número  $n_i$  de lotes a produzir para cada produto, a determinação do tempo de ciclo é obtível por LP. Biegler *et al.* [1997] apresentam um algoritmo que possibilita o cálculo das folgas  $SL_{ikj}$  (*slack*), no caso geral de processamento no estágio  $j$  de um produto  $i$  seguido pelo produto  $k$ , com tempo de limpeza  $CL_{ikj}$  associado, e que se descreveu como Algoritmo *Folgas\_do\_Ciclo*.

Então, para determinar a sequência óptima dos lotes, é aplicada [Pekny e Miller, 1991] uma formulação adaptada da do problema do caixeiro viajante (*traveling salesman problem, TSP*) onde: os nós correspondem aos lotes individuais; os arcos bidireccionados, entre cada par  $(l,m)$  de lotes, representam as transições possíveis em termos de sequência de fabrico. De acordo com a modelação adoptada, a cada um dos sentidos desses arcos bidireccionados encontra-se associada uma variável binária:

$$\text{b.20} \quad y_{lm} = \begin{cases} 1, & \text{se o lote } l \text{ se segue ao lote } m; \\ 0, & \text{caso } l \text{ não suceda a } m; \end{cases}$$

Tal modelação, segundo o problema de caixeiro viajante, apresentará sérias dificuldades de resolução quando o número de lotes for elevado, ou mesmo poderá apresentar soluções com subciclos. Além de que o factor principal do problema de dimensionamento é a minimização do tempo de ciclo  $CT$ , bem como a optimização do número de lotes  $n_i$  para cada produto. Tais dificuldades serão tratadas através da agregação de transições dos  $N$  produtos, definindo-se o número de transições  $Nch$  (*changeovers*) do produto  $i$  para o produto  $k$  através da relação,

$$\text{b.21} \quad Nch_{ik} = \sum_{l \in \text{SetB}(i)} \sum_{l \in \text{SetB}(k)} y_{lm}$$

onde  $\text{SetB}(i)$  representa o conjunto dos lotes onde se processou o produto  $i$ :

$$\text{b.22} \quad \text{SetB}(i) = \{l : \text{o lote } l \text{ corresponde ao produto } i\}$$

Donde, o modelo *cycsched\_ip\_tsp* (de IP, mas frequentemente resolúvel de forma directa através de LP) representa um problema de TSP agregado e conduz à minimização do tempo de ciclo [Birewar e Grossmann, 1989a], obtendo-se também o sequenciamento ao longo desse ciclo de produção.

### Modelo *cycsched\_ip\_tsp*:

$$\text{b.23-a} \quad [\text{min}] \quad CT$$

sujeito a

$$\text{b.23-b} \quad \sum_{k=1}^N Nch_{ik} = n_i, \quad \forall i$$

$$\text{b.23-c} \quad \sum_{i=1}^N Nch_{ik} = n_k, \quad \forall k$$

$$\text{b.23-d} \quad Nch_{ii} \leq n_i - 1, \quad \forall i$$

$$\text{b.23-e} \quad CT - \sum_{i=1}^N \sum_{k=1}^N SL_{ikj} Nch_{ik} \geq \sum_{i=1}^N n_i \tau_{ij} \quad , \quad \forall j$$

$$\text{b.23-f} \quad CT \geq 0; \quad Nch_{ik} = 0, 1, 2, 3, \dots \quad \forall i, k$$

Com base neste modelo agregado e resolúvel através de LP, que foi desenvolvido tendo em vista a minimização do ciclo de tempo, o problema contínuo de dimensionamento dos lotes formula-se tratando o número de lotes  $n_i$  como variáveis de decisão e majorando o tempo global de duração ao horizonte,

$$\text{b.24} \quad \sum_{i=1}^N \left[ n_i \tau_{ij} + \sum_{k=1}^N SL_{ikj} Nch_{ik} \right] \leq H \quad , \quad \forall j$$

Birewar e Grossmann [1989b] desenvolveram o modelo *batch\_nlp\_fsmz*, supondo ambiente de produção multiproducto, em diversos estágios a processo único, adoptando campanhas de produtos múltiplos e considerando política de espera nula, modelo que se apresenta em seguida (*Flowshop/ Single Machine/ MPC/ ZW*).

**Modelo *batch\_nlp\_fsmz*:**

$$\text{b.25-a} \quad [min] \quad z = \sum_{j=1}^M \alpha_j V_j^{\beta_j}$$

sujeito a

$$\text{b.25-b} \quad V_j - S_{ij} B_i \geq 0 \quad , \quad \forall i, j$$

$$\text{b.25-c} \quad n_i B_i = Q_i \quad , \quad \forall i$$

Relações *b.23-b* a *b.23-d* e *b.24*

$$\text{b.25-d} \quad V_j^{Low} \leq V_j \leq V_j^{Upp} \quad , \quad \forall j$$

$$\text{b.25-e} \quad n_i, B_i, Nch_{ik} \geq 0 \quad , \quad \forall i, k$$

De forma semelhante, também se poderá optar pela reformulação já conhecida e referente à discretização das dimensões de equipamento [Voudouris e Grossmann, 1992]. Assim, tem-se um modelo MILP conducente a resultados mais rigorosos do que os obtidos através do modelo NLP e posterior arredondamento aos valores normalizados, pois explicitam-se as dimensões disponíveis para os processos.

O número de lotes produzidos para cada produto  $i$ , encontra-se limitado pelo número de lotes  $n_i$  definido na minimização do ciclo de sequenciamento,

$$\text{b.26} \quad S_{ij} Q_i \sum_{s=1}^{NS(j)} \frac{y_{js}}{dv_{js}} \leq n_i \quad , \quad \forall i, j$$

Similarmente ao modelo anterior de NLP, o tempo de duração do ciclo de sequenciamento é majorado pelo horizonte temporal, bem como são incorporadas as restrições do TSP agregado.

Então, mantendo os pressupostos anteriores (de ambiente *flowshop* com estágios de processo único; de campanhas de produtos múltiplos; de política de

espera nula) Voudouris e Grossmann [1992] apresentaram o modelo aqui designado por *batch\_milp\_fsmz* (*Flowshop/ Single Machine/ MPC/ ZW*).

**Modelo *batch\_milp\_fsmz*:**

$$\text{b.27-a} \quad [\min] \quad z = \sum_{j=1}^M \sum_{s=1}^{NS(j)} c_{js} y_{js}$$

sujeito a

$$\text{b.27-b} \quad S_{ij} Q_i \sum_{s=1}^{NS(j)} \frac{y_{js}}{dv_{js}} - n_i \leq 0, \quad \forall i, j$$

$$\text{b.27-c} \quad \sum_{s=1}^{NS(j)} y_{js} = 1, \quad \forall j$$

Relações *b.23-b* a *b.23-d* e *b.24*

$$\text{b.27-h, i} \quad n_i, Nch_{ik} \geq 0, \quad \forall i, k; \quad y_{js} \in \{0,1\}, \quad \forall j, s$$

O modelo *batch\_milp\_fsmz* apresenta resultados concludentes quando comparado com o anterior modelo contínuo, *batch\_nlp\_fsmz*, pois além da remoção das não linearidades, a discretização dos volumes permite uma abordagem realista e cujos benefícios estão referenciados.

Esta formulação é generalizada considerando a existência de processos múltiplos (em paralelo) e negligenciando os tempos de limpeza, para o que se introduz o seguinte grupo de restrições em substituição das restrições em *b.24*:

$$\text{b.28} \quad \sum_{i=1}^N n_i \tau_{ij} \leq H \cdot \sum_{s=1}^{NS(j)} \sum_{p=1}^{NP(j)} p(j) y_{jsp}, \quad \forall j$$

De certo modo, generaliza-se o modelo anterior incorporando uma nova dimensão do problema: o número de processos  $p(j)$  implementados em cada estágio. Ajustando a formulação, Voudouris e Grossmann [1992] desenvolveram o modelo *batch\_milp\_fmmz*, considerando: ambiente multiproduto, processos múltiplos em cada estágio; campanhas com produtos múltiplos; política de espera nula (*Flowshop/ Multiple Machine/ MPC/ ZW*).

**Modelo *batch\_milp\_fmmz*:**

$$\text{b.29-a} \quad [\min] \quad z = \sum_{j=1}^M \sum_{s=1}^{NS(j)} \sum_{p=1}^{NP(j)} c_{jst} y_{jst}$$

sujeito a

$$\text{b.29-b} \quad S_{ij} Q_i \sum_{s=1}^{NS(j)} \sum_{p=1}^{NP(j)} \frac{y_{jst}}{dv_{jst}} - n_i \leq 0, \quad \forall i, j$$

$$\text{b.29-c} \quad \sum_{s=1}^{NS(j)} \sum_{p=1}^{NP(j)} y_{jst} = 1, \quad \forall j$$

$$\text{b.29-d} \quad \sum_{i=1}^N n_i \tau_{ij} - H \cdot \sum_{s=1}^{NS(j)} \sum_{p=1}^{NP(j)} p(j) y_{jsp} \leq 0, \quad \forall j$$

Relações *b.23-b* a *b.23-d*

$$\text{b.29-e, f} \quad n_i, Nch_{ik} \geq 0, \quad \forall i, k; \quad y_{jsp} \in \{0,1\}, \quad \forall j, s, p$$

Ainda no dimensionamento óptimo de unidades multiproducto, Voudouris e Grossmann [1993] supõem que: **i)** a sequência de processamento é similar para todos os produtos (*flowshop*); **ii)** em cada estágio apenas se implementa só processo com dimensão normalizada; **iii)** o tempo de produção de cada lote é independente da quantidade. Neste enquadramento, entendem tratar do sequenciamento do ciclo, bem como avaliar economicamente os inventários. Para tal, utilizaram o critério do VLA, considerando os seguintes parâmetros,

*CI* – custo de investimento;

*R* – montante da venda dos produtos, líquido de materiais e de utilidades;

*CO* – custo operacional, envolvendo custos de inventários e de preparação dos ciclos;

*intx* – taxa de actualização;

*Ny* – número de anos de vida do projecto;

*tx* – taxa de impostos;

*mint* – custo de preparação, associado a cada repetição do ciclo de produção;

*Ncyc* – número de ciclos de sequenciamento;

*Hcyc* – horizonte temporal dos ciclos de sequenciamento;

*h<sub>i</sub>* – custo unitário de inventário do produto *i*;

e obtendo:

$$\text{b.30} \quad NPV(CI, CO) = -CI + (R - CO)(1 - tx) \cdot \frac{1 - (1 + intx)^{Ny}}{intx \cdot (1 + intx)^{Ny}}$$

$$+ \frac{CI}{N_y} \cdot tx \cdot \frac{1 - (1 + intx)^{Ny}}{intx \cdot (1 + intx)^{Ny}}$$

O custo de inventários e de preparação dos ciclos estão incorporados na parcela respeitante aos custos operacionais,

$$\text{b.31} \quad CO = mint \cdot Ncyc + \sum_{i=1}^N h_i \left( \frac{Q_i}{2} \right) (Hcyc - t_i)$$

e o custo de investimento é descrito pela totalidade dos custos dos equipamentos normalizados:

$$\text{b.32} \quad CI = \sum_{j=1}^M \sum_{s=1}^{NS(j)} (\alpha_j \cdot dv_{js}^{\beta_j}) y_{js} = \sum_{j=1}^M \sum_{s=1}^{NS(j)} c_{js} y_{js}$$

Introduz-se uma nova variável, o tempo desagregado do horizonte parcelar de sequenciamento de ciclo  $e_{js}$  (resultante do produto cruzado  $Hc \cdot y_{js}$ ), o qual deve observar a igualdade de agregação:

$$\text{b.33} \quad Hc = \sum_{s=1}^{NS(j)} e_{js}, \quad \forall j$$

Caracteriza-se esta variável como semicontínua, pois o tempo desagregado do horizonte de ciclos  $e_{js}$  respeitará o limite superior definido pelo horizonte  $H$ , além de que o valor desagregado será nulo se a respectiva variável binária  $y_{js}$  também se anular:

$$\text{b.34} \quad e_{js} \leq H y_{js}, \quad \forall j, s$$

O número de ciclos necessários para satisfazer as procuras de todos os produtos  $i$  e em cada estágio  $j$ , não ultrapassará o número de ciclos disponível (para cada produto e ao longo do horizonte parcelar de sequenciamento):

$$\text{b.35} \quad \frac{S_{ij} Q_i}{H} \sum_{s=1}^{NS(j)} \left( \frac{e_{js}}{dv_{js}} \right) \leq n_i, \quad \forall i, j$$

A definição do tempo disponível para a produção de cada produto  $i$ , utilizando apenas valores respeitantes ao último estágio de produção  $M$ , decorre de:

$$\text{b.36} \quad t_i = n_i \tau_{ij'} + \sum_{k=1}^N SL_{ikj'} Nch_{ik}, \quad j' = M$$

Especificando a relação de igualdade na restrição de eliminação de subciclos no sequenciamento, o modelo determinará uma solução que observe a produção em campanhas de produto único (estabelecendo-se que o número de transições de produção entre o mesmo produto é o número máximo possível, ou seja, o número de ciclos a menos de uma unidade, a única opção de sequenciamento é a produção sucessiva desse mesmo produto, em campanha SPC)

$$\text{b.37} \quad Nch_{ii} = n_i - 1, \quad \forall i$$

Pretende-se também que a partição do horizonte  $H$  seja efectuada segundo um número inteiro  $c$  de períodos de sequenciamento parcelares  $Hc$ , para o que haverá de desagregar este horizonte parcelar  $Hc$  através das variáveis  $Hcyc_c$ ; estas variáveis do horizonte parcelar desagregado devem verificar as igualdades de agregação, de cariz parcelar e de cariz total,

$$\text{b.38} \quad \sum_{c=1}^{NC} c \cdot Hcyc_c = H$$

$$\text{b.39} \quad \sum_{c=1}^{NC} Hcyc_c = Hc$$

O limite superior lógico conjugado com a soma disjuntiva das variáveis binárias associadas garante que apenas uma destas variáveis desagregadas não se vai anular (assumindo então o valor do próprio horizonte parcelar,  $Hc$ ):

$$\text{b.40} \quad Hcyc_c \leq r_c H, \quad \forall c$$

**b.41** 
$$\sum_{c=1}^{NC} r_c = 1$$

As relações anteriores conjugam-se no modelo *batch\_milp\_fssi*, o qual visa a maximização do VLA do sistema de produção descontínua, atende aos custos de investimento, aos custos operacionais e aos custos incorridos com inventários, pelo que também considera um elevado número de variáveis e de restrições.

**Modelo *batch\_milp\_fssi*:**

**b.42-a** 
$$[max] \quad NPV(CI, CO)$$

sujeito a

Relações *b.31—b.36*

**b.42-b** 
$$\sum_{s=1}^{NS(j)} y_{js} = 1, \quad \forall j$$

Relações *b.23-b* e *b.23-c*

**b.42-c** 
$$Nch_{ii} = n_i - 1, \quad \forall i$$

**b.42-d** 
$$\sum_{i=1}^N \left[ n_i \tau_{ij} + \sum_{k=1}^N SL_{ikj} Nch_{ik} \right] \leq Hcyc, \quad \forall j$$

Relações *b.38—b.41*

**b.42-e, f** 
$$n_i, t_i, Hc, Hcyc_c, e_{js}, Nch_{ik} \geq 0; \quad r_c, y_{js} \in \{0;1\}, \quad \forall i, j, k, s, c$$

A presente formulação permite balancear os encargos com a armazenagem decorrente de horizontes alongados, os quais dividem o horizonte global num menor número de partições; na aproximação contrária, se o horizonte temporal comportar um maior número de partições e ciclos de sequenciamento mais curtos, os níveis de armazenamento poderão ser inferiores, incorrendo-se em menores custos de inventário, bem como as capacidades de armazenagem também poderão ser inferiores. Porém, o aumento do número de transições reduz o tempo disponível para produção, pelo que haverá necessidade de implementar processos de maior capacidade, de forma a satisfazer as quantidades procuradas. Para ciclos de sequenciamento diminutos, o custo do investimento tende a aumentar pois os equipamentos de processo são relativamente mais caros que os de armazenagem, ocorrendo ainda a diminuição dos custos —operacionais e de investimento— relativos aos inventários (e vice-versa).

Também, foi imposto que o número de ciclos de sequenciamento *Ncyc* deve ocorrer segundo um número inteiro *c*, para que não seja interrompida a produção no último ciclo do horizonte. De facto, estima-se que a perda de produção, originada pelo arredondamento de um número fraccionário de ciclos ao número inteiro imediatamente inferior, possa ser de relevo. A introdução de um limite mínimo para o horizonte parcelar *Hc*, com a delimitação máxima dos de ciclos de sequenciamento *c*, condiciona o nível (mínimo) dos inventários pois a acumulação de materiais é

exigida ao longo do ciclo de sequenciamento; assim, tais constrangimentos conduzem à existência de um *safety stock*, de utilidade inestimável perante a incerteza e a flutuação das procuras.

Um dos factores primordiais para o tratamento do problema é a consideração dos tempos de limpeza nas diversas transições, que são usualmente incorporados no cálculo das folgas *SL*: para valores diminutos dos tempos de limpeza, o número de transições pode elevar-se e conduzir a uma superior utilização do sistema, associado ao maior número de ciclos de sequenciamento de menor duração, e com a diminuição dos custos relacionados com inventários; o aumento dos tempos de limpeza conduz à diminuição geral do tempo disponível para produção, com a necessidade de maiores investimentos em unidades de maior dimensão, decorrendo daí a diminuição do objectivo económico, *VLA*; por sua vez, o aumento da duração do ciclo de sequenciamento, causará os devidos custos em termos de armazenagem de materiais.

Salienta-se que os efeitos descritos foram verificados perante um sistema de processo único, a operar em campanha de produto único sem capacidade de armazenagem intermédia. No intuito de analisar a influência dos aspectos mais significativos para o projecto do sistema, onde para além de considerar as referidas interacções dos tempos de limpeza e dos custos de inventário, Voudouris e Grossmann [1993] adoptam uma perspectiva mais detalhada do sistema ao buscarem otimizar a sua configuração global:

- no primeiro nível de decisão, tratam a alocação de tarefas e equipamentos;
- no segundo nível, configuram as diversas implementações de unidades em paralelo (individuais ou agrupadas, operando em fase ou desfasadamente);
- no terceiro, através da especificação dos pontos intermédios para introdução das unidades de armazenagem, definem os estágios de produção como grupos de processos em operação de espera nula.

Neste enquadramento, encara-se não só um problema de projecto (com decisões quanto à selecção e dimensionamento de unidades normalizadas, para produção e armazenagem) e sequenciamento de produção (com temporização e sequenciamento óptimos do ciclo de produção, considerando os encargos com os inventários) do sistema *batch*, mas também se pretende incorporar aquelas decisões de síntese. Então, desenvolveram um modelo MINLP extenso e não-convexo, que através de linearizações exactas e rigorosas foi reformulado em modelo MILP, para promover o método de resolução e a optimalidade da solução.

A reformulação direccionada ao modelo MILP considerou: a introdução de variáveis binárias para referência das configurações alternativas em cada estágio de produção, de restrições respeitantes à normalização de dimensões, de variáveis e restrições relativas à operação em fase ou desfasada; a introdução de diversos grupos de variáveis, de carácter binário ou semicontínuo, para eliminação de produtos cruzados, assim como as restrições de equivalência respeitantes; a introdução de variáveis binárias para discretização das dimensões disponíveis para as unidades de armazenagem, bem como o tratamento adequado à remoção dos produtos cruzados.

Ora, como o modelo MINLP predecessor incluía já numerosos grupos de restrições (definições dos custos operacionais e de inventário, dos custos de investimento; delimitações máximas às capacidades de produção e de armazenagem, rigorosas ou de natureza heurística; restrições relacionando o horizonte global e o

horizonte parcelar do ciclo de sequenciamento; relações conducentes ao sequenciamento óptimo do ciclo de produção, por aproximação ao TSP agregado, através de desigualdades de carácter lógico e eliminatório), obtém-se um modelo MILP de grande dimensão, cuja resolução é caracterizada por um elevado grau de dificuldade: um sistema *batch* para a produção de 3 produtos, constituído por 3 estágios sucessivos (mistura, reacção, separação), considerando 4 configurações alternativas em cada estágio (processo único; dois processos em paralelo, agrupados e operando em fase; dois processos em paralelo, constituindo dois grupos desfasados; dois grupos desfasados, cada um constituído por dois processos a laborar em fase), exige tempo de cálculo da ordem de  $10^3$  segundos.

A proposta de resolução deste tipo de problemas engloba a decomposição do sistema, segundo diferentes configurações e cuja avaliação decorre em sequência. O método de decomposição explora as condições lógicas do modelo de grande dimensão, através da fixação de variáveis referentes a decisões sobre a configuração do sistema, sendo gerados subproblemas onde são eliminadas numerosas restrições disjuntivas. A resolução de cada um destes subproblemas é bem mais simples do que a do problema original, e além de produzir “cortes” para a função objectivo, também permite reduzir significativamente: o tempo de cálculo; o número de nós enumerados no *branch-and-bound*; o número de iterações de LP efectuadas.

Dadas as dificuldades de modelação e resolução verificadas, Voudouris e Grossmann [1996] optam por direccionar os seus esforços para outros enquadramentos, como o de dimensionamento e sequenciamento em ambiente *jobshop* (*multipurpose*).

#### 2.2.4 Outros enquadramentos

Nesta secção são descritos sucintamente outros trabalhos de interesse, nomeadamente, quanto à abordagem do sequenciamento através da consideração de redes, de estado-tarefa (*state task network*, STN) ou recurso-tarefa (*resource task network*, RTN), ao enquadramento do dimensionamento de processos descontínuos sob incerteza, bem como se revêem alguns trabalhos que apresentam técnicas de resolução alternativas. Deste modo, atendendo à limitada literatura sobre o dimensionamento e sequenciamento de processos descontínuos *flowshop*, pretende contribuir-se para um enquadramento com os trabalhos contemporâneos aos dos modelos de Voudouris e Grossmann [1992, 1993] descritos.

Kondili *et al.* [1993] apresentam um modelo MILP generalizado, aplicável ao sequenciamento de uma grande variedade de processos descontínuos, em ambiente multiproducto ou multiobjectivo. Neste caso, a requerida produção de curto prazo é definida em função da procura em diferentes períodos temporais, a qual assume a forma realista de data de entrega, e os tempos de transição são considerados negligenciáveis. Para operar o sistema descontínuo em modo de campanha, deverá melhorar-se a formulação de maneira a obter estruturas regulares de produção e a definir-se o sequenciamento dos ciclos. Este modelo assume uma representação em rede estado-tarefa que apresenta dois tipos de nós: nós de estado, correspondentes a componentes materiais (matérias-primas, compostos intermédios, produtos finais); nós de tarefa, correspondentes a fases processuais ou actividades requeridas.

Cada rede deste tipo estado-tarefa permite o tratamento de um só produto, onde produtos diferentes requerem o tratamento de redes também diferentes. Por conseguinte, a consideração de produtos múltiplos exige o tratamento de uma multiplicidade de redes estado-tarefa. Note-se que, neste enquadramento: a alocação

equipamento-tarefa é efectuada através da resolução óptima do modelo, bem como se determina as quantidades de produção  $B_i$  dos lotes; a ocorrência de armazenagem de materiais verifica-se através da acumulação no respectivo nó-estado; define-se a utilização óptima da rede de utilidades associada. A representação de partilha de materiais e de produtos intermédios em linhas de produção cruzadas, a mistura ou a separação de lotes de produção, a reciclagem de materiais, introduzem-se de forma implícita na formulação desenvolvida.

Barbosa-Póvoa e Macchietto [1994] formulam uma superestrutura de rede estado-tarefa como um modelo MILP que optimiza: as conexões entre as unidades de processo; o dimensionamento e os custos das linhas de transferência; as políticas de armazenagem e o dimensionamento respeitante. Pode assumir-se uma abordagem simultânea do dimensionamento e do sequenciamento, bem como pode ser seguida uma abordagem hierárquica (optimização do equipamento principal negligenciando os custos das conexões, seguida da configuração estrutural das conexões para o equipamento definido no passo anterior). Baseado neste modelo, Barbosa-Póvoa e Pantelides [1997] utilizam uma formulação em rede recurso-tarefa que de uma forma mais directa permite o tratamento de problemas complexos de sequenciamento de processos descontínuos, nomeadamente, considerando interacções entre aspectos de produção e de distribuição. Obviamente que a consideração de estruturas complexas com um elevado número de elementos origina problemas MILP de grande dimensão, com a dificuldade associada. Barbosa-Póvoa e Pantelides [1999] utilizam uma estratégia de decomposição, que envolve a resolução iterativa de um problema mestre (selecciona o equipamento principal) e um subproblema (optimiza o dimensionamento desse equipamento, selecciona unidades de armazenagem e trata do sequenciamento).

Wellons e Reklaitis [1989] apresentam uma formulação para o problema do dimensionamento de processos descontínuos em ambiente de incerteza, onde distinguem as restrições a satisfazer em “duras” (*hard*) e “moles” (*soft*), penalizando não só a violação destas últimas, como também a perda esperada de lucro devido à não-satisfação da procura. Assumem a ocorrência de expansão de capacidades faseada, sob a forma de unidades idênticas de equipamento a operar de forma desfasada, tendo em vista a cabal resposta às variações de longo prazo nas procuras, evitando assim a necessidade de sobredimensionamentos excessivos e que se traduz numa efectiva redução de custos se o previsível aumento de procura não se verificar.

Subrahmanyam *et al.* [1994] estudam o dimensionamento de processos descontínuos onde cada cenário discreto de procura é uma ocorrência ou realização particular dos parâmetros incertos, descrito através duma probabilidade conhecida previamente. O problema MILP é decomposto num problema onde se trata o dimensionamento dos equipamento e a sua afectação às tarefas, mas onde o sequenciamento detalhado é negligenciado: por exemplo, num horizonte de 5 anos, não é necessário considerar o sequenciamento integral das operações, pelo que o horizonte é dividido em períodos parcelares de tempo, em cada um dos quais será resolvido um problema de sequenciamento. Na sequência desta formulação, Subrahmanyam *et al.* [1996a] desenvolvem um algoritmo de decomposição baseado na relação teórica que derivam entre o problema agregado e a formulação de sequenciamento detalhado. Com base em tal algoritmo, associado a um procedimento heurístico que auxilia o desempenho computacional, conseguem obter soluções óptimas para o problema geral, aplicando o referido algoritmo a diversos casos industriais. Ainda, no âmbito da resolução de MILPs de grande dimensão,

Subrahmanyam *et al.* [1996b] burilam a implementação de problemas de optimização de casos industriais em plataformas de computação distribuída. Aplicam um método de *branch-and-bound* paralelizável, obtendo soluções economicamente equivalentes, mas diferentes em termos das configurações a implementar.

Ierapetritou e Pistikopoulos [1995] enquadraram o problema do dimensionamento de processos descontínuos sob incerteza das procuras como 2SSP, considerando conhecida a forma da distribuição de probabilidades. Formularam um modelo não linear e não convexo de grande dimensão, mas que apresenta uma estrutura especial que lhe permite um tratamento numérico eficiente pelo algoritmo de optimização global (*Global Optimization, GO*). Ierapetritou e Pistikopoulos [1996] alargaram a abordagem da incerteza, não só considerando-a nos parâmetros externos da procura, mas também nos parâmetros internos dos processos. A distribuição multivariada de probabilidade é tratada por integração gaussiana, sendo possível considerar diferentes distribuições de probabilidade, ou mesmo diferentes modos de produção, mas com o inconveniente de ser acompanhado de um elevado número de variáveis e de restrições não-convexas.

Petkov e Maranas [1998] introduzem uma nova abordagem para o dimensionamento de unidades multiproducto descontínuas, considerando: campanhas de produto único (SPC); incerteza na procura dos produtos, assumindo distribuições de probabilidade normais; possibilidade de diversos processos, de igual dimensão, em cada estágio. Desta forma, o problema formulado inicialmente como um 2SSP é transformado num problema de MINLP determinístico e convexo, o qual permite a resolução de instâncias de grande dimensão, com um elevado número de procuras incertas. Revela-se também a equivalência entre a penalização de rupturas de produção e a imposição de elevados valores para a probabilidade de satisfação da procura, aspecto este que é crucial para a selecção óptima do número de processos e das respectivas capacidades, em cada estágio.

A complexidade dos problemas de dimensionamento e sequenciamento de processos descontínuos conduz à adopção de metodologias alternativas, tais como as dos descritos procedimentos evolucionários em Xia e Macchietto [1997] e Tan e Mah [1998]. Também Pekny e Miller [1991] recomendam a utilização de procedimentos não exactos, heurísticos, quando não se dispõe de um método satisfatório em face do problema existente. Estes últimos referidos autores, tratam o problema de sequenciamento *flowshop* com política de espera nula, tanto de forma exacta, através de um algoritmo de *branch-and-bound*, como de forma heurística, através de procedimentos de permutas e de *simulated annealing*. Então, avaliam comparativamente o desempenho relativo daqueles métodos, exacto e heurísticos.

## Sinopse

Neste Capítulo, analisam-se casos de optimização direccionados para a indústria de processos químicos e que privilegiem a abordagem por Programação Matemática, recorrendo à MILP. Dada a vastidão do âmbito visado, conjugado com o detalhe dos modelos matemáticos de interesse para este estudo, foram visados especificamente dois casos da literatura: o planeamento de processos, seguindo os desenvolvimentos de Sahinidis *et al.* [1989]; e o dimensionamento e sequenciamento de processos descontínuos, como em Voudouris e Grossmann [1992].

Também, revêem-se casos de optimização dirigidos à aplicação industrial directa, constatando-se a sua escassa presença nas revistas científicas, nomeadamente, nas mais conceituadas da Engenharia Química. Esta dificuldade resultará, porventura de objectivos algo antagónicos, discrepantes nas estratégias das instituições científicas e das empresas industriais, ou em aspectos de circulação de informação.

O primeiro caso de optimização dirige-se ao planeamento dos investimentos em sistemas de processos químicos, iniciando-se por aspectos diversos e de carácter aplicado, e dirigindo-se ao tratamento da incerteza inerente, nomeadamente, quando se considera um longo horizonte temporal. Enquadra-se o problema, analisando-se a qualidade da modelação matemática por MILP dos modelos de planeamento de processos focados, constatando-se uma evolução contínua e duplamente direccionada:

*i)* por um lado, pretende-se que a formulação desenvolvida se aproxime o mais possível da realidade, quer introduzindo diferentes modos de produção para permitir a flexibilidade da rede de processos, quer tratando da incerteza intrínseca dos valores previsionais do modelo;

*ii)* por outro lado, logo na fase de modelação incorpora-se um conjunto de características que favorecem as técnicas de resolução, como sejam a reformulação da subestrutura de ELS, a adição de planos de corte, assim como a introdução de procedimentos heurísticos e a investigação analítica das propriedades destes procedimentos.

O segundo caso atenta no dimensionamento e sequenciamento de processos descontínuos, ou seja, consiste no problema que ocorre quando se pretendem contemplar aspectos —específicos e relevantes— do sequenciamento da produção ainda na fase de dimensionamento. Enquadra-se apenas o ambiente multiproduto (*flowshop*), não se visando aqui o ambiente multiobjectivo (*jobshop*). Atendendo aos diversos modelos matemáticos descritos neste âmbito, observa-se:

*i)* a sucessiva incorporação de graus de liberdade —ampliando o número de processos por estágio, permitindo diferentes políticas para operação e armazenagem, e englobando os encargos económicos (arranques, operações, inventários)—, ou seja, as modelações vão generalizando o leque das opções, até pela combinatoriedade das soluções, pelo que o espaço de procura dos problemas vai assumindo fronteiras mais alargadas;

*ii)* a focalização da modelação em aspectos de representação em redes para estabelecer o sequenciamento da produção, formulando-se restrições de carácter lógico, tornando-se convexa a formulação por discretização ou mesmo aplicando transformações logarítmicas; nesta vertente, procura-se constranger o espaço de procura, aplicando cortes específicos, nomeadamente, com fundamento no problema de dimensionamento de lotes, ou com base em estruturas especiais (SOS) para a solução óptima;

*iii)* o menor impacto deste segundo caso (relativamente ao caso de planeamento de processos descrito), na literatura publicada e em termos industriais, sendo também referidos outros enquadramentos —como representações em redes, técnicas alternativas, heurísticas e métodos de decomposição, ou mesmo a implementação em computação distribuída.



# Implementação computacional

Com o desenvolvimento dos equipamentos informáticos e dos *packages* de optimização, os modelos de Programação Matemática, mesmo os de grande dimensão, podem ser implementados e processados, disseminando a sua aplicação. Contudo, dada a complexidade computacional inerente aos modelos de MILP como os usuais em problemas de sequenciamento, o tratamento deste tipo de modelos permanece ainda como um desafio. Assim, para um modelo contendo variáveis binárias, o crescimento exponencial do número de soluções constitui o elemento principal para o aumento de dificuldade da resolução exacta (óptima).

De uma forma geral, o desenvolvimento modular e a programação estruturada constituem os pontos fulcrais da programação em linguagem de alto nível, envolvendo procedimentos secundários, como sub-rotinas e funções. O desenvolvimento modular consiste na produção de subprocedimentos ou módulos, que podem ser desenvolvidos e testados separadamente, o que permite tratar e fasear tarefas de elevada dificuldade técnica. O desenvolvimento modular *top-down* permite a identificação dos módulos, iniciando-se pelos objectivos mais gerais do programa, seguindo-se a divisão sucessiva nas suas funções principais, subdividindo-se ainda em segmentos modulares mais específicos. Esta abordagem permite uma compreensão mais rápida e facilita a verificação e a manutenção dos procedimentos.

Em Engenharia Química, a abordagem por módulos permite desenvolver enquadramentos gerais para síntese de processos, como indicam Daichendt e Grossmann [1997]: trata-se da optimização simultânea de sistemas complexos de grande dimensão, exemplificados por módulos de reacção química, separação por destilação, e rede de permutadores de calor. Numa fase de aproximação inicial, os diversos módulos podem utilizar uma representação termodinâmica, matematicamente mais simples, assumindo depois configurações mais complexas com o desenrolar do cálculo.

A fiabilidade da implementação computacional é um factor básico das aplicações, mas em ambiente de engenharia devem ser utilizados níveis de rigor relacionados com a gravidade das consequências e o contexto. A validação da

implementação tem por objectivo verificar se o nível de confiança nos resultados é aceitável, constituindo-se até como uma etapa crítica na modelação do problema.

Em termos computacionais, devem ser corrigidos os erros de sintaxe aquando da compilação, os erros de construção que ocorrem na fase de ligação, os erros de processamento que aparecem na fase de execução, ou mesmo os erros de codificação lógica do algoritmo (ciclos infinitos, erros nas fórmulas). Os programas devem ser verificados com uma sucessão de testes, garantindo inicialmente a fiabilidade de cada módulo (testes modulares), certificando a integração dos módulos no programa (testes de desenvolvimento), procedendo a testes (gerais) ao sistema global —com dados típicos, com dados válidos não usuais, com dados inválidos—, finalizando-se com a aplicação de testes (operacionais) de execução em ambiente realista. Usualmente, a implementação das bibliotecas matemáticas comerciais é verificável de forma automática, pois são fornecidos ficheiros executáveis que utilizam as bibliotecas em cálculos de teste, para comparar os resultados assim obtidos com os valores de referência.

Em termos de aplicação a um caso real, para assegurar a qualidade dos resultados finais, é conveniente confirmar: a validade do modelo desenvolvido e da sua coerência; a qualidade dos dados utilizados; a operacionalidade dos valores calculados; a correspondência entre os seus comportamentos e os efeitos reais. Para obter soluções fiáveis através do modelo interessam medidas para avaliação do desempenho, ou mesmo valores de referência para tais medidas, por exemplo, através de dados históricos. Concretamente, refere-se aqui o usual teste retrospectivo, que utiliza o registo histórico para certificar o comportamento e a aplicação correcta do modelo perante uma situação anterior. Assim, pode comparar-se a solução decorrente do modelo matemático com a solução implementada, o que servirá de indicador quanto às possíveis melhorias a atingir com a utilização do modelo. Esse teste permitirá obter informações adicionais, quer sobre aspectos que não estarão ainda implementados no modelo, quer sobre efeitos negligenciados, ou mesmo sobre as predições realizadas perante acções alternativas.

A linguagem Fortran [Adams *et al.*, 1992], construída para processar fórmulas matemáticas —possuindo funções matemáticas intrínsecas (números complexos, cálculo vectorial e matricial), permitindo a extensão da precisão de cálculo e atendendo à sua estruturação modular—, constitui-se como uma linguagem preferencial para a computação nas áreas das ciências e engenharia. Para desenvolvimento de aplicações informáticas é muito utilizada a linguagem C, mas, pese embora as suas características de flexibilidade e de acesso de baixo nível ao *hardware*, constata-se como sendo de qualidade inferior ao Fortran em termos de cálculo numérico.

A utilização de aplicações (*packages*) exige apenas o conhecimento dos fundamentos dos métodos, mas muitos problemas não são resolúveis directamente através dessas aplicações, havendo que proceder a desenvolvimentos específicos. Logo, é conveniente conhecer tanto as capacidades, como as limitações dos *packages*. Frequentemente, a maneira mais rápida de os utilizar consiste em encontrar um exemplo ilustrativo, normalmente fornecido com a documentação, e sobre o qual se procederá às modificações necessárias.

As rotinas matemáticas disponíveis nos *package* são desenvolvidas de forma a não serem afectadas pelos erros de *underflow* e *overflow*, retornando informação associada a erros nos dados, na tipologia dos argumentos, ou nos aspectos dimensionais. A informação sobre o tipo e severidade do erro ocorrido é usualmente

especificada através dum valor numérico, onde uma execução sem quaisquer erros apresenta um código de erro nulo (por exemplo, IFAIL = 0, segundo a NAG).

Tem-se dirigido a implementação computacional para a construção de modelos matemáticos e “orientação para objectos” (*object oriented*), no intuito de simplificar a incorporação de modelos de MP nos sistemas de suporte à decisão, tão frequentes na indústria. Um ambiente de modelação é aqui utilizado, GAMS [Brooke *et al.*, 1992], embora seja possível encontrar outros, como por exemplo o LINDO [Schrage, 1986] (para LP ou MILP), mas que não possui a capacidade do ambiente GAMS para a definição implícita de conjuntos, parâmetros, equações, o que dificulta a utilização do LINDO para modelos de grandes dimensões.

Nas secções seguintes, apresentam-se as implementações efectuadas no âmbito dos problemas em estudo, enquadrando quer o planeamento de processos (*process planning*), quer o dimensionamento e sequenciamento de processos descontínuos (*design and scheduling of batch processes*). Descrevem-se aqui aspectos numéricos e computacionais que suportam os desenvolvimentos em capítulos subsequentes, no sentido de congregar tais aspectos num só capítulo e assim permitir uma visão alargada.

Ao problema de planeamento de processos dedica-se a primeira secção, descrevendo-se a implementação em Fortran, através da qual se recorre à biblioteca matemática NAG para a optimização dos modelos MILP. Implementam-se modelos de índole determinística, estocástica (em duas fases), com robustez, sendo ainda apresentadas no Apêndice 3 diversas aplicações de suporte, como a definição do espaço de procura, a metodologia de amostragem, e outras.

Na segunda secção —constatada, a dificuldade da modelação em Fortran no caso anterior —, tratam-se os modelos de dimensionamento e sequenciamento em ambiente GAMS, o qual permite “optimizar” o próprio processo de codificação. Implementam-se modelos determinísticos e generaliza-se oportunamente (conforme Capítulo 5) para um enquadramento estocástico, bem como se trata uma modelação robusta associada. Conjuga-se com o desenvolvimento de heurísticas (descrito no Capítulo 4) em Fortran, destinadas a melhorar a resolução de referência dos *solvers* disponíveis (tendo-se utilizado principalmente o OSL da IBM associado ao GAMS), bem como se pormenorizam algumas técnicas de delimitação do espaço de procura e de fixação ou inicialização de variáveis, que tornam a resolução mais eficiente.

### **3.1 Aplicação a modelos de planeamento de processos (Fortran)**

Descreve-se a implementação em linguagem Fortran de modelos de planeamento de processos, adequando o código aqui desenvolvido de forma a utilizar rotinas da biblioteca matemática NAG específicas para a resolução dos problemas MILP correspondentes, bem como se testa a implementação efectuada. Este conjunto de acções suporta o desenvolvimento dos modelos estocásticos associados, que visam promover a robustez da solução e do modelo perante a incerteza das instâncias probabilísticas, assim como se aproveita tal implementação para aplicar esquemas de decomposição, onde se codificam os subproblemas Mestre e de recurso (segundo um esquema de decomposição de Benders).

Dado que se pretende a conformação da função objectivo, da matriz das restrições, da delimitação das variáveis contínuas e discretas, segundo as rotinas disponíveis na biblioteca NAG, verifica-se que o processo de implementação se torna bastante moroso e exigente. Logo, interessa sistematizar a metodologia de implementação, de forma a acelerá-la e a aproximar algumas características dos ambientes de modelação.

A validação da implementação decorre de modos diversos, como sejam, a verificação directa dos resultados apresentados na literatura, a análise de variações extremas, a justificação da evolução da função objectivo em resposta a perturbações de parâmetros sensíveis do problema, ou através da verificação de resultados de referência.

### 3.1.1 Modelo determinístico

Apresenta-se a metodologia seguida na implementação do modelo determinístico para planeamento de processos *plan\_milp\_lrp*, descrito na Sec. 2.1. O programa principal organiza o algoritmo e aplica as rotinas desenvolvidas, implementando o modelo de forma a utilizar a biblioteca NAG. As rotinas de maior interesse no módulo principal (*main*) correspondem às principais acções a efectuar:

LEITUR – leitura de parâmetros (READIN) e dados do problema (READAT);

INIT – inicialização de vectores e matrizes do método de resolução;

FUNOBJ – definição da função objectivo do modelo;

MATRIX – definição das restrições do modelo;

*naglib* – resolução do MILP, pela rotina H02BBF da NAG;

RESULT – apresentação dos resultados.

O módulo principal lê os parâmetros e dados do problema, compõe e inicializa os vectores e matrizes necessários; então, especifica os coeficientes da função objectivo, para depois definir a matriz das restrições, com as respectivas delimitações; a resolução do problema MILP processa-se através da rotina H02BBF da biblioteca NAG, sendo os resultados apresentados no formato que se deseja.

Neste módulo, é também alocado o espaço de trabalho necessário, não só relativamente ao modelo implementado, mas também no respeitante ao método de resolução. O número de variáveis  $n$  e de restrições  $m$  (*leading dimension array, lda*), a sua soma  $lb$ , são de interesse específico, sendo estes parâmetros directamente obtidos da cardinalidade dos parâmetros do problema, isto é, do número de períodos de tempo  $NT$ , do número de processos  $NP$ , de compostos químicos  $NC$ , de mercados para compra e venda  $NM$ .

Em código Fortran 77, as variáveis principais do modelo implementado são armazenadas em bloco COMMON e assim transmitidas às diversas rotinas. Contudo, a incorporação de variáveis globais na versão actualizada de Fortran 90 permite, cautelosamente, o fluxo global da informação deste tipo.

A rotina LEITUR (aliás, READIN e READAT) recolhe os parâmetros referentes ao método de resolução (o tipo de optimização, tolerâncias, número máximo de iterações, tipo de procura a efectuar, tipo de comunicação desde a rotina),

os parâmetros e os dados do problema a processar, efectuando-se também a escrita dos valores recolhidos para efeitos de verificação.

A rotina INIT inicializa com valor nulo as variáveis da implementação: os vectores da solução  $X(n)$  e da estimativa inicial  $Xest(n)$ ; os coeficientes da função objectivo  $cvec(n)$ ; o vector indicador  $intvar(n)$  do carácter contínuo ou discreto das variáveis; a matriz  $a(m,n)$  dos coeficientes das restrições; os vectores dos termos independentes das variáveis e restrições,  $bl(n+m)$  e  $bu(n+m)$ , correspondendo respectivamente aos lados esquerdo (*left hand side*, LHS) e direito (*right hand side*, RHS) das desigualdades —em caso de restrição de igualdade, os valores apresentados em  $bl$  e  $bu$  serão iguais para as linhas da matriz de restrições referentes a cada restrição de igualdade. Contudo, atendendo ao pressuposto de todas as variáveis contínuas serem não-negativas, os elementos do vector  $bu(n)$  respeitantes ao RHS da gama de variação das variáveis apresentarão um valor muito elevado (*bigbound*, ou também *bigbnd*):

$$x(n) \geq 0 - bl(n) = 0; bu(n) = bigbound$$

Na rotina FUNOBJ, especificam-se os coeficientes da função objectivo sequencialmente, obtendo-se desde logo o seu enquadramento em termos de colunas da matriz de restrições atribuindo valores de referência aos parâmetros  $jcol$ . Também, torna-se mais eficiente aproveitar os diversos ciclos de atribuição de valores aos coeficientes, para se proceder à identificação das variáveis discretas —por defeito, a atribuição de valor nulo ao vector  $intvar$  identifica como contínuas todas as variáveis do modelo.

As variáveis importantes da rotina FUNOBJ decorrem da função objectivo do modelo, cujos coeficientes correspondem aos elementos do vector:

$cvec(n)$  – coeficientes da função objectivo.

No caso de ocorrência de variáveis redundantes, variáveis estas que não se encontram explicitadas em termos de função objectivo, os respectivos coeficientes serão obviamente nulos, tal como foram inicializados em INIT. Porém, é de interesse proceder à introdução destas variáveis na função objectivo, para serem directamente referenciadas na matriz de restrições. Caso necessário, também aqui se poderá especificar a gama de variação ou fixar com valor constante (*const.*) alguma das variáveis, respectivamente:

$$LB \leq x(jcol) \leq UB - bl(jcol) = LB; bu(jcol) = UB$$

$$x(jcol) = const. - bl(jcol) = const.; bu(jcol) = const.$$

Ainda: dado que a optimização efectuada pela rotina H02BBF se dirige, por defeito, no sentido de minimizar o valor-objectivo, então deve considerar-se o vector  $cvec(n)$  com valores simétricos aos inicialmente atribuídos, pois no problema de planeamento de processos a intenção é a de maximizar o objectivo VLA.

A rotina MATRIX permite especificar a matriz das restrições do modelo a optimizar. Para tal, são utilizadas um conjunto de rotinas secundárias, não só para definição dos coeficientes da matriz  $a(m,n)$  das restrições, mas também para especificação do tipo de relação (igualdade, desigualdade) e dos termos independentes presentes (LHS, RHS),  $bl(m)$  e  $bu(m)$ . Assim, cada uma destas rotinas secundárias especifica um grupo de restrições do modelo *plan\_milp\_lrp* para planeamento de processos:

- GAEXPA – gama possível para as expansões de capacidades, para cada processo em cada período de tempo;
- BCAPAC – balanço às capacidades, para cada processo em cada período;
- LSPROD – limite superior de produção, idem;
- BMASSG – balanço mássico global para cada componente, ao longo de todos os períodos;
- LNEXPA – limite superior ao número de expansões, para cada processo;
- LINVES – limite superior ao investimento, em cada período.

Os parâmetros relevantes na rotina MATRIX, a definir em cada um dos grupos de restrições referidos, são:

$a(m,n)$  – coeficientes da matriz das restrições;

$bl(n+1:), bu(n+1:)$  – termos independentes das restrições (LHS, RHS).

Assim, cada rotina especifica um dos grupos de restrições referidos, pelo que terá de se estabelecer a tipologia das restrições (exemplificando: um duplo ciclo envolvendo todos os  $NT$  períodos e todos os  $NP$  processos), configurar as variáveis de interesse (especificando quais as colunas de referência, através dos parâmetros  $jcol$ ), bem como relacionar essas variáveis nas diversas restrições do grupo. Este último aspecto será específico de cada restrição ou grupo de restrições, pelo que exigirá uma atenção particular do programador.

Para cada linha da matriz  $a(i\text{lin},:)$ , também se deve estipular a respectiva gama de variação (LHS, RHS), sendo tal par de atribuições efectivado no mesmo ciclo de operações em que se especifica a linha da matriz:

$$a(i\text{lin},:) \geq 0 - bl(n+i\text{lin}) = 0; bu(n+i\text{lin}) = \text{bigbound}$$

$$a(i\text{lin},:) \leq 0 - bl(n+i\text{lin}) = -\text{bigbound}; bu(n+i\text{lin}) = 0$$

$$LB \leq a(i\text{lin},:) \leq UB - bl(n+i\text{lin}) = LB; bu(n+i\text{lin}) = UB$$

$$A(i\text{lin},:) = \text{const.} - bl(n+i\text{lin}) = \text{const.}; bu(n+i\text{lin}) = \text{const.}$$

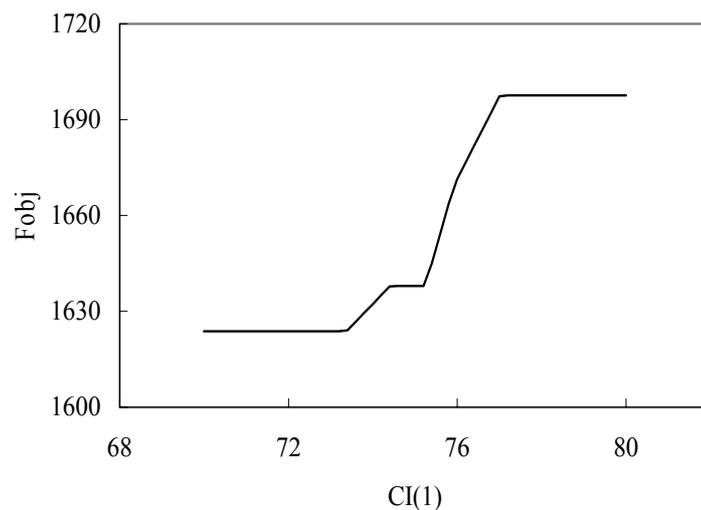
Atente-se no deslocamento da atribuição de  $n$  unidades no par de vectores de delimitação,  $bl(n+m)$  e  $bu(n+m)$ : por imposição da rotina H02BBF, os primeiros  $n$  elementos correspondem à definição da gama de variação das  $n$  variáveis do modelo; apenas depois serão especificados os  $m$  pares de termos independentes (LHS, RHS) relativos às restrições do modelo.

A rotina RESULT apresenta os resultados do cálculo otimizador, utilizando-se a simples forma sequencial —pela qual as variáveis foram definidas em FUNOBJ— para a apresentação dos resultados. Os formatos aplicados podem facilmente ser adequados às finalidades visadas, como para elaboração de figuras, para tratamento estatístico, para análise comparativa de resultados. É conveniente apresentar os parâmetros respeitantes ao método de resolução que são de interesse geral, pois em termos de implementação de MILP é importante conhecer: o tempo de execução; o número de iterações de LP; o número de nós da árvore do *branch-and-bound*; o número de soluções inteiras obtidas; o majorante do desvio no valor-

objectivo; a tolerância para a integralidade das variáveis discretas. Para aplicações de esquemas de cálculo especiais (planos de corte, geração de subproblemas, incorporação de restrições específicas, aplicação de heurísticas), convirá também explicitar os parâmetros que permitam ajuizar da efectividade do esquema implementado.

Esta implementação do modelo determinístico é validada por comparação directa dos resultados obtidos com os valores disponíveis na literatura [Sahinidis *et al.*, 1989]. Atendendo a que as tabelas contendo tais valores apresentavam erros tipográficos (o total das quantidades vendidas superava o total das produções!), contactou-se o investigador principal da comunicação, que disponibilizou os respectivos dados e resultados. O modelo implementado em código Fortran verificou inteiramente o trio de cenários disponibilizado.

Também para validação, analisa-se a evolução da função objectivo em resposta à variação a flutuações de parâmetros sensíveis do problema, nomeadamente, à variação do valor limite de investimento dentro da gama de valores de interesse. A resposta da função objectivo à variação específica da *upper bound* para o custo de investimento no período inicial  $CI(1)$ , na vizinhança de valores sensíveis, apresenta-se na *Figura 3.1*.



**Figura 3.1** Variação da função objectivo com o valor limite  $CI(1)$  para investimento no período inicial.

Analisa-se o primeiro exemplo ilustrativo relatado por Sahinidis *et al.* [1989], onde se considerou um sistema com três processos produtivos, cujo plano de expansão de capacidades é optimizado ao longo de três períodos temporais. Observa-se que, para valores diminutos do custo de investimento inicial  $CI(1)$ , o valor da função objectivo mantém-se constante, pois visa expandir-se a capacidade de produção dos processos 1 e 3, mas apenas no período 2. Com o aumento sucessivo do montante para investimento  $CI(1)$ , poder-se-á expandir a capacidade do processo 1 logo no período inicial, pelo que os níveis de produção poderão acompanhar tal acréscimo de capacidade e o valor objectivo apresenta uma evolução crescente e linear. No entanto, atinge-se um patamar intermédio, onde o aumento do investimento inicial não se repercute sobre a função objectivo que apresenta valor constante, pois a restrição correspondente ao limite de investimento encontra-se

inactiva. Posteriormente, ocorre uma modificação da solução binária, dado que o aumento do montante  $CI(1)$  permitirá a expansão das capacidades dos processos 1 e 3 logo no período inicial. O subsequente aumento do valor objectivo apresenta declives distintos, consistindo: primeiro, na utilização apenas de materiais intermédios produzidos no sistema de processos; depois, também na compra desses materiais intermédios, logo que se exaure a capacidade de produção respeitante. A partir de um certo valor de  $CI(1)$ , verifica-se que o montante para investimento não constitui constrangimento para a expansão do sistema de processos, posto que a função objectivo não apresenta qualquer flutuação.

### 3.1.2 Modelo estocástico com recurso

A partir do modelo determinístico prepara-se o modelo estocástico correspondente ao problema em duas fases, discriminando o subproblema de projecto e a gama de subproblemas de recurso. É necessário definir a métrica ou estimador de desempenho, seleccionar a métrica de variabilidade, especificar penalizações por acomodação de desvios na modelação efectuada.

As rotinas principais, definidas no módulo inicial, efectuem acções similares às desenvolvidas no âmbito do problema determinístico, mas agora com um contexto mais alargado pois que se considera um conjunto de amostras de valores aleatórios. As designações das *nsp* variáveis, parâmetros, rotinas, *etc.*, assemelham-se às designações do problema determinístico associado, mas adicionando o sufixo explicativo “\_SP”.

Assim, na rotina LEITUR\_SP procede-se à leitura de dados para o problema estocástico, que correspondem aos dados do problema determinístico ao qual se associam os parâmetros de incerteza. Assume-se que os dados do problema determinístico correspondem aos valores esperados dos parâmetros — $xmu(nsp)$ — e constituem a primeira amostra do espaço probabilístico possível, sendo desenvolvidas as restantes amostras através dos parâmetros de variabilidade respeitantes: o desvio padrão — $sigma(nsp)$ — e a referência ou tipologia para a distribuição de probabilidades aplicável — $ktipo(nsp)$ . Também é necessário proceder à amostragem estratificada dos valores normalizados para os parâmetros incertos, utilizando a rotina HSS2 (*Hammersley Sampling Sequence*), desenvolvida e adaptada a partir do algoritmo descrito por Diwekar e Kalagnanam [1997]. Aplicando a rotina TRANS\_PDF para efectuar a translação às amostras de valores normalizados, em conformidade com a distribuição de probabilidades explicitada (tipologia, valor esperado, variabilidade e outros parâmetros de interesse), obtém-se o conjunto de amostras que o modelo estocástico irá utilizar para construir os cenários aleatórios. De forma semelhante e aplicando os mesmos parâmetros para definir cada função de densidade de distribuição, obtém-se a probabilidade associada a cada um dos cenários aleatórios através da rotina PDF (*Probability Density Function*). No caso de amostragem de Monte Carlo, com igual probabilidade de ocorrência para os conjuntos amostrais, aplica-se o trivial cálculo da distribuição uniforme.

Na rotina INIT\_SP inicializam-se os vectores e matrizes necessários ao método de resolução — $x\_sp(nsp)$ ,  $xest\_sp(nsp)$ ,  $cvec\_sp(nsp)$ ,  $intvar\_sp(nsp)$ ,  $a\_sp(msp, nsp)$ ,  $bl\_sp(nsp+msp)$ ,  $bl\_sp(nsp+msp)$ — utilizando a forma usual: anulam-se os respectivos valores iniciais, excepto os valores correspondentes às *UB* das variáveis (não-negativas) que tomam o valor de *bigbound*.

Em termos de alocação do espaço de memória e de trabalho, atente-se que se deverá adicionar uma nova dimensão para incorporar os *nr* cenários amostrais, tanto

às variáveis de recurso do problema, como aos parâmetros incertos. Ainda, haverá que adicionar novas linhas e colunas para a matriz das restrições do modelo estocástico, concretamente: linhas para a definição dos valores esperados *def\_csi* e dos desvios, *def\_dsvn* e *def\_dsvp*, bem como das restrições que dependem de parâmetros amostrais, referentes a cada cenário considerado no problema de segunda fase; colunas associadas às variáveis de recurso, para cada cenário.

Na rotina FUNOBJ\_SP trata-se da função objectivo do modelo estocástico, relacionando as variáveis correspondentes ao valor esperado do objectivo VLA—*csi(nr)*—, à variabilidade decorrente dos desvios provocados por flutuações nos custos de recurso —*dsvp(nr)*, *dsvn(nr)*—, aos desvios por subutilização da capacidade instalada para cada processo, em todos os períodos de tempo e todas as amostras —*Zp(np.nt.nr)*—. Os coeficientes da função objectivo são especificados sequencialmente, para obter os valores de referência das colunas da matriz de restrições, *jcol\_csi*, *jcol\_dsvn*, *jcol\_zp*.

Aproveitam-se os ciclos de atribuição existentes nesta rotina para identificação sequencial das variáveis discretas, bem como para referenciar todas as variáveis redundantes, ou as variáveis do problema cuja ponderação em termos da função objectivo estocástica seja nula. Concretamente, encontram-se nesta situação as variáveis de recurso, que correspondem a variáveis que apresentam valores diferentes em cada cenário, mas que não são discriminadas na função objectivo estocástica.

De forma ordenada, na rotina MATRIX\_SP trata-se da definição das restrições do modelo estocástico, onde a submatriz referente ao primeiro cenário coincide com a matriz de restrições do problema determinístico associado:

$$a\_sp(:,n) = a$$

$$bl(nsp:nsp+m) = bl ; bu(nsp:nsp+m) = bu$$

Ainda para esta primeira amostra, constituída pelos valores esperados dos dados do problema, é necessário definir os respeitantes estimadores para o valor esperado, *csi(I)*, e para a variabilidade, *dsvn(I)* e *dsvp(I)*.

A definição das restrições correspondentes às restantes amostras desenrola-se eficientemente em ciclo DO, de modo que a o problema estocástico obtido seja passível de geração por cenários. Assim, para cada cenário, são ciclicamente tratadas as seguintes rotinas:

LSPROD\_SP – limite superior de produção, para cada processo em cada período;

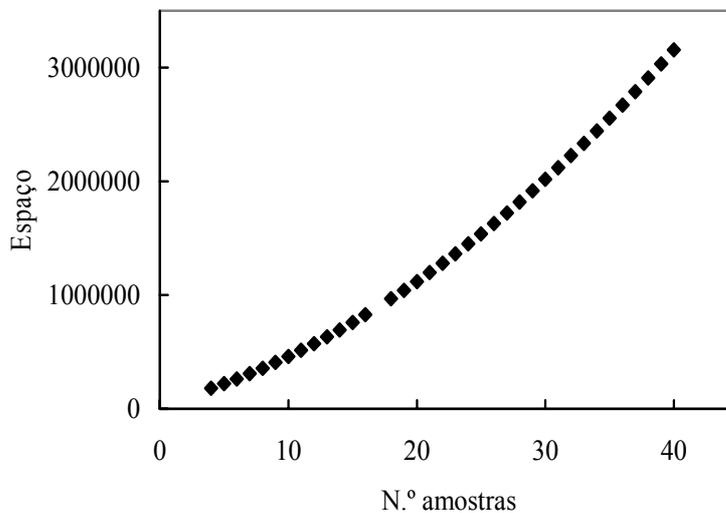
BMASSG\_SP – balanço mássico global para cada componente, ao longo de todos os períodos;

DEF\_CSI – definição do valor objectivo referente à instância aleatória em análise;

DEF\_DSVN – definição do estimador de variabilidade, desvio negativo, referente à instância em análise.

Repare-se que, no âmbito da matriz das restrições para o problema estocástico, observa-se a construção de uma estrutura angular pelas submatrizes dos cenários: o deslocamento descendente corresponde ao número de restrições de segunda fase, em conformidade com cada cenário; a translação horizontal

corresponde ao número de variáveis de recurso por amostra, cujos valores se definem na segunda fase do 2SSP.



**Figura 3.2** Espaço de memória necessário ao programa estocástico, em função do crescente número de amostras.

A resolução do problema estocástico MILP processa-se através da rotina H02BBF da biblioteca NAG, existindo uma limitação decorrente do número de amostras, de cerca de 40 a 60 amostras aleatórias, que exaurem o espaço de memória disponível. Tal número limite de amostras conduz a problemas MILP onde o número total de variáveis, discretas e contínuas, rondam cerca de 1200 a 1500, tal como o número total de restrições (de primeira e de segunda fase).

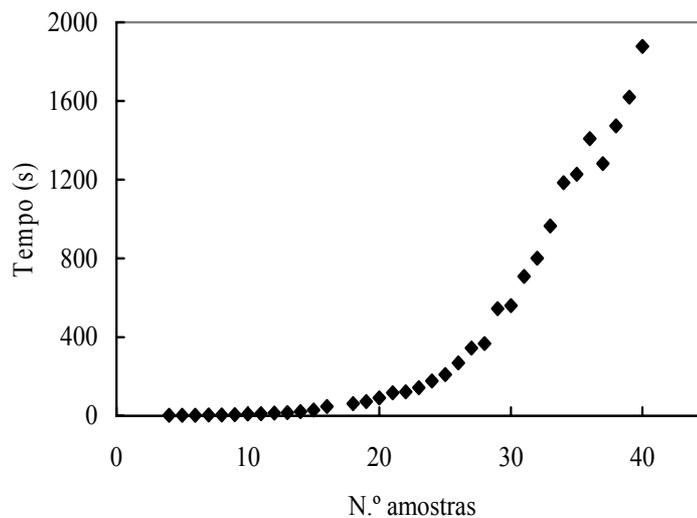
Apresenta-se na *Figura 3.2* a evolução crescente da necessidade de espaço em memória, para resolução do programa estocástico, em função do crescente número de amostras: embora o aumento de restrições e de variáveis seja directamente proporcional ao número de amostras (em que essa proporção corresponde, respectivamente, ao número de variáveis de recurso e ao número de restrições tratadas em cada instância amostra), o espaço de trabalho alocado apresenta um crescimento aproximadamente quadrático.

No entanto, o crescimento do tempo de execução do programa em função do aumento do número de amostras manifesta-se verdadeiramente exponencial, onde apenas algumas dezenas de amostras conduzem a problemas intratáveis para os meios disponíveis. Tal comportamento observa-se da *Figura 3.3*, donde se constata que um problema com cerca de 30 amostras exige aproximadamente 10 minutos de cálculo, mas problemas com 40 amostras aleatórias já necessitam de cerca de 30 minutos para completarem a sua execução.

A apresentação dos resultados é processada através da rotina RESULT\_SP, onde os valores óptimos para cada um dos tipos de variáveis (capacidades de produção, expansões de capacidades, variáveis binárias para as expansões, fluxos médios de produção, quantidades médias compradas e vendidas dos diferentes componentes, ou outras) são escritos em ficheiros dedicados, através da oportuna utilização das ferramentas para armazenagem de dados.

Para validação desta implementação do modelo estocástico, utilizam-se múltiplas amostras de um mesmo cenário ponderadas com iguais probabilidades, para

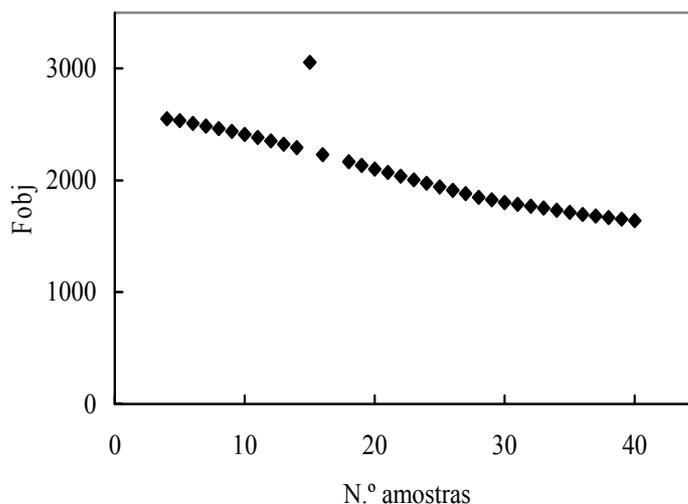
que o valor esperado óptimo dado pelo programa estocástico corresponda exactamente ao resultado obtido através da aplicação daquele cenário ao programa determinístico: assim, assegura-se que a utilização de diferentes cenários não se encontre comprometida desde o início. Dada a impossibilidade de encontrar conjuntos de dados e resultados na literatura aberta que garantissem, através da verificação de tais grupos de valores, a validade do modelo implementado, optou-se pela técnica acima descrita: para o número de amostras tratável e igualando todas as instâncias amostrais, certificou-se que o ponto e a solução óptimos do programa estocástico correspondiam aos valores óptimos do problema determinístico, otimizado perante aquela mesma amostra.



**Figura 3.3** Evolução do tempo de execução do programa estocástico com o crescimento do número de amostras.

Caso se proponha uma distribuição de probabilidades não uniforme, essa mesma distribuição deverá ser verificada em termos dos resultados das instâncias amostrais para a função objectivo estocástica, totalizando um valor esperado correspondente ao do programa determinístico: verificando-se tais valores parcelares, comprova-se que o modelo emula devidamente os efeitos decorrentes da distribuição de probabilidades aplicada.

Finalmente, apresenta-se na *Figura 3.4* a evolução do valor óptimo da função objectivo com o crescimento do número de cenários aleatórios: o valor-objectivo vai diminuindo de forma aproximadamente linear, pois embora as parcelas referentes ao valor esperado, dos fluxos materiais e dos encargos fixos derivados da solução da binária, não sofram modificações significativas, verifica-se a penalização associada à variabilidade crescente com o número de instâncias. Também, a gama do espaço probabilístico tratado revela-se bastante reduzida, pois as poucas dezenas de amostras recolhidas não permitem confirmar a robustez da solução perante o número de parâmetros incertos aqui considerado. Aliás, a pronta observação da *Figura 3.4* revela uma linha em plena fase descendente, não se vislumbrando sinais de convergência para um valor estável (assinalada por um eventual “patamar horizontal”).



**Figura 3.4** Evolução do valor da função objectivo do programa estocástico com o número de amostras.

### 3.1.3 Tratamento dos problemas para decomposição

Utilizando a implementação para o problema estocástico em duas fases e aplicando as mesmas instâncias amostrais, a aplicação de um método de decomposição que preveja a resolução de problemas MILP de menores dimensões torna-se uma alternativa vantajosa. Os benefícios das técnicas de decomposição revelam-se em termos da qualidade dos resultados, quando o número de amostras consideradas é elevado, além de que possibilitam esquemas de resolução em paralelo para os subproblemas separáveis.

A secção principal do programa estocástico por decomposição é semelhante à do programa estocástico anteriormente codificado, com a associação dos parâmetros de incerteza aos dados do problema determinístico. Para construção do espaço probabilístico possível, utilizam-se sequencialmente os procedimentos codificados nas rotinas HSS2, TRANS\_PDF e PDF: geram-se as amostras de valores normalizados, efectua-se a translação destes valores para a gama de variação respectiva e consoante a distribuição de probabilidades estipulada, bem como se atribui a probabilidade respeitante. No caso de utilização de um conjunto aleatório de cenários discretos, estes serão especificados, bem como as respectivas probabilidades.

Na presente modelação, não é necessário proceder à inicialização de vectores e matrizes ao método de resolução da biblioteca, pois tal procedimento será efectuado no âmbito de cada um dos problemas. Aí, o procedimento INIT\_MST inicializará os blocos necessários à execução do problema Mestre, em primeira fase, enquanto que se inicializam os blocos exigidos na resolução do problema de segunda fase (Sub) no procedimento INIT\_SUB.

Note-se que se devem adicionar blocos para alocação dos valores que as variáveis tomam em cada uma das iterações, exigindo espaço de memória para comportar os fluxos de informação entre os problemas Mestre e Sub: as variáveis de recurso tomam valores no âmbito do problema de segunda fase que servem como dados, valores constantes, a fornecer ao problema Mestre; neste problema de primeira fase, determinam-se os valores para as variáveis de projecto que, em reciclo, vão (re)alimentar o problema Sub.

De facto, o esquema de decomposição de Benders implementado utiliza valores calculados num problema como dados para o outro problema, compondo-se um ciclo iterativo até que a diferença entre os valores-objectivo dos problemas Mestre e Sub, respectivamente, *upper bound* e *lower bound* do programa estocástico em duas fases, seja inferior a uma tolerância dada. A estimativa inicial —para as variáveis de projecto e outros parâmetros tratados como constantes— que alimenta o problema Sub pode ser construída a partir de uma execução do programa determinístico, utilizando-se aí os valores esperados dos parâmetros incertos (*expected value problem*, EVP), ao qual se segue o procedimento *devp\_asub*; da resolução do problema Sub de LP obtêm-se os valores para as variáveis de recurso em cada uma das instâncias amostrais; actualizam-se os valores, da *lower bound* para o programa estocástico e dos parâmetros a transmitir ao problema Mestre (procedimento *dsub\_amst*); da optimização deste problema Mestre são definidos novamente os valores para as variáveis de projecto, actualiza-se a *upper bound* e os parâmetros a incorporar na realimentação de dados ao problema Sub (procedimento *dmst\_asub*). São os seguintes os procedimentos principais:

EVP – inicialização do método de decomposição, com resolução do problema EVP;

BENDERS\_SUB – resolução do problema Sub de LP, para especificação da *lower bound* do ciclo de Benders;

BENDERS\_MST – resolução do problema Mestre de MILP, para especificação da *upper bound* do ciclo de Benders;

RESULT – escrita dos resultados, em formatos e ficheiros adequados.

Para o tratamento do problema EVP, utiliza-se o modelo determinístico com os valores esperados dos parâmetros incertos, obtendo-se uma estimativa dos valores das variáveis de projecto, de primeira fase, que alimentam o problema Sub e cujo valor da função objectivo representa uma *lower bound* para o problema estocástico. Os procedimentos aí seguidos são os usuais para a definição de vectores e matriz das restrições utilizados pela biblioteca NAG:

INIT\_EVP – inicialização de vectores e matrizes do método, para o problema EVP;

FUNOBJ\_EVP – especificação da função objectivo do problema EVP;

MATRIX\_EVP – definição das restrições do problema EVP;

*Naglib* – resolução do problema EVP (MILP) através da rotina H02BFF da biblioteca NAG;

DEVP\_ASUB – preparação dos valores de RHS para o problema Sub.

O procedimento principal BENDERS\_SUB prepara a optimização do problema Sub, recorre à rotina E04MFF da biblioteca NAG para efectuar a maximização do LP, e retorna os valores óptimos das variáveis de recurso, em cada iteração e perante as instâncias amostrais consideradas.

A construção deste procedimento principal segue a metodologia aqui aplicada à codificação de modelos MILP e LP, pelo que no âmbito do problema Sub: são inicializados os blocos requeridos pela rotina E04MFF da biblioteca NAG, especifica-se o vector dos coeficientes da função objectivo, define-se a matriz dos

coeficientes das restrições, e executa-se o programa de optimização da biblioteca. Ordenadamente, os procedimentos secundários a efectuar são:

INIT\_SUB – inicialização de vectores e matrizes do método, para o problema Sub;

FUNOBJ\_SUB – especificação da função objectivo do problema Sub;

MATRIX\_SUB – definição das restrições do problema Sub;

*Naglib* – resolução do problema Sub de LP pela rotina E04MFF;

DSUB\_AMST – preparação dos valores de RHS para o problema Mestre.

Atente-se que o esquema de decomposição de Benders possibilita a selecção de diferentes conjuntos de variáveis como variáveis complicativas, pelo que é implicitamente considerado que os fluxos de dados, em recirculação entre os problemas Mestre e Sub (procedimentos *dmst\_asub* e *dsub\_amst*), podem assumir diferentes estruturas. De uma forma geral, tais dados acabam por constituir termos constantes a incorporar como RHS, em cada grupo de restrições onde sejam necessários. Esses termos constantes, parcelas ou somatórios, são identificados através do prefixo “RHS\_”, a antepor à designação do grupo de restrições a que pertencem. Como exemplo, os valores constantes e referentes às diversas parcelas que contêm somatórios nas restrições de definição do valor esperado, DEF\_CSI, são designados por RHS\_DEF\_CSI.

O procedimento principal BENDERS\_MST configura o problema Mestre, utilizando a rotina de biblioteca H02BBF para a maximização do MILP que este problema representa. Em cada iteração, obtêm-se os valores óptimos para as variáveis de projecto, a incorporar como termos constantes no problema Sub, sendo chamados os procedimentos secundários:

INIT\_MST – inicialização de vectores e matrizes do método, para o problema Mestre;

FUNOBJ\_MST – definição da função objectivo deste problema;

MATRIX\_MST – especificação das restrições deste problema;

*Naglib* – resolução do MILP;

DMST\_ASUB – preparação dos valores de RHS para o problema Sub.

O teste desta implementação decorre da verificação, para exemplos ilustrativos de reduzidas dimensões, dos valores óptimos obtidos a partir da resolução directa do modelo não decomposto, isto é, do modelo estocástico com recurso considerado anteriormente. Por conseguinte, aplicando ao método de decomposição as mesmas instâncias amostrais que foram utilizadas no problema estocástico original, o método de decomposição deverá apresentar resultados aproximados, com a tolerância desejada.

No caso de utilização de um número de cenários significativo, quer por amostragem, quer por especificação de instâncias discretas plausíveis, a matriz das restrições do problema Sub apresenta-se com um elevado número de coeficientes nulos, ou seja, torna-se uma matriz esparsa. Assim, a resolução utilizando a biblioteca matemática NAG processa-se através da rotina E04NKF, específica para matrizes esparsas, e não da rotina E04MFF, para matrizes densas.

No entanto, para correcta utilização da rotina E04NKF importa definir diversos parâmetros, entre os quais se destacam:

$nnz$  – número de coeficientes não-nulos, considerando a matriz das restrições e os coeficientes da função objectivo;

$A(nnz)$  – os referidos valores, não-nulos;

$HA(nnz)$  – índice das linhas dos coeficientes não nulos em  $A(nnz)$ ;

$KA(nsub+1)$  – índice do coeficiente em  $A(nnz)$  onde se inicializa cada coluna.

Os procedimentos inicialmente efectuados são os mesmos que no ciclo de Benders para matrizes densas (INIT\_SUB, FUNOBJ\_SUB, MATRIX\_SUB), mas têm de se alterar seguidamente, antes de chamada a rotina de optimização, com os seguintes procedimentos:

*contagem* – contagem do número de coeficientes não-nulos,  $nnz$ , considerando a matriz das restrições e os coeficientes da função objectivo; alocar espaço de memória para os vectores  $A(nnz)$  e  $HA(nnz)$ ;

*especificação* – especificação dos valores para os vectores  $A$ ,  $HA$  e  $KA$ , num duplo ciclo, por colunas e por linhas.

Antes de recorrer à optimização pela rotina E04NKF, é possível definir alguns dos parâmetros opcionais, como sejam: a imposição de valores básicos para os parâmetros (*Defaults*); o nível de informação sobre o método de resolução (*Print Level*); a unidade para mensagens, a frequência de verificação do cálculo dos resíduos (*Check Frequency*); a frequência de factorização da matriz-base (*Factorization Frequency*); a tolerância para a admissibilidade de cada restrição (*Feasibility Tolerance*); os valores de infinito (*Infinite Bound Size*, *Infinite Step Size*); o limite máximo de iterações (*Iteration Limit*), entre outros

A especificação de parâmetros opcionais para a rotina E04NKF, diferentes dos parâmetros de *Default*, pode realizar-se, alternativamente, através de duas rotinas da biblioteca:

E04NLF – utiliza um ficheiro externo, com a especificação de um parâmetro opcional em cada linha;

E04NMF – especifica directamente um parâmetro opcional em cada chamada (CALL).

A aplicação correcta dos parâmetros para a rotina E04NKF exige uma continuada experimentação, verificando-se grande dificuldade na convergência do método numérico, em face do número elevado de variáveis e restrições, com o carácter esparso associado.

### 3.2 Aplicação modular ao dimensionamento e sequenciamento de processos descontínuos (ambiente GAMS)

Nesta secção, descreve-se a metodologia utilizada na implementação dos modelos para dimensionamento e sequenciamento de processos *batch*, em ambiente de modelação GAMS. Os modelos foram descritos na Sec. 2.2, pelo que se pretende adequar o código aqui desenvolvido em ambiente GAMS, bem como se testa a implementação efectuada. Este conjunto de acções fundamenta o desenvolvimento de heurísticas associadas aos modelos apresentados, que serão certificadas através de análise probabilística.

O ambiente GAMS permite implementar modelos de Optimização (LP, NLP, MILP, SLP,...), mesmo modelos complexos e de grande dimensão, de uma forma compreensível e directa: utilizam-se declarações algébricas, numa linguagem “amigável” para o modelador e que aproxima a sua aplicação matemática e computacional. Por exemplo, a definição e a declaração de parâmetros e variáveis permite a composição implícita dos grupos das equações e restrições dos modelos, o que se traduzirá na construção automática de cada uma dessas equações e restrições por substituição de cada um dos parâmetros e variáveis. Ou seja, o ambiente GAMS constrói de uma forma automática a matriz das restrições e a função objectivo do modelo, permitindo até a sua exportação através do formato MPS (*Mathematical Programming System format*).

Os princípios gerais do ambiente GAMS assentam em aspectos de bases de dados relacionais e de MP, para satisfazer as necessidades dos modeladores: as bases de dados relacionais proporcionam a organização dos dados e o seu tratamento, como por exemplo, através da definição dinâmica de conjuntos; a MP permite a descrição do problema, ao qual se associa o método de resolução específico. A separação entre estas componentes, representação dos dados e do modelo, permite a extensão da dimensão dos problemas sem um aumento da complexidade da sua codificação. Também, a afectação dos recursos computacionais a um dado problema é realizada de forma automática, como a especificação das dimensões de vectores e matrizes, ou mesmo do espaço de trabalho.

A codificação do modelo é direccionada tanto para o utilizador como para o meio computacional, quer representando o modelo de uma forma facilmente apreensível, quer tratando os dados de uma forma algébrica, ou mesmo introduzindo texto explicativo dos diferentes componentes (conjuntos, parâmetros, variáveis, equações) dos modelos. Assim, toda a informação necessária à compreensão do modelo poder-se-á encontrar num só documento ou ficheiro.

A resolução de um problema de Optimização poderá ocorrer através da utilização de diferentes motores de cálculo (*solvers*), cada um com as suas especificidades de aplicação (LP, NLP, MILP, *etc.*), sendo possível a transportar o código de um meio computacional para outro, mesmo em sistemas operativos diferentes (WINDOWS, UNIX, ...), sem qualquer alteração.

Também, é disponibilizada uma biblioteca de modelos, onde os modelos existentes poderão servir de suporte para novos desenvolvimentos, quer em problemas de Optimização de cariz matemático, em áreas financeiras e de economia, ou mesmo no âmbito específico da Engenharia Química, onde se incluem, entre outros, modelos de: planeamento de processos; sequenciamento de processos descontínuos; síntese de redes de permutadores de calor; integração energética em

redes de colunas de destilação. De realçar que a generalidade destes modelos, quer os de Engenharia Química, quer outros, se encontra descrita na literatura aberta.

### 3.2.1 Estrutura modular da implementação

O programa principal (*main*) organiza a estrutura e aplica os módulos desenvolvidos, de maneira a implementar o modelo no ambiente GAMS. Os módulos de interesse são referenciados no programa *main* através de instruções *include*, e estão associados às principais acções a efectuar, geralmente:

PARAM – declaração e definição de índices, parâmetros e dados do problema;

MOD – declaração e definição de variáveis, equações e modelos a tratar; resolução dos modelos;

RESULT – apresentação dos resultados.

Assim, o programa principal discrimina a sequência dos procedimentos adoptados, que pressupõe: a definição inicial dos parâmetros e dados do problema; a especificação da função objectivo e restrições, constituindo o modelo; a sua resolução, através de um dos *solvers* disponíveis; a apresentação dos resultados no formato desejado. Enquanto o programa principal exhibe o sufixo (*.gms*), os módulos a incluir apresentarão o sufixo (*.inc*), pretendendo-se distribuir o desenvolvimento do código pelos diversos componentes modulares.

No módulo PARAM inicializa-se a declaração e definição dos índices (*sets*) associados aos diversos parâmetros e variáveis. Também se declaram e definem os escalares, vectores e matrizes necessários como parâmetros (*scalars, parameters*) do modelo, o que pode ocorrer de diferentes formas: por listagem, tabelas (*tables*) ou especificação directa.

Note-se que a declaração e a definição destes índices e parâmetros pode ocorrer em momentos separados, como veremos adiante na fase probabilística, onde esta informação é inicialmente declarada mas é depois definida sequencialmente em cada instância aleatória.

Na definição dos conjuntos dos índices (*sets*) pode levar-se em conta: o aspecto sequencial, por exemplo, definindo-se de uma forma implícita o conjunto de elementos inteiros desde um a cem ( $1*100$ ); o aspecto dinâmico, permitindo intersecções, complementaridades ou reuniões com diferentes conjuntos. Utilizando a instrução *ALIAS(i,j)* definem-se dois índices, *i* e *j*, idênticos.

No módulo MOD trata-se previamente da declaração e definição de variáveis (livres, positivas, negativas, binárias, inteiras), a partir das quais se processa a declaração e definição das equações a tratar, onde serão referenciadas aquelas variáveis. Usualmente, as variáveis contínuas são positivas, havendo o cuidado de especificar quais as variáveis inteiras e binárias; quanto à variável que representa o objectivo a otimizar (por exemplo, o custo total a minimizar, o lucro total a maximizar), deve ser definida como uma variável escalar e livre. A declaração das equações e a sua definição devem ocorrer em instruções separadas, assumindo uma forma algébrica similar à da formulação matemática usual, onde se utilizam símbolos intrínsecos para somas (SUM) e produtos (PRODUCT) de variáveis e parâmetros, bem como para funções trigonométricas, logarítmicas, probabilísticas, lógicas, ou outras. Proceder-se à definição dos modelos, quer assumindo todas as equações descritas (ALL), ou especificando-as individualmente; a resolução dos modelos

através da instrução SOLVE, referindo a designação do modelo em causa (TRANSPORT), a tipologia do modelo (LP), o sentido de optimização (MINIMIZING) e a variável livre da função objectivo (COST), como por exemplo, para o conhecido problema do Transporte,

“SOLVE TRANSPORT USING LP MINIMIZING COST”.

É possível a discriminação de alguns parâmetros relativos ao método de resolução, como: o número de iterações máximo (ITERLIM); o recurso-tempo utilizado (RESLIM); o desvio relativo quanto ao valor óptimo (OPTCR, parâmetro este que admite um valor entre 0 e 1); o desvio absoluto quanto ao valor óptimo (OPTCA).

No módulo RESULT prepara-se a apresentação dos resultados do cálculo de optimização, directamente através da instrução DISPLAY, sendo possível apresentar valores primais (sufixo “.l”, de *level*), valores duais (sufixo “.m”, de *marginal*), limites inferiores (sufixo “.lo”, de *lower bound*), limites superiores (sufixo “.up”, de *upper bound*).

Os formatos aplicados podem ser adequados às finalidades visadas, como para elaboração de figuras, para tratamento estatístico, para análise comparativa de resultados, sendo os valores inscritos em ficheiros externos através da instrução PUT. Exemplificando:

- “PUT X:5:0;” escreve o valor de X (inteiro) com 5 dígitos, considerando 0 dígitos decimais;
- “PUT X:12:4;” escreve o valor de X (fracção) formatado com 12 dígitos, dos quais 4 dígitos são decimais;
- “LOOP (i, PUT X(i):5:0);” escreve os valores do vector de inteiros X(i).

Esta implementação dos modelos de interesse é validada por comparação directa dos resultados aqui obtidos com os valores disponíveis na literatura, quando sejam disponibilizados os respectivos conjuntos de dados e resultados.

Ora, o modelo *batch\_milp\_fssz* aqui implementado em ambiente GAMS, não verificou os resultados disponibilizados na literatura: de facto, na apresentação original do modelo [Voudouris e Grossmann, 1992] não constava o valor do parâmetro H, referente ao horizonte temporal; esse mesmo exemplo numérico, contendo dados e resultados, constava em Biegler *et al.* [1997], páginas 725 a 728, reportando um valor de H = 6200 (horas); com este valor, os resultados obtidos por resolução através de diferentes *solvers* eram coincidentes numa solução melhor do que a descrita naquelas referências. Tentando contactar-se os primeiros autores, não se conseguiu resposta.

Assim, foi necessário confirmar a implementação efectuada através da comparação com instâncias específicas de outros modelos. Ou seja, no sentido de garantir a coerência entre os diversos modelos analisados, procedeu-se à seguinte metodologia comparativa:

- entre os modelos *batch\_milp\_fssz* (*Flowshop/Single machine/SPC/ZW*) e *batch\_milp\_fmsz* (*Flowshop/Multiple machine/SPC/ZW*), restringindo-se neste último o número de processos em cada estágio a  $NP(j) = 1$ ; efectuando-se duas séries de 100 amostras aleatórias, os resultados mostraram-se iguais aos do primeiro modelo, formulado considerando processo único em cada estágio;

- entre os modelos *batch\_milp\_fsmz* (*Flowshop/Single machine/MPC/ZW*) e *batch\_milp\_fmmz* (*Flowshop/Multiple machine/MPC/ZW*), também se restringindo aqui o número de processos em cada estágio a  $NP(j) = 1$  e considerando folgas nulas,  $SLK(i,k,j) = 0$ , para estabelecer concordância com as hipóteses subjacentes entre estes dois modelos; realizando-se duas séries de 30 amostras aleatórias, os resultados do segundo modelo revelaram-se iguais aos do primeiro modelo, que foi formulado como de processo único em cada estágio.

Deste modo foi possível estabelecer a coerência relativa entre os modelos em análise. Uma mesma metodologia comparativa, mas dirigida ao aspecto da política de campanha utilizada, produto único (SPC) *versus* produtos múltiplos (MPC), permitiu esclarecer algumas imprecisões ou indefinições subjacentes a estes modelos.

### 3.2.2 Análise probabilística de heurísticas

No âmbito do desenvolvimento de heurísticas, propõe-se a realização de um estudo de análise probabilística dos seus resultados, nomeadamente quanto ao desvio do valor da função objectivo, ao desvio da solução, ou quanto aos tempos de execução, pelo que haverá que adequar a estrutura de implementação em ambiente GAMS anteriormente proposta. Então, no ficheiro principal (MAIN) que organiza o processo sequencial de cálculo, será incluído um ficheiro que descreve o modelo (MOD), ao qual se segue um ciclo a repetir segundo o número de instâncias amostrais consideradas: definição da instância probabilística dos dados, seguindo-se a respeitante resolução do problema, quer pela via do *solver* disponível, quer através da aproximação heurística, finalizando-se com a apresentação dos resultados e respectivos desvios. Então, a estrutura geral contempla os seguintes módulos:

MOD – declaração de índices e parâmetros; declaração e definição de variáveis, equações e modelos a tratar;

(em ciclo) – DATA – definição de índices, parâmetros e dados (aleatórios) do problema;

MIP – resolução da instância através de *solver*;

HEUR – resolução por aproximação heurística;

RESULT – apresentação dos resultados e desvios.

O módulo MOD, a incluir inicialmente, pretende a usual definição e declaração de variáveis, equações e modelo, de forma a ser utilizado de forma invariável perante as múltiplas instâncias amostrais do problema. No entanto, há que considerar neste módulo a declaração prévia, mas não a definição, dos parâmetros do problema, para a correcta definição das equações e modelo.

No módulo DATA, que inicia cada ciclo, procede-se à definição dos parâmetros do modelo, recorrendo a funções intrínsecas de natureza probabilística: a função  $UNIFORM(arg1, arg2)$  permite definir os valores desejados com uma distribuição uniforme entre os valores de  $arg1$  e  $arg2$ ; a função  $NORMAL(arg1, arg2)$  define os valores aleatórios com uma distribuição normal de média  $arg1$  e desvio padrão  $arg2$ . Para garantir a o carácter repetitivo das experiências, especificou-se a “semente” (SEED) da qual se deriva a geração de valores aleatórios. A especificação da “semente” deve ocorrer através da explicitação de um número inteiro, não sendo possível a sua definição implícita. No intuito da

utilização dos dados aleatórios por um procedimento heurístico, que se for algo complexo não será possível exprimi-lo com as restritas regras do ambiente GAMS, pretende-se a escrita dos dados para um ficheiro externo. Deste modo, será possível a implementação das heurísticas noutra linguagem, como o Fortran.

Neste ponto há que realçar a possibilidade de existência de desvios numéricos entre o tratamento, em GAMS e em Fortran, da função potência (utilizada para o cálculo dos custos de equipamento), devendo atender-se à sensibilidade numérica desta função, se for utilizado diferente número de dígitos significativos naqueles ambientes. A via usada foi a de se calcularem os parâmetros dos custos de equipamento a partir dos factores elementares (coeficiente e expoente) apenas no ambiente GAMS, sendo esses parâmetros dos custos escritos para ficheiro externo e utilizados igualmente na optimização MIP e na aproximação HEUR, em vez da situação inicial de escrita externa dos factores elementares e cálculo paralelo dos custos de equipamento.

Quanto à geração aleatória dos dados amostrais, é de referir que tal aspecto terá de ser inicializado através da “semente” em cada uma das amostras consideradas na análise probabilística: o tratamento sequencial dessas amostras não se mostrou passível de aplicação aos diferentes modelos analisados, pois estes supunham dimensões diferentes para o conjunto de dados, embora houvesse coincidência destes dados na quase totalidade dos casos. Exemplificando, embora todos os casos em estudo utilizassem as mesmas procuras, tempos de processamento em cada estágio, parâmetros dos equipamentos (custos e dimensões discretas), factores de dimensão, certos parâmetros como os tempos de limpeza, número de processos em cada estágio, ou as folgas nos estágios apenas são consideradas, adicionalmente, em certos casos.

No módulo MIP trata-se da resolução óptima do problema matemático, através da instrução *solve*, antecipadamente à qual se definem os parâmetros de optimalidade. De facto, é de interesse garantir que a solução obtida corresponde ao óptimo do problema, pelo que se especifica:

- “OPTCR = 0.0;” indicando que o desvio relativo entre a solução actual e a solução óptima é nulo;
- “OPTCA = 0.0;” significando que o desvio absoluto entre a solução actual e a solução óptima também é nulo.

Os resultados obtidos para a solução óptima são escritos para um ficheiro externo de resultados, para posterior análise dos desvios da heurística. São também de interesse os parâmetros da resolução efectuada:

- “RESLIM = 3600;” estabelece o limite temporal de uma hora de cálculo para finalizar a optimização de cada instância aleatória;
- “ITERLIM = 1000000000;” define o número máximo de iterações para optimização de cada amostra;
- “nome\_do\_modelo.resusd” permite especificar o tempo de cálculo utilizado em cada amostra;
- “nome\_do\_modelo.modelstat” providencia o estado da solução final apresentada para o modelo (valor 1 para solução óptima);

- “nome\_do\_modelo.solverstat” informa do estado final do solver (valor 1 para solução óptima);

Quanto ao módulo HEUR, que trata da resolução via heurística, pode ser implementado em GAMS caso seja um algoritmo simples, sendo automática a execução deste módulo. Perante uma situação mais complexa, o algoritmo é implementado em Fortran, pelo que haverá lugar à seguinte sequência: leitura dos dados aleatórios em ficheiro externo; resolução heurística; escrita de resultados. Nesta situação, o que ocorria era a execução dos módulos DATA e MIP para a totalidade da amostra em análise em ambiente GAMS, ao qual se seguia a execução dos procedimentos HEUR e RESULT, também para toda a amostra.

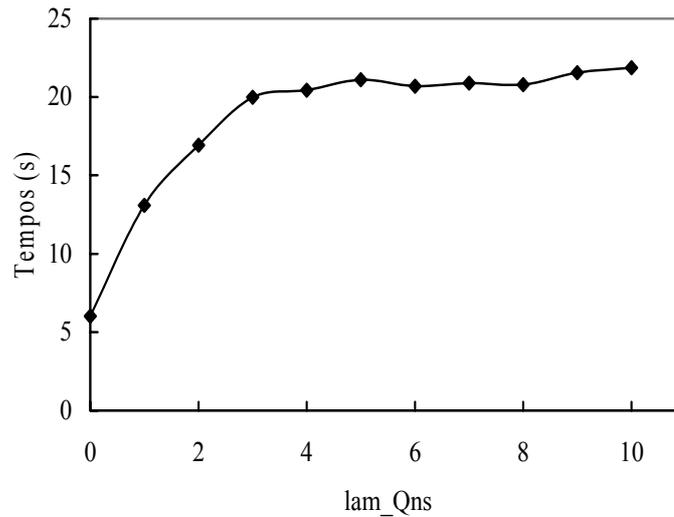
No módulo RESULT, através da leitura dos resultados das resoluções exacta e aproximada, procede-se à determinação dos desvios e tempos associados, escrevendo-se os valores em formato adequado à análise de resultados, nomeadamente, quanto à construção de gráficos e tabelas. Neste mesmo módulo, são quantificados os desvios respeitantes aos valores aproximativos da função objectivo e da solução binária associada, determinando-se os estimadores desejados: percentagens de instâncias inadmissíveis; percentagens de instâncias com valor heurístico exacto; valores esperados dos desvios percentuais na função objectivo, para as instâncias admissíveis; desvio padrão para esses desvios percentuais; percentagens de instâncias com solução binária heurística exacta; valores esperados para os desvios na solução binária.

### 3.2.3 Generalização estocástica

No âmbito da generalização dos modelos de dimensionamento e sequenciamento de processos descontínuos, implementaram-se versões estocásticas do modelo com processos múltiplos em cada estágio e com campanhas de produto único (SPC), na forma do modelo *spbatch\_milp\_fmsz* (Sec. 5.2), versando dois casos de estudo em ambiente de procura incerta e sob a forma de cenários discretos: a minimização dos custos de investimento, satisfazendo parcialmente a procura incerta, mas penalizando-se a procura não-satisfeita; a maximização do VLA robusto, considerando o valor esperado do retorno pelos produtos vendidos, os custos de investimento, penalizando o excesso (folga) das capacidades instaladas e a variabilidade associada à incerteza nos parâmetros.

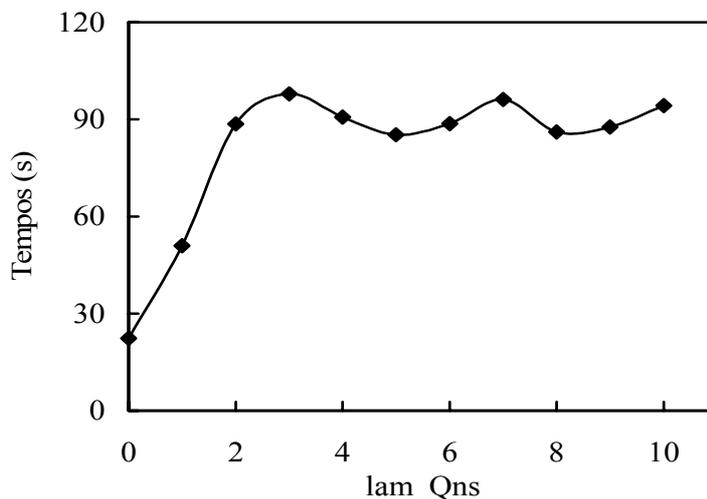
Dada a dimensão que tais problemas estocásticos, com robustez e multiperíodo, podem atingir —pois apresentam um número de variáveis e restrições dependentes de seis índices diferentes (números de estágios, de produtos, de dimensões discretas em cada estágio, de processos por estágio, de períodos de tempo, de cenários discretos)—, um factor relevante é o tempo necessário para a optimização de cada instância. Atendendo a que se procede ao estudo de diferentes configurações para os sistemas de processos descontínuos, bem como à parametrização das penalidades associadas, o tempo de resolução de cada instância apresenta um efeito propagador sobre o tempo de resolução global. Assim, o factor tempo de resolução é tratado de forma a conduzir à sua diminuição, descrevendo-se a metodologia utilizada.

Atente-se na evolução dos tempos de resolução para o problema estocástico em *spbatch\_ex1*, nomeadamente, quanto à sua variação com o número de cenários,  $NR$ , ou até com o parâmetro de penalização da procura não-satisfeita,  $\lambda Qns$  (ou  $lam\_Qns$ ).



**Figura 3.5** Variação dos tempos de execução (para  $NR = 15$ ) na gama de penalização considerada.

Da *Figura 3.5*, observa-se que o tempo de execução aumenta significativamente no início da penalização da procura não-satisfeita, o parâmetro  $\lambda Qns$  (ou  $lam\_Qns$ ), atingindo depois um valor de referência, à volta do qual se mantém o tempo de execução, mesmo para valores de penalização mais elevados do que os representados. Também, a mesma tipologia de evolução ocorre para outras instâncias com diferentes números de cenários, como se pode observar da evolução da linha dos tempos na *Figura 3.6*, mas para 30 cenários discretos.



**Figura 3.6** Variação dos tempos de execução (para  $NR = 30$ ) na gama de penalização considerada.

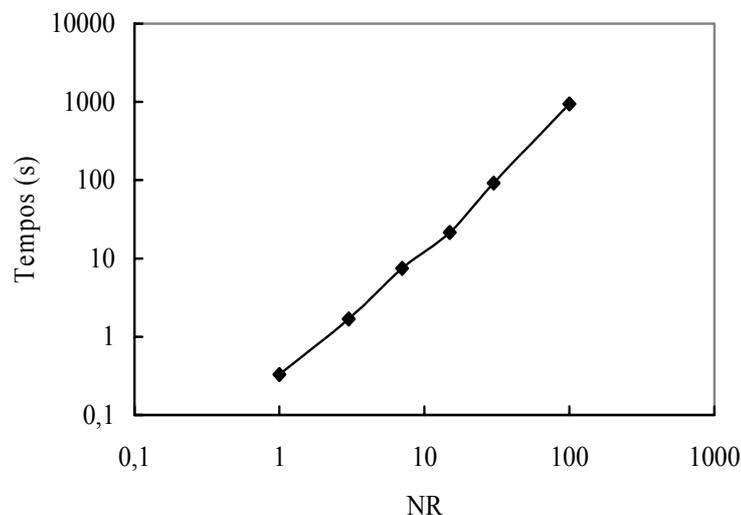
Também nesta situação, para  $NR = 30$ , pode-se confirmar uma forte variação dos tempos de execução na zona inicial do parâmetro de penalização,  $\lambda Qns$ , ao qual se segue uma zona de valores para os quais o tempo de execução varia em torno de um valor relativamente estável. No entanto, embora a evolução dos tempos nestas

duas instâncias,  $NR = 15$  e  $NR = 30$ , sejam semelhantes, deve afirmar-se a diferença de ordem dos valores representados: enquanto o número de cenários aumentou para o dobro, de 15 para 30, os tempos de execução aumentaram mais de quatro vezes (considerando os valores de estabilidade, de cerca de 20 segundos passou para cerca de 90 segundos).

Esta variação sugere um tratamento exponencial (ou logarítmico) para a variação dos parâmetros tratados, número de cenários  $NR$  e tempos de execução, tendo-se obtido a representação em escala logarítmica (decimal) que se apresenta na *Figura 3.7*. Aí representam-se os valores médios dos tempos de execução, sendo cada ponto obtido da média aritmética dos tempos de 40 instâncias com o mesmo número de cenários  $NR$ .

Por curiosidade, refira-se que para  $NR = 100$  se obteve um valor médio de cerca de 900 segundos, ocorrendo aí dois “patamares” de estabilidade: um primeiro, logo depois dos valores iniciais de  $\lambda Qns$  (superiores a 3), é ligeiramente superior àquele valor médio, representando cerca de 1000 segundos; quando  $\lambda Qns$  aumenta para valores além de 30, ocorre um degrau negativo nos tempos médios para valores de cerca de 800 segundos.

Uma simples regressão linear logarítmica entre os valores dos tempos médios e dos cenários  $NR$ , considerando-se a base decimal (10) —mas válida noutras bases (por exemplo, na base  $e$ )—, conduz a um mesmo expoente de 1.722, mas com o valor da intercepção linear dependente da base em que se realiza esta regressão.



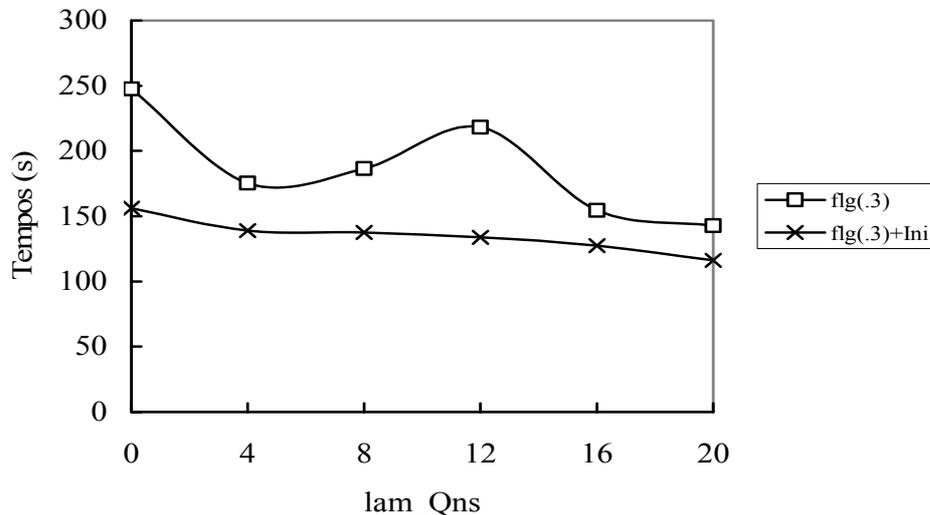
**Figura 3.7** Variação dos tempos médios de execução, em termos do número de cenários considerados, em escala logarítmica.

Para um eficaz tratamento computacional das sucessivas instâncias, analisou-se o efeito da inicialização de variáveis, nomeadamente, a fixação de valores de referência, ou a especificação de gamas de variação (*upper* e *lower bounds*). Esta sucinta análise foi realizada perante uma instância ( $\lambda dsv = 1$ ;  $\lambda flg = 0.2$ ;  $\lambda Qns = 4$ ) do exemplo numérico *spbatch\_ex2* em EX2B ( $N = 4$ ;  $M = 3$ ;  $NS = 5$ ;  $NP = 3$ ;  $NT = 10$ ;  $NR = 5$ ), estipulando-se as seguintes situações de inicialização, bem como se descrevem os resultados obtidos:

- 
- A) produzir resolução de referência não inicializada, ou seja, considerando-se todas as variáveis com valor inicialmente nulo; obteve-se um tempo de execução de 146 segundos;
  - B) fixar valor da função objectivo, atribuindo-lhe o valor óptimo obtido na resolução de referência em A); a solução óptima não foi alcançada após 329 segundos (com um tempo-limite para execução do *solver* OSL de 300 segundos), sendo apenas reportada uma solução admissível fraccionária;
  - C) colocar uma *lower bound* “dura” na função objectivo a maximizar, atribuindo-lhe o valor óptimo obtido na resolução de referência em A); foi obtida a solução óptima ao fim de 126 segundos;
  - D) colocar uma *lower bound* “macia” na função objectivo a maximizar, atribuindo-lhe um possível desvio negativo de 3% do valor óptimo de referência; a solução óptima obteve-se passados 132 segundos;
  - E) colocar uma *upper bound* “dura” na função objectivo a maximizar, atribuindo-lhe o valor óptimo obtido em A); a resolução cessou ao fim de 320 segundos, devido ao limite aos tempos de execução, sendo reportada uma solução inteira subóptima;
  - F) colocar uma *upper bound* “macia” na função objectivo a maximizar, atribuindo-lhe um possível desvio positivo de 3% do valor óptimo em A); a execução do optimizador terminou passados 300 segundos, da qual resultou uma solução fraccionária subóptima;
  - G) fixar os valores das variáveis binárias,  $y(j,s,p)$ , aos valores óptimos obtidos na resolução de referência em A); a solução óptima foi confirmada no tempo de 7 segundos;
  - H) colocar *lower bounds* e *upper bounds* para as variáveis binárias,  $y(j,s,p)$ , em conformidade com os valores óptimos de A); se o valor óptimo era nulo, especificar *upper bound* nula; perante valor óptimo unitário, especificar *lower bound* unitária; com estas restrições à variação das variáveis binárias, a solução óptima foi confirmada em 0.5 segundos;
  - I) seleccionar uma vizinhança, a partir da solução binária óptima em A), para tornar mais eficiente a pesquisa; foi atingida a solução óptima, sendo necessários 125 segundos;
  - J) similar ao ponto anterior, em I), mas definir uma vizinhança mais alargada; a solução óptima foi determinada em 104 segundos;
  - K) inicializar todas as variáveis com os respectivos valores óptimos, anteriormente obtidos na resolução de referência em A); apenas depois de 155 segundos é que a solução óptima foi atingida.

Através da qualidade das soluções obtidas, bem como comparando os tempos de cálculo necessários, verifica-se desta experiência numérica que: é de interesse especificar as variáveis binárias, preferivelmente através de *bounds* à gama de variação; a fixação e a colocação de *upper bounds* sobre o valor da função objectivo revelaram-se genericamente contraproducentes, sem se atingir a solução óptima no tempo de cálculo disponível, apenas tendo relativo sucesso a imposição de *lower bounds* ao valor-objectivo (a maximizar); a inicialização das variáveis com os

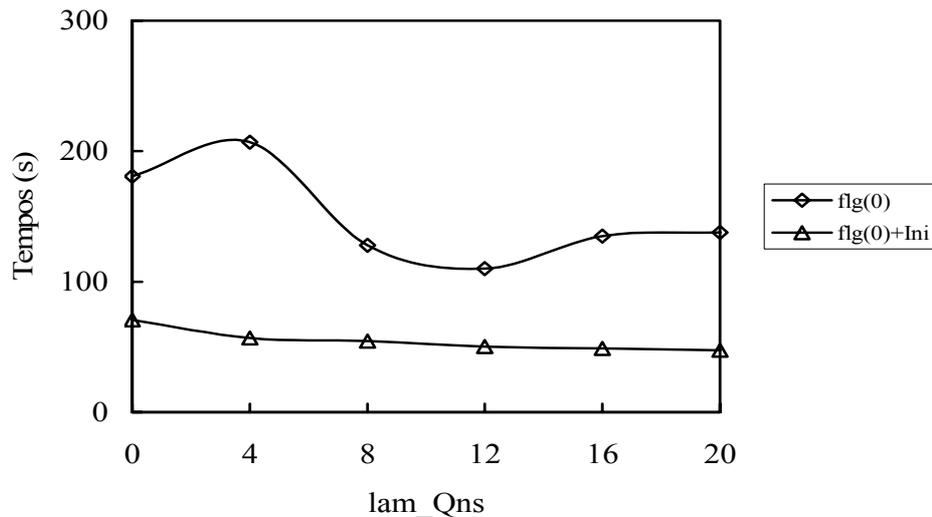
respectivos valores óptimos não revelou quaisquer efeitos em termos de diminuição do tempo de resolução; a definição de uma vizinhança da solução óptima para aí se efectuar a optimização numérica conduziu a uma diminuição dos tempos de resolução, embora esta diminuição não se apresente directamente relacionada com a diminuição da vizinhança de procura.



**Figura 3.8** Tempos de execução, comparando instâncias com valor não-nulo para  $\lambda flg$ , especificando-se (ou não) vizinhança para a pesquisa da solução binária.

Esta última técnica de inicialização descrita, estipulando uma vizinhança para procura da solução óptima, foi aplicada sistematicamente na optimização de instâncias (sufixo +Ini) do referido exemplo EX2B, tendo-se obtido reduções significativas nos tempos de resolução: cerca de 35% de redução em termos globais, considerando os diferentes valores dos parâmetros de penalização; tomando apenas as instâncias onde a penalidade por folgas é nula,  $\lambda flg = 0.$ , a redução de tempos excedeu os 50%. Alguns valores comparativos são apresentados na *Figura 3.8*, sendo tal relação de tempos representativa para as diferentes instâncias de parâmetros avaliadas (isto é, comparando o efeito da inicialização para  $\lambda flg = 0.3$ , fixando  $\lambda dsv$  e  $\lambda Qns$ ).

De forma semelhante, observa-se o efeito da inicialização (sufixo +Ini) por definição duma vizinhança para a procura da solução binária, que é deveras mais vincado: a redução nos tempos de execução excede, em regra, os 50%, como se revela comparando as linhas representadas na *Figura 3.9*, respeitando a um valor nulo para o parâmetro da penalização da folga. Repare-se também, já que estas referidas figuras apresentam a mesma escala para os tempos (de 0 a 300 segundos), que as instâncias com parâmetro nulo para a penalização das folgas de capacidade,  $\lambda flg = 0$ , são optimizadas em tempos substancialmente menores. Tal parece indicar que, considerando as folgas de capacidade no problema através da sua penalização na função objectivo robusta, impõe um critério cuja ponderação necessita de maiores tempos de execução.



**Figura 3.9** Tempos de execução, comparando instâncias com valores nulo para  $\lambda flg$ , especificando-se (ou não) vizinhança para a pesquisa da solução binária.

## Sinopse

Na Sec.3.1, seguiu-se uma abordagem evolutiva em termos da implementação em Fortran (utilizando a máquina Alfa, descrita em Apêndice 3) de diversos modelos de planeamento de processos:

- iniciou-se com um modelo determinístico, tratando modularmente de inicializar parâmetros e variáveis, definir os coeficientes da função objectivo, especificar os elementos da matriz das restrições e os termos independentes, obtendo-se os resultados na formatação desejada;
- seguiu-se uma formulação estocástica, com recurso (2SSP), incorporando-se a amostragem estratificada, a definição do espaço de procura das variáveis incertas e a das probabilidades associadas;
- pretendendo resolver problemas de grande dimensão, implementou-se a decomposição de Benders para um modelo robusto, revelando-se extremas dificuldades no tratamento do problema esparsos.

As implementações foram testadas, por verificação de valores da literatura ou por adequação dos resultados a instâncias específicas dos dados. Assim, enquanto que a implementação do modelo determinístico corresponde à de Sahinidis *et al.* [1989], descrito no Capítulo 2, utilizando os seus dados e resultados como objecto de certificação, toda a restante implementação efectuada é autónoma. Dado que a implementação é desenvolvida de uma forma modular, torna-se geral a sua aplicação aos modelos estocásticos, como o modelo robusto desenvolvido no Capítulo 5.

A dificuldade da implementação em Fortran dos diferentes modelos de MILP, tornou necessário uma metodologia para definição dos coeficientes, quer da função objectivo, quer da matriz das restrições e termos independentes, o que permitiu a utilização dos métodos da biblioteca NAG. No entanto, afirma-se a necessidade de

meios mais eficientes para a implementação de modelos matemáticos, como se verifica através da aplicação do ambiente GAMS.

Na Sec. 3.2, descreveu-se a metodologia de implementação modular em ambiente GAMS de diferentes modelos para dimensionamento e sequenciamento de processos descontínuos, nos seguintes passos sucessivos:

- implementaram-se modelos determinísticos da literatura, especificando modularmente parâmetros e variáveis, definindo a função objectivo e as restrições que constituem cada modelo e formatando os resultados; utilizou-se a máquina Beta (características descritas em Apêndice 3);
- realizou-se análise probabilística, conjugando a resolução de instâncias múltiplas para os diferentes exemplos numéricos relativos a cada modelo, por utilização do ambiente GAMS, com a implementação em Fortran de diversos procedimentos heurísticos, e procedendo à respectiva quantificação de desvios; tal como no passo anterior, utilizou-se a máquina Beta, para os devidos efeitos comparativos;
- generalizou-se a enquadramento estocástico a formulação de interesse (entre as quatro estudadas), com promoção da robustez, em face da incerteza associada às procuras e retornos dos produtos; a qualidade da formulação generalizada foi avaliada em termos de parâmetros técnicos e económicos de relevo; neste passo, utilizou-se a máquina Gama (descrita em Apêndice 3).

As diferentes implementações dos modelos foram verificadas, quer através de exemplos da literatura, quer por instâncias comparáveis de diferentes modelos. Tal como no trabalho numérico efectuado na Sec. 3.1, também aqui se iniciaram os trabalhos de implementação com a certificação de modelos determinísticos descritos no Capítulo 2, conforme Voudouris e Grossmann [1992]; a análise probabilística e a generalização estocástica correspondem aos trabalhos aqui desenvolvidos, respectivamente, nos Capítulos 4 e 5.

Em termos comparativos, revelou-se mais eficiente a implementação de modelos MILP através do ambiente GAMS, relativamente à conduzida em Fortran: *i)* quer em termos da codificação, pois o GAMS apresenta uma linguagem direccionada para tal; *ii)* quer em termos de correcção ou ajuste da formulação. Para os *solvers* utilizados, principalmente o OSL, foram buriladas técnicas de inicialização e de delimitação de pesquisa que os tornam mais rápidos, embora não fosse realizada uma análise comparativa entre os diferentes *solvers*. No entanto, a programação de ciclos de repetição e condicionais em ambiente GAMS, bem como o desenvolvimento de procedimentos de atribuição, torna-se uma tarefa morosa e difícil, quando comparado com o Fortran. Tal aspecto manifestou-se de forma evidente no desenvolvimento de heurísticas (Capítulo 4), quando era pretendida a automatização da resolução heurística juntamente com a resolução via *solver*. Um exemplo agudo destas dificuldades encontra-se em Apêndice 3, dirigindo-se à formação de campanhas de produtos múltiplos a partir dos valores óptimos de transições de produção, formação que em Fortran se apresenta mais compacta e compreensível.





### Análise dos modelos e heurísticas

As técnicas de otimização inteira e mista são frequentemente utilizadas em Engenharia Química, permitindo tratar uma grande variedade de problemas dada a flexibilidade e rigor das formulações. No entanto, quando a dimensão do problema aumenta, verifica-se geralmente um crescimento exponencial do esforço computacional, o que se traduz na elevada dificuldade ou mesmo impossibilidade em desenvolver algoritmos exactos e eficientes. A eficiência computacional dos problemas de otimização combinatoria que se nos deparam constitui um desafio, especialmente se a necessidade de obtenção de resultados decorre em tempo restrito, ou mesmo em tempo real.

Relativamente aos problemas MILP de interesse propõe-se o desenvolvimento de investigação analítica e da complexidade computacional (sendo os fundamentos desta descritos no Anexo 4-A). A investigação analítica é utilizada como instrumento teórico para caracterizar tanto o comportamento de heurísticas como de algoritmos exactos [Liu e Sahinidis, 1997], de forma a conjugar as vantagens destes métodos de resolução. Perante a dificuldade dum problema combinatorio de grande dimensão, interessa promover a rapidez do cálculo aplicando determinada regra heurística, mas também haverá que garantir a optimalidade da solução, o que será proporcionável pelo método exacto. Donde, torna-se curial proceder a um estudo teórico das heurísticas dirigidas aos problemas de otimização que patenteiam elevado cariz combinatorio. A abordagem teórica envolve a classificação da complexidade, nomeadamente, identificando: problemas de IP que se mostrem como casos especiais daqueles; problemas polinomialmente redutíveis a eles; e transformações lineares que permitam uma relação de equivalência (do hiperespaço de soluções admissíveis e da solução óptima) entre os problemas MILP e IP em estudo.

Desenvolvem-se regras heurísticas que visam providenciar soluções de qualidade utilizando um tempo de cálculo razoável. Designadamente, para cada problema de interesse, trata-se de estimar a solução óptima, o valor óptimo da função objectivo, quer estabelecendo correlações entre os parâmetros mais críticos do problema, quer visando a construção de regras heurísticas. Estas regras permitirão a

abordagem de problemas difíceis (por ex., problemas NP-hard) de grandes dimensões e estarão suportadas numa formulação adaptável, bem como em conceitos simples e intuitivos que permitirão explorar o conhecimento sobre o problema, mas que não darão a garantia da solução óptima. Direccionadas para o desenvolvimento de heurísticas, apresentam-se descrições simplificadas destas regras em Anexo 4-B. Também, os relatos das experiências realizadas por outros constituem um suporte que convém não negligenciar. De forma sintética, revêem-se algumas técnicas empíricas:

- desenvolvimento do procedimento heurístico com base em LP —recorrendo ao programa linear relaxado, LP-rel e usando propriedades da dualidade e da complementaridade, investigam-se as propriedades analíticas e os desempenhos; desta forma, aproveita-se a complexidade típica da LP, reconhecida como polinomial;
- análise de complexidade — visando simplificações a subproblemas marginais ou a casos especiais, ou buscando transformações em tempo polinomial a problemas conhecidos, é possível a explanação matemática das propriedades; habitualmente, analisam-se tanto aproximações estáticas (especificidades de um único período temporal), como aproximações dinâmicas (abordagem multiperíodo, estudo do problema simplificado ao longo do horizonte temporal);
- utilização simultânea (hibridação) de regras heurísticas — aplicando heurísticas diferentes, baseadas em pressupostos diversos ou mesmo antagónicos, pretende aproximar-se mais fielmente a combinatoriedade da solução (exemplificando: no problema de planeamento de processos, a regra heurística de fixação das variáveis binárias no período inicial origina desvios na solução para horizontes longos; esta heurística pode ser corrigida ou melhorada, eventualmente, introduzindo uma outra regra que estipula uma agregação dos períodos temporais);
- reconhecimento de relações de causa e efeito — identificando o nexo de causalidade (por exemplo, a variabilidade dos parâmetros afectará a variabilidade do ponto óptimo), permite favorecer a ocorrência se tal for benéfico, ou procurar a inversão dos efeitos em caso contrário;
- aplicação de procedimentos conhecidos — nomeadamente, para MILP, existem procedimentos cujos benefícios são consagrados (preprocessamento, SOS, redução de coeficientes, fixação de variáveis, planos de corte).

Realiza-se uma análise probabilística às técnicas heurísticas propostas, pelo que será possível caracterizar quantitativamente o seu desempenho, o qual é obténivel perante situações típicas do problema de grande dimensão. Usualmente, em problemas de cariz combinatório (*knapsack*, TSP), processam-se os seguintes tipos de análise probabilística: dos tempos de resolução e utilizado pelo algoritmo; do erro ou desvio entre a estimativa obtida pela heurística e a solução óptima; do correspondente desvio entre os valores-objectivo da estimativa e da solução óptima.

As principais desvantagens da análise probabilística decorrem da hipótese assumida para a distribuição de probabilidade dos dados e parâmetros do problema, bem como de os seus resultados só serem válidos de forma assintótica, isto é, para problemas de grande dimensão. Neste último caso, conviria recorrer a uma análise

empírica para verificação dos resultados probabilísticos em problemas de menor dimensão.

Pretendem recolher-se ensinamentos quanto à construção da estimativa para a solução óptima de cada modelo, a partir da qual decorre a determinação do valor-objectivo óptimo. Importa aqui distinguir:

- determinação dos valores óptimos para as variáveis binárias — caso se dispusesse dos valores binários complicativos, a optimização subsequente processar-se-ia através de LP; deste modo, é usual tratar separadamente estas variáveis por métodos de relaxação e de decomposição, até agrupando-as nalgum submodelo que possa ser eficientemente processado;
- determinação dos valores óptimos para as variáveis contínuas — como descrito acima, é proveitoso otimizar as variáveis contínuas conhecendo a configuração da solução binária; no entanto, tal não invalida que se desenvolva algum esquema de cálculo onde a definição dos valores binários decorra, indirectamente, da determinação das variáveis contínuas.

Assim, no âmbito dos problemas equacionados, englobando quer o planeamento de processos flexíveis (prefixo *plan*), quer o dimensionamento e sequenciamento de sistemas descontínuos (prefixo *batch*), procede-se a:

- enquadramento da problemática, realçando os aspectos mais significativos do valor-objectivo e do espaço de procura das soluções admissíveis definido pelas restrições, reavaliando as principais limitações;
- abordagem à complexidade computacional, por análise de versões específicas do problema (versão estática, versão dinâmica, casos especiais);
- desenvolvimento de heurísticas que, como usualmente, pretendem encontrar soluções admissíveis de boa qualidade, estimando o ponto óptimo ou a solução óptima, de forma a diminuir o esforço de cálculo dos problemas de maior dimensão;
- explanação matemática das propriedades das heurísticas desenvolvidas, quer optando pela sua fundamentação, quer pela abordagem probabilística onde se caracterizam estatisticamente diversas métricas (desvios da solução, valor óptimo, tempo de cálculo);
- apresentação de resultados numéricos comprovativos do desempenho dos procedimentos, quantificando o esforço computacional (usualmente, tempo de CPU para cálculo), através de tabelas descrevendo a evolução dos parâmetros de interesse; assim, comprova-se a valia das heurísticas, através do relato detalhado da avaliação experimental.

Aplicando os conhecimentos de índole teórica e probabilística assim obtidos, possibilita-se uma adequação a problemas de optimização com grandes dificuldades de resolução, como quando se pretende melhorar a robustez da solução óptima ou a flexibilidade do sistema modelado, levando até em conta as incertezas na definição dos parâmetros do problema.

## 4.1 Abordagem teórica ao planeamento de processos

Procede-se a uma abordagem da complexidade computacional dos problemas descritos pelos modelos de planeamento de processos (em Sec. 2.1), onde se classificam esses problemas segundo o pior caso possível de desempenho do respeitante algoritmo mais eficiente.

Refira-se que um problema é NP-hard (ver Anexo 4.A) caso exista um problema NP-completo que possa ser polinomialmente reduzido naquele, sendo essa redução em tempo polinomial. Donde, um problema NP-hard é pelo menos tão difícil como um problema NP-completo, e um algoritmo polinomial para um problema NP-hard implica que  $P = NP$ .

A abordagem genérica, para comprovar que um problema particular de optimização  $Q$  é NP-hard, consiste em identificar problemas reconhecidamente NP-hard como sendo casos especiais do problema  $Q$ . E também: identificar um problema reconhecido como NP-completo que seja passível de redução polinomial para o problema de admissibilidade de minorante (*Lower Bound Feasibility*), o qual consiste num caso especial do problema  $Q$ .

### 4.1.1 O problema com processos dedicados

Todos os modelos MILP de planeamento de processos descritos na Sec. 2.1, apresentam ( $NP.NT$ ) variáveis discretas correspondentes às decisões binárias (0/1) sobre expansão de capacidades de cada processo, em todos os períodos de tempo. Neste aspecto, ocorre uma situação de equivalência entre todos os casos estudados, sejam determinísticos (*plan\_milp\_lrp*, *plan\_milp\_lrp\_flex*, *plan\_milp\_lrp\_ref*, *plan\_milp\_schem*) ou estocásticos (*plan\_milp\_spp* e *plan\_milp\_spp\_ro*). Apresenta-se, na *Tabela 4.1*, o número de variáveis e restrições para cada um dos modelos de planeamento de processos.

As restantes variáveis no modelo com processos dedicados *plan\_milp\_lrp*, referentes aos fluxos de produção, às expansões de capacidade e às capacidades dos processos, às quantidades compradas e vendidas dos componentes nos diversos mercados, são contínuas e não-negativas.

O objectivo é o de maximizar o VLA da rede de processos contínuos dedicados, num horizonte multiperíodo, definindo-se os valores óptimos para:

- a expansão de capacidades e as capacidades dos processos, novos ou preexistentes, ao longo do horizonte de planeamento;
- fluxos materiais na rede de processos, em cada período de tempo;
- compras de matérias reagentes e vendas de produtos, em cada período.

No modelo de planeamento de processos a longo prazo *plan\_milp\_lrp* são necessários os dados referentes às disponibilidades e procuras de componentes, valores previsionais actualizados dos preços das matérias e produtos, bem como valores actualizados para os custos de operação e de investimento dos processos, em todos os períodos de tempo considerados ao longo do horizonte de planeamento.

De referir que o fluxo do produto principal em cada processo se apresenta como valor de referência, quer para definição da capacidade do processo, quer para determinação dos fluxos dos outros componentes associados ao processo, reagentes ou produtos.

**Tabela 4.1** Variáveis e restrições dos modelos para planeamento de processos.

Modelo	Variáveis binárias	Variáveis contínuas	Restrições
<i>plan_milp_lrp</i>	$NP.NT$	$3.NP.NT + 2.NM.NC.NT$	$NC.NT + 5.NP.NT + NP + NT$
<i>Plan_milp_lrp_flex</i>	$NP.NT$	$2.NP.NT + 2.NM.NC.NT + 2.NP.NC.NT$	$NC.NT + 3.NP.NT + 3.NP.NC.NT + NP + NT$
<i>plan_milp_lrp_schem</i>	$NP.NT$	$2.NP.NT + 2.NM.NC.NT + NP.NC.NT + 2.NP.NC.NS.NT$	$NC.NT + 3.NP.NT + 3.NP.NC.NT + NP + NT$
<i>plan_milp_lrp_ref</i>	$NP.NT$	$3.NP.NT + 2.NM.NC.NT + 2.NP.NC.NT + NP.NT.[NT.(NT+1)/2]$	$NC.NT + 5.NP.NT + 3.NP.NT.[NT.(NT+1)/2]$
<i>plan_milp_spp</i>	$NP.NT$	$2.NP.NT + 2.NM.NC.NT.NR + NP.NT.NR$	$NC.NT.NR + 3.NP.NT + NP.NT.NR + NP + NT$
<i>plan_milp_spp_ro</i>	$NP.NT$	$2.NP.NT + 2.NM.NC.NT.NR + NP.NT.NR + NR$	$NR + NC.NT.NR + 3.NP.NT + NP.NT.NR + NP + NT$

Considere-se o problema com processos dedicados em *plan\_milp\_lrp*, com um único período de tempo, correspondendo ao caso específico da definição da rede de processos por selecção dos processos a implementar. Mostra-se que este problema é NP-hard, verificando-se que o problema NP-completo de *Knapsack* é equivalente ao problema de admissibilidade (*feasibility*) associado com uma instância específica do problema em *plan\_milp\_lrp*, de forma similar à utilizada por Ahmed *et al.* [2000].

Seja a instância do problema *Knapsack*, dada como segue [Garey *et al.*, 1979]:

INSTÂNCIA: Conjunto finito  $P$ , para cada  $i \in P$  uma dimensão  $s_i \in \mathbb{Z}^+$  e um valor  $v_i \in \mathbb{Z}^+$ , e números inteiros positivos  $B$  e  $K$ .

QUESTÃO: Existe um subconjunto  $P' \subseteq P$  tal que  $\sum_{i \in P'} s_i \leq B$  e  $\sum_{i \in P'} v_i \geq K$ ?

Defina-se a instância do problema em *plan\_milp\_lrp*, assumindo-se que:

- o número de períodos de tempo  $NT=1$ ; deste modo, o índice correspondente  $t$  é redundante e pode desprezar-se;
- seja cada processo químico  $i \in P$ ; seja o conjunto de componentes químicos  $C = C_{reag} \cup \{p\}$  onde  $C_{reag}$  é o conjunto dos reagentes e  $p$  é o produto final para todos os processos;
- na função objectivo: anulam-se os custos de compra e venda dos componentes,  $\gamma_j = \Gamma_j = 0$ , para todo o  $j \in C$ ; anulam-se os custos de operação dos processos e os custos variáveis de expansão de capacidades,

$\alpha_i = \delta_i = 0$ , para qualquer  $i \in P$ ; os custos fixos de expansão fazem-se  $\beta_i = s_i$ ,  $\forall i \in P$ ;

- os limites às expansões de capacidades,  $QE_i^{Low} = 0$ ,  $QE_i^{Upp} = v_i$ ,  $\forall i \in P$ ;
- A capacidade inicial dos processos é nula,  $Q_{i0} = 0$ ,  $\forall i \in P$ ;
- Os coeficientes de balanço de massa para cada processo, na situação de produto único  $p$ ,  $\mu_{ip} = 1$ ,  $\forall i \in P$ ; para os reagentes, supõe-se um reagente único por processo, onde  $\mu_{i,j'} = 1$ ,  $i' \in P$ ,  $j' \in C$ , sendo os restantes coeficientes nulos;
- as disponibilidades dos reagentes não têm limitações,  $a_j^{Low} = -\infty$ ,  $a_j^{Upp} = +\infty$ ,  $\forall j \in C$ ; para o produto  $p$ , a disponibilidade de compra é nula,  $a_p^{Low} = a_p^{Upp} = 0$ ,  $p \in C$ ;
- as procuras dos reagentes são nulas,  $d_j^{Low} = d_j^{Upp} = 0$ ,  $\forall j \in C$ ; a procura do produto  $p$  está limitada,  $d_p^{Low} = K$ ,  $d_p^{Upp} = +\infty$ ,  $p \in C$ ;

Desprezando todos os constrangimentos e variáveis redundantes, a formulação do problema de planeamento de processos reduz-se ao problema *plan\_milp\_lrp\_knapsack*:

**Modelo *plan\_milp\_lrp\_knapsack*:**

$$\text{ap.1} \quad [\min] \Phi = \sum_{i=1}^{NP} s_i y_i$$

sujeito a

$$\text{ap.1-b} \quad W_i \leq y_i v_i, \quad \forall i$$

$$\text{ap.1-c} \quad K \leq \sum_{i=1}^{NP} W_i$$

$$\text{ap.1-d} \quad W_i \geq 0, \quad \forall i$$

$$\text{ap.1-e} \quad y_i \in \{0,1\}, \quad \forall i$$

Advém a seguinte proposição:

*Proposição 4. 1:* O problema de decisão *Knapsack* apresenta uma resposta “sim” se e só se o problema em *plan\_milp\_lrp\_knapsack* tem uma solução com valor da função objectivo no máximo  $B$ .

Suponha-se que o problema em *plan\_milp\_lrp\_knapsack* tem soluções  $(W_i^*, y_i^*) \forall i$ , com valor objectivo no máximo  $B$ ; pelos constrangimentos e função objectivo deste problema, tem-se

$$\text{ap.2} \quad \sum_{i=1}^{NP} s_i y_i^* \leq B$$

$$\text{ap.3} \quad \sum_{i=1}^{NP} v_i y_i^* \geq K$$

Pode construir-se um subconjunto  $P'$  de  $P$  referente aos valores das variáveis binárias com valor unitário,  $P' = \{i \in P : y_i^* = 1\}$ , o qual satisfaz as desigualdades do problema *Knapsack* e que assim apresenta uma decisão “sim”.

Suponha-se que o problema *Knapsack* tem uma resposta “sim”; então existe um subconjunto  $P'$  de  $P$  tal que:

$$\text{ap.4} \quad \sum_{i \in P'} s_i \leq B$$

$$\text{ap.5} \quad \sum_{i \in P'} v_i \geq K$$

Pode construir-se uma solução do problema em *plan\_milp\_lrp\_knapsack*, da seguinte forma: valores nulos para as variáveis  $y$  de decisão binária, excepto  $y_i^* = 1, i \in P'$ ; valores nulos para as variáveis contínuas  $W$ , excepto  $W_i^* = v_i, i \in P'$ ; uma solução construída desta maneira é uma solução admissível do problema em *plan\_milp\_lrp\_knapsack* e o valor da função objectivo é no máximo  $B$ .

Por conseguinte, qualquer instância do problema *Knapsack* é redutível ao problema de admissibilidade (*feasibility*) de determinar quando o problema de *plan\_milp\_lrp\_knapsack* apresenta uma solução no máximo igual a  $B$ , sendo essa redução claramente polinomial. Daí que a existência de um algoritmo polinomial para a resolução de *plan\_milp\_lrp\_knapsack* implicaria que  $P = NP$ , donde:

*Teorema 4.1:* O problema em *plan\_milp\_lrp\_knapsack* é NP-hard.

O problema em *plan\_milp\_lrp\_knapsack* constitui-se como uma instância específica do problema *plan\_milp\_lrp*, o qual também não será possível de resolução por um algoritmo polinomial, por inerência:

*Teorema 4.2:* O problema em *plan\_milp\_lrp* é NP-hard.

#### 4.1.2 O problema com processos flexíveis

O modelo *plan\_milp\_lrp\_flex* para planeamento de longo prazo com processos flexíveis, contínuos ou descontínuos, obtém-se do modelo *plan\_milp\_lrp* introduzindo novas restrições e definindo as taxas de produção relativas  $\rho$  e dos factores de capacidade  $Q$ , para se obter devidamente as novas (*NP.NC.NT*) quantidades associadas aos diferentes esquemas de produção flexível,  $\theta_{ijt}$ .

Pretende-se estabelecer uma correspondência, uma relação de equivalência, entre as respectivas funções objectivo, bem como do espaço de procura das soluções (definidos pelas restrições) dos modelos *plan\_milp\_lrp\_flex* e *plan\_milp\_lrp*. Deste modo, haverá também uma equivalência entre os valores obtidos para solução de cada um dos modelos, sendo que se deseja mostrar que o problema definido em

$plan\_milp\_lrp$  se encontra associado a uma instância específica do problema definido em  $plan\_milp\_lrp\_flex$ .

Considere-se a seguinte instância do problema em  $plan\_milp\_lrp\_flex$ , definindo-se que:

- em cada processo  $i$ , encontra-se apenas um esquema de produção (processos dedicados), com  $card(M_i) = 1$ ;
- a taxa de produção relativa, para o único componente principal  $j$  no processo  $i$ , é unitária, ou seja,  $\rho_{ij} = 1$ ; note-se que a relação (processo/componente principal) é unívoca, pois existindo um só esquema em cada processo, o produto principal respeitante ao processo é também único;
- os coeficientes de balanço material,  $\mu_{ij}$ , apresentam como redundante o índice referente ao componente principal  $j$ , pois a relação processo/componente principal é unívoca, reduzindo-se assim os índices de interesse;
- o número de horas  $H_{it}$  disponível para cada processo é total em cada período de tempo, de forma que, aplicando o factor de capacidade  $Q_{it}$  do processo, o resultado seja numérica e dimensionalmente igual à capacidade de produção no modelo  $plan\_milp\_lrp$ ;
- as restrições de limite ao investimento e de limite superior ao número de expansões são tornadas redundantes, assumindo-se valores de *bigbound* e de  $NEXP(i) = NT$ ,  $\forall i$ , respectivamente; isto apenas porque estas restrições não foram consideradas formalmente no modelo  $plan\_milp\_lrp$  descrito na Sec. 2.1, pois de outra forma ocorreria equivalência entre estas restrições nos dois modelos em estudo;
- ocorre equivalência nos restantes dados necessários à resolução, tanto do problema em  $plan\_milp\_lrp\_flex$  como em  $plan\_milp\_lrp$ , ou seja, nos valores de preços de compra e venda dos componentes químicos, nos custos de operação, nos custos fixos e variáveis de expansão de capacidades, nas capacidades de produção iniciais, nos limites superior e inferior para expansão de capacidades, nos limites superior e inferior tanto para a disponibilidade como para a procura dos componentes.

Existem grupos de restrições que são equivalentes em ambos os modelos em análise. Nomeadamente, o balanço às capacidades, os limites lógicos à expansão de capacidades, os limites inferior e superior aos valores de compra e venda de componentes. No entanto, para a composição do modelo  $plan\_milp\_lrp\_flex$  foram considerados ainda os seguintes grupos de restrições:

ap.6 
$$W_{jt} = \rho_{ij} \theta_{jt} \quad , \quad \forall i, j, t$$

ap.7 
$$\sum_{j \in M_i} \theta_{jt} \leq H_{ij} Q_{it} \quad , \quad \forall i, j, t$$

$$\text{ap.8} \quad W_{ijt} = \sum_{j' \in M_i} \mu_{ijj'} W_{ij't} \quad , \quad \forall i, j, t$$

Ora, sendo unitária a taxa de produção relativa,  $\rho_{ij} = 1$ , obtém-se da equação *ap.6* que

$$\text{ap.9} \quad W_{ijt} = \theta_{ijt} \quad , \quad \forall i, j, t$$

Também, sendo unívoca a relação (processo/componente principal), o produto principal respeitante ao processo é também único, reduzindo-se assim os índices  $i$  e  $j$  a um só índice. Desta maneira, substituindo a igualdade *ap.9* na equação de limite superior de produção *ap.7*, reduzindo-se o índice redundante pois  $\text{card}(M_i) = 1$ , calculando-se a capacidade de produção a partir do número de horas  $H_{it}$  disponível e do factor de capacidade, obtém-se um conjunto de restrições equivalentes ao existente em *plan\_milp\_lrp*, da forma

$$\text{ap.10} \quad W_{m,t} \leq Q_{it} \quad , \quad \forall i, t$$

Do conjunto de restrições em *ap.8*, pelo conjunto de argumentos descrito acima (cardinalidade de  $M_i$ , redundância dos índices  $i$  e  $j$ ) e pela equivalência nos coeficientes de balanço material  $\mu_{ijj'}$  relativamente aos correspondentes coeficientes  $\mu_{ik}$  em *plan\_milp\_lrp*, observa-se que a seguinte relação se mantém em *plan\_milp\_lrp*:

$$\text{ap.11} \quad W_{kt} \leq \mu_{ik} W_{m,t} \quad , \quad \forall k, i, t$$

Deste modo, o conjunto de variáveis  $W_{ijt}$  em *plan\_milp\_lrp\_flex* equivale às variáveis  $W_{kt}$  em *plan\_milp\_lrp*, pois estão sujeitas a um conjunto equivalente de restrições, logo de espaço de procura de soluções. De facto, são também equivalentes as relações de balanço material global, a cada componente em cada período de tempo.

Em consequência do exposto, as funções objectivo nos dois modelos são equivalentes, pois distinguem-se apenas nestas variáveis de fluxo. Os índices  $i$  e  $j$  nos correspondentes custos de produção são excessivos, reduzindo-se apenas a um índice, e equivalendo esses custos aos custos de produção em *plan\_milp\_lrp*.

Verificando-se equivalência entre os espaços de procura (restrições) e as funções objectivo de ambos os problemas, podemos afirmar que a instância específica do problema em *plan\_milp\_lrp\_flex* é equivalente ao problema em *plan\_milp\_lrp*. Donde o seguinte teorema:

*Teorema 4.3:* O problema em *plan\_milp\_lrp\_flex* é NP-hard.

### 4.1.3 O problema com esquemas de produção flexível

O modelo *plan\_milp\_lrp\_schem* aperfeiçoa o conceito de flexibilidade, é direccionado à programação de investimentos num horizonte de longo prazo em redes de processos, considerando processos dedicados ou flexíveis. Neste modelo, são individualizados os diferentes esquemas de produção disponíveis em cada processo, de maneira que os fluxos produzidos nos processos terão de satisfazer a agregação dos diversos esquemas considerados.

O intuito é o de estabelecer uma equivalência entre as funções objectivo e os espaços de procura das soluções dos modelos *plan\_milp\_lrp\_schem* e *plan\_milp\_lrp*.

Então, ocorrerá uma equivalência entre os valores das soluções de cada um dos modelos, pretendendo-se mostrar que o problema em *plan\_milp\_lrp* está associado a uma instância do problema em *plan\_milp\_lrp\_schem*.

Seja a instância específica do problema em *plan\_milp\_lrp\_schem*, assumindo-se que:

- em cada processo  $i$ , encontra-se apenas um esquema de produção (processos dedicados), com  $NS = \text{card}(PS(i)) = \text{card}(JM(i_o, s)) = 1$ ;
- a taxa de produção relativa para o (único) componente principal  $j$  no processo  $i$ , é unitária, ou seja,  $\rho_{ijs} = 1$ ; a relação (processo/esquema/ componente principal) é unívoca, pois existe um só esquema em cada processo, com um só produto principal, reduzindo-se assim o trio de índices de interesse a um só índice;
- os coeficientes de balanço material,  $\mu_{ijs}$ , apresentam como excessivo apenas o índice referente ao esquema  $s$ , pois é respeitante a todos os componentes  $j$  associados ao processo  $i$  e não apenas a um componente principal;
- a rede de processos supõe que cada componente secundário apenas ocorre em um processo/esquema de produção; desta forma, não são necessários constrangimentos adicionais de agregação de fluxos;
- o número de horas  $H_{it}$  disponível para cada processo é total em cada período de tempo, de forma que, aplicando o factor de capacidade  $Q_{it}$  do processo, o resultado seja numérica e dimensionalmente igual à capacidade de produção no modelo *plan\_milp\_lrp*;
- os grupos de restrições referentes ao limite ao investimento e ao limite ao número de expansões serão redundantes, definindo-se valores para os RHS de *bigbound* e de  $NEXP(i) = NT$ ,  $\forall i$ , respectivamente;
- ocorre equivalência nos restantes dados, quer no problema em *plan\_milp\_lrp\_schem*, quer em *plan\_milp\_lrp*: nos preços de compra e venda dos componentes, nos custos de operação, nos custos fixos e variáveis das expansões, nos valores para capacidade de produção inicial, nos limites (superior e inferior) para expansão de capacidades, nos limites (superior e inferior) para disponibilidades e procuras dos componentes.

Nos modelos em análise, ocorrem restrições que são equivalentes, concretamente, os grupos de restrições relativos: aos balanço às capacidades; aos limites (lógicos) à expansão de capacidades; aos limites (inferior e superior) às compras e vendas de componentes. Não obstante, a construção do modelo *plan\_milp\_lrp\_schem* exigiu que fossem considerados também os seguintes grupos de restrições:

ap.12

$$W_{ijt} = \sum_{s \in PS(i)} \rho_{ijs} \theta_{ijst}, \quad \forall i, j, t$$

ap.13

$$\sum_{j \in JM(i, s)} \sum_{s \in PS(i)} \theta_{ijst} \leq H_{ij} Q_{it}, \quad \forall i, j, t$$

$$\text{ap.14} \quad W_{ijt} = \sum_{s \in PS(i)} \mu_{ijs} W_{ij'st} \quad , \quad \forall i, j, t$$

Supondo unitárias a cardinalidade do conjunto de esquemas em cada processo,  $card(PS(i)) = 1$ , bem como da taxa de produção relativa,  $\rho_{ijs} = 1$ , e reduzindo-se o número de índices de interesse em (processo/esquema/componente principal), a equação *ap.12* reduz-se a

$$\text{ap.15} \quad W_{it} = \theta_{it} \quad , \quad \forall i, t$$

Ora, substituindo a igualdade *ap.15* na equação de limite superior de produção *ap.13*, supondo  $card(PS(i)) = card(JM(i_o, s)) = 1$ , reduzindo-se os índices redundantes, calculando-se a capacidade de produção a partir do número de horas  $H_{it}$  disponível e do factor de capacidade, ocorre um conjunto de restrições equivalentes ao existente em *plan\_milp\_lrp*, da forma (já descrita em *ap.10*)

$$W_{m,t} \leq Q_{it} \quad , \quad \forall i, t$$

Nas restrições em *ap.14*, dados os pressupostos acima referidos (cardinalidade unitária de  $PS(i)$ , redundância dos índices  $i, j'$  e  $s$ ), existência de um só agregado “processo/esquema/componente secundário”, e pela equivalência nos coeficientes de balanço material  $\mu_{ijs}$  relativamente aos coeficientes  $\mu_{ik}$  em *plan\_milp\_lrp*, verifica-se que se mantém em *plan\_milp\_lrp* o seguinte grupo de restrições (já descrito em *ap.11*):

$$W_{kt} \leq \mu_{ik} W_{m,t} \quad , \quad \forall k, i, t$$

Por conseguinte, o grupo de variáveis  $W_{ijt}$  em *plan\_milp\_lrp\_schem* equivale às variáveis  $W_{kt}$  em *plan\_milp\_lrp*, pois estão submetidas a um conjunto equivalente de restrições. Ainda, esta relação de equivalência mantém-se para as relações de balanço material global a cada componente.

Logo, são equivalentes as funções objectivo nos dois modelos, pois distinguem-se apenas na parcela onde constavam as variáveis de fluxo dos componentes principais: os índices  $i$  e  $j$  nos respeitantes custos de produção são redundantes, reduzindo-se apenas a um índice, e equivalendo esses custos aos custos de produção em *plan\_milp\_lrp*; observa-se que  $card(PS(i)) = card(JM(i_o, s)) = 1$ ; são também unitárias as taxas relativas de produção,  $\rho_{ijs} = 1$ .

Desta maneira, comprova-se a equivalência respectiva entre os espaços de procura e as funções objectivo dos problemas em questão, afirmando-se que a instância específica do problema em *plan\_milp\_lrp\_schem* é equivalente ao problema em *plan\_milp\_lrp*. Decorre daqui o teorema:

*Teorema 4.4:* O problema em *plan\_milp\_lrp\_schem* é NP-hard.

#### 4.1.4 O problema com reformulação de restrições

O modelo *plan\_milp\_lrp\_ref* para planeamento de processos apresenta um número de variáveis e de restrições que é superior ao do modelo *plan\_milp\_lrp*, com a estratégia de reformulação a permitir obter modelos MILP com melhores relaxações lineares no método de *branch-and-bound*.

Introduziram-se novas restrições e novas variáveis (de “produção de capacidade” cumulativa), mas utiliza-se integralmente o modelo em *plan\_milp\_lrp*, seja na função objectivo, seja nas restrições, coincidindo também nas restantes variáveis utilizadas.

Os dados necessários à resolução de ambos os modelos, *plan\_milp\_lrp\_ref* e *plan\_milp\_lrp* são equivalentes. A nova formulação *plan\_milp\_lrp\_ref* requer exactamente os mesmos dados que a do problema original *plan\_milp\_lrp*, diferenciando-se deste último devido ao aumento significativo do número de variáveis contínuas (as “produções de capacidade” cumulativas) e o número de restrições. De facto, manter-se-ão a função objectivo, os balanços de capacidades em cada período e os limites de expansão (“produção”) de capacidade. As restrições referentes às variáveis de fluxos materiais, de compras e vendas mantêm-se inalteráveis, sendo equivalentes nos dois modelos. No entanto, são introduzidas os seguintes  $[3.NP.NT.NT.(NT+1)/2]$  grupos de restrições:

$$\text{ap.16} \quad \varphi_{it\tau} \leq QE_{it} \quad , \quad \forall i, t, \tau \geq t$$

$$\text{ap.17} \quad \varphi_{it\tau} \leq QE_{it\tau}^{accum} y_{it} \quad , \quad \forall i, t, \tau \geq t$$

$$\text{ap.18} \quad \sum_{\tau=1}^t \varphi_{i\tau} \geq (W_{m,t} - Q_{i0}) \quad , \quad \forall i, t, \tau \leq t$$

É um facto que as funções objectivo, sendo coincidentes nos seus coeficientes (custos e preços) e na definição das suas variáveis, são equivalentes em ambos os modelos em análise.

No entanto, o espaço de procura em *plan\_milp\_lrp\_ref* encontra-se reduzido pela introdução das novas restrições sobre a expansão (“produção”) de capacidades. Estas restrições não afectam o espaço de procura das variáveis inteiras  $\{0,1\}$ , o qual se mantém inalterado, mas pretendem aproximar os valores das variáveis de expansão  $QE$  dos valores das necessidades cumulativas dadas pela “procura de capacidade”. Assim sendo, não existe equivalência entre os respeitantes espaços de procura de soluções admissíveis nos dois modelos.

Consequentemente, segundo a metodologia que tem vindo a ser seguida, não se pode afirmar a equivalência entre os problemas em *plan\_milp\_lrp\_ref* e *plan\_milp\_lrp*, ou alguma instância destes.

O que não significa que o problema em *plan\_milp\_lrp\_ref* não seja NP-hard, mas verificou-se empiricamente que é de resolução mais eficiente pelo método do *branch-and-bound* do que o problema em *plan\_milp\_lrp*.

#### 4.1.5 Os problemas estocástico e robusto

O problema modelado em *plan\_milp\_spp* consiste num problema estocástico de planeamento de processos sob incerteza, numa abordagem em duas fases, 2SSP. A incerteza ocorre nos coeficientes RHS sobre as procuras e disponibilidades dos componentes químicos, as quais são consideradas variáveis aleatórias com distribuições conhecidas. Por conseguinte, as respectivas variáveis de compra e venda de componentes, bem como as variáveis de fluxos materiais assumem também um carácter aleatório. Estas variáveis serão determinadas em segunda fase, são as designadas variáveis de recurso, enquanto que em primeira fase são definidas as

variáveis determinísticas de projecto: capacidades, expansões de capacidades e decisões binárias sobre a expansão (ou não) dessas capacidades.

Na função objectivo considera-se o valor esperado referente às variáveis de recurso, sendo o valor respeitante a cada cenário ponderado pela probabilidade correspondente. As variáveis de recurso apresentam uma dimensão adicional, relativamente ao modelo em *plan\_milp\_lrp*, expressa através de um índice adicional,  $r$  (*random*).

Quanto às restrições de primeira fase, estas consideram os grupos de restrições no âmbito dos balanços de capacidade, de limites (superior e inferior) lógicos às expansões, bem como os limites superiores ao investimento e ao número de expansões, sendo coincidentes com os correspondentes grupos de restrições em *plan\_milp\_lrp*. Em segunda fase, são tratadas as restrições relativas aos parâmetros aleatórios, considerando-se um conjunto adicional de restrições relativas aos  $NR$  cenários aleatórios, nos grupos de: limites superiores de capacidades; balanços globais a cada componente: limites, superiores e inferiores, às procuras e disponibilidades de materiais.

Pretende-se estabelecer uma relação de equivalência entre as funções objectivo e os espaços de procura dos modelos *plan\_milp\_spp* e *plan\_milp\_lrp*. Consequentemente, haverá também equivalência entre os valores obtidos em cada um dos modelos, desejando-se verificar que o problema em *plan\_milp\_lrp* está associado a uma instância específica do problema em *plan\_milp\_spp*.

Considere-se a instância do problema em *plan\_milp\_spp*, determinando-se que:

- ocorre apenas um cenário aleatório (problema determinístico), com probabilidade unitária,  $NR = prob(r) = 1$ ;
- para os materiais reagentes  $j$ , em cada processo  $i$ , os valores dos coeficientes de balanços materiais  $\eta_{ij} = \mu_{ij}$ ; para os produtos,  $\mu_{ij} = \mu_{ij}$  (em *plan\_milp\_spp* relativamente a *plan\_milp\_lrp*), em especial  $\mu_{ij'} = 1$ , se  $j'$  representa os produtos principais;
- os valores para os RHS, nos grupos de restrições referentes ao limite ao investimento e ao limite ao número de expansões, serão, respectivamente, *bigbound* e  $NEXP(i) = NT$ ,  $\forall i$ ;
- os limites (superior e inferior) para disponibilidades e procuras dos componentes apresentam, reduzindo-se o índice unitário  $r$ , valores equivalentes aos valores determinísticos em *plan\_milp\_lrp*;
- ocorre equivalência nos restantes dados não aleatórios, relativamente aos problemas em *plan\_milp\_lrp\_pp* e em *plan\_milp\_lrp*; nomeadamente, nos preços de compra e venda, nos custos de operação e nos custos (fixos e variáveis) das expansões de capacidades, nas capacidades iniciais, nos limites (superior e inferior) para expansão de capacidades.

Supondo-se um único cenário aleatório (com probabilidade unitária), o índice  $r$  nas variáveis de recurso é redundante, não estabelece qualquer distinção entre as correspondentes variáveis estocásticas e determinísticas. Ainda, considera-se que a minimização do simétrico do VLA corresponde à sua maximização.

Donde, as funções objectivo em  $plan\_milp\_spp$  e  $plan\_milp\_lrp$  são equivalentes.

As restrições de primeira fase são equivalentes nos dois modelos.

Quanto às restrições de segunda fase, verificando-se a redundância do índice unitário  $r$ , estas são equivalentes às restrições do programa determinístico. Então:

*Teorema 4.5:* O problema em  $plan\_milp\_spp$  é NP-hard.

Ainda, O problema em  $plan\_milp\_spp\_ro$  consiste num programa de planeamento de processos estocástico em duas fases, 2SSP, no âmbito de uma optimização robusta, onde se penaliza a variabilidade dos custos de recurso. A incerteza ocorre nos termos independentes, propagando-se às variáveis de fluxos, compras e vendas que são tratadas em segunda fase. A função objectivo considera o valor esperado respeitante àquelas variáveis de recurso, adicionando-se uma parcela de penalização dos desvios positivos (superiores à média) dos custos, ponderados pelos valores das probabilidades de cada cenário  $r$ . Evitando-se a utilização da variância, que é uma medida simétrica do risco, evita-se também a introdução de não-linearidades no modelo.

O modelo  $plan\_milp\_spp\_ro$  obtém-se do modelo  $plan\_milp\_spp$  por adição da parcela de penalização da variabilidade na função objectivo, bem como por introdução de um grupo de NR restrições referentes à definição da métrica de variabilidade em cada cenário aleatório, os desvios positivos  $dsvp_r$ .

A finalidade consiste em obter uma relação de equivalência entre as funções objectivo e os espaços de procura dos modelos  $plan\_milp\_spp\_ro$  e  $plan\_milp\_spp$ . Deste modo, ocorrerá equivalência entre as soluções de cada um dos modelos, pretendendo-se provar que o problema  $plan\_milp\_spp$  está associado a uma instância do problema em  $plan\_milp\_spp\_ro$ .

Seja a instância específica do problema em  $plan\_milp\_spp\_ro$ , definindo-se que:

- o parâmetro de penalização é nulo,  $\omega = 0$  (optimização não-robusta);
- ocorre apenas um cenário aleatório (problema determinístico), com probabilidade unitária,  $NR = prob(r) = 1$ ;
- todos os restantes dados são equivalentes para ambos os problemas.

Dado que o parâmetro de penalização é nulo, ocorre equivalência entre a função objectivo do problema  $plan\_milp\_spp\_ro$  e a do problema em  $plan\_milp\_spp$ .

No respeitante ao espaço de procura das soluções admissíveis, verifica-se a equivalência entre todos os grupos de restrições e variáveis, observando-se no modelo  $plan\_milp\_spp\_ro$  a introdução de um grupo de restrições relativas à definição do desvio  $dsvp$ . Atendendo a que se considera apenas 1 (um) cenário aleatório com probabilidade unitária, o valor deste desvio é nulo, obtendo-se assim um problema determinístico que se verificou anteriormente pertencer à classe NP-hard.

*Teorema 4.6:* O problema em  $plan\_milp\_spp\_ro$  é NP-hard.

## 4.2 Análise probabilística de heurísticas para o dimensionamento e sequenciamento de processos descontínuos

Na presente secção revêem-se e analisam-se os diferentes modelos direccionados ao dimensionamento e sequenciamento de processos *batch*, anteriormente descritos na Sec. 2.2, nomeadamente, os de formulação MILP que assumiam a política de espera nula (*zero wait*, ZW): quer perante campanhas de produto único (*single product campaign*, SPC) ou produtos múltiplos (*multiple product campaign*, MPC), quer assumindo processo único (*single machine*) ou processos múltiplos (*multiple machine*) em cada estágio.

Procede-se a um breve enquadramento da complexidade computacional dos problemas aqui em análise (*batch\_milp\_fssz*: *Flowshop/ Single machine/ SPC/ Zero wait*; *batch\_milp\_fmsz*: idem, mas *multiple machine*; *batch\_milp\_fmz*: agora *single machine* mas MPC; *batch\_milp\_fmmz*: *multiple machine* e MPC), para os quais se prevê que sejam todos da classe NP-hard, dada a equivalência que se pode inferir entre instâncias específicas dos diferentes problemas, conforme se descreve no Apêndice 4-A.

Consequentemente, propõe-se o desenvolvimento de procedimentos heurísticos, de aproximação ao valor óptimo da função objectivo dos problemas, ou de construção da solução binária óptima respeitante. Realiza-se uma análise probabilística das heurísticas desenvolvidas, avaliando os desvios do valor óptimo e da solução binária óptima, bem como comparando a utilização de tempo computacional.

Apresentam-se resultados computacionais, não só relativamente à análise de heurísticas, mas também no intuito de certificar a modelação e a implementação numérica dos diferentes problemas estudados.

Na *Tabela 4.2* apresentam-se o número de variáveis (binárias e contínuas) e restrições dos referidos modelos, atendendo aos números considerados de produtos  $N$ , de estágios  $M$ , de dimensões discretas  $NS(j)$  e de número de equipamentos  $NP(j)$  em paralelo em cada estágio. Sem perda de generalidade, suponha-se que os valores superiores para os números de dimensões,  $NS$ , e de processos,  $NP$ , não dependam do estágio  $j$  que se considere.

**Tabela 4.2** Variáveis e restrições dos modelos para dimensionamento e sequenciamento de processos descontínuos analisados.

Modelo	Variáveis binárias	Variáveis Contínuas	Restrições
<i>batch_milp_fssz</i>	$M.NS$	$N$	$M.N + M + 1$
<i>batch_milp_fmsz</i>	$M.NS.NP$	$2.N + N.M.NS.NP$	$M.N + M + 1 + 2.M.N + M.N.NS.NP$
<i>batch_milp_fmz</i>	$M.NS$	$N + N^2$	$M.N + 2.M + 3.N$
<i>batch_milp_fmmz</i>	$M.NS.NP$	$N + N^2$	$M.N + 2.M + 3.N$

Numa primeira abordagem, observa-se que o número de equipamentos  $NP$  possível em cada estágio vem acrescentar uma outra ordem de grandeza ao número de variáveis binárias; tal ordem de grandeza deve-se também ao número de dimensões discretas  $NS$  consideradas em cada estágio. Em ambiente *flowshop*, o número de estágios  $M$  é, em termos práticos, relativamente reduzido, não sendo ultrapassada no estudo numérico a ordem da dezena. Quanto ao número de variáveis contínuas, este depende em grande parte do número de produtos  $N$ , sendo esta relação quadrática no âmbito dos problemas que consideram uma política de campanha MPC; dada a significância do número de produtos  $N$  sobre a dimensão dos problemas verifica-se que, no entanto, este parâmetro não influi sobre o número de variáveis binárias. Quanto ao número de restrições, depende linearmente dos números de produtos  $N$  e de estágios  $M$ , sendo da mesma ordem de grandeza que o número de variáveis binárias nos problemas que consideram *single machine*, exceptuando-se o problema *batch\_milp\_fmsz*, onde o número de restrições é da mesma ordem de grandeza do número de variáveis contínuas. Estas observações são válidas, obviamente, na gama de parâmetros utilizados para os exemplos numéricos.

#### 4.2.1 O modelo com processo único e em campanha de produto único

O dimensionamento óptimo da unidade de produção com estágios de processo único (*single machine*), em ambiente multiproducto (*flowshop*) operando em campanha de produto único (*single product campaign*, SPC) e com espera nula (*zero wait*, ZW), pode obter-se do modelo *batch\_milp\_fssz*, que trata as dimensões dos equipamentos de forma discreta (*Flowshop/ Single machine/ SPC/ ZW*). Neste enquadramento do problema, pretende-se satisfazer as procuras dos diversos produtos no âmbito do horizonte temporal, minimizando o custo total do equipamento necessário. Atendendo às políticas de produção SPC e de transferência ZW, o tempo de ciclo  $CT_i$  para cada produto é determinado pelo máximo dos tempos de processamento  $\tau_{ij}$  do produto nos diversos estágios. Por conseguinte, o número de ciclos de produção encontra-se limitado pela dimensão do horizonte de planeamento e, para que as quantidades produzidas observem os constrangimentos temporais, é necessário que o sistema de produção seja dimensionado em concordância. O balanceamento ocorre entre os tempos de produção disponibilizados para cada produto,  $t_i$ , e o dimensionamento discreto dos estágios, o qual se associa à especificação das variáveis binárias,  $y_{js}$ . No intuito de satisfazer o horizonte temporal e não optar pelo aumento da dimensão dos lotes de produção, um aumento do tempo de campanha para um dado produto traduz-se em menor tempo para os outros produtos, havendo que aumentar a capacidade dos estágios. Então, implementando maiores capacidades de produção nos estágios limitantes, os quais definem os ciclos de tempo para os produtos, permitirá obter um menor investimento que decorre da diminuição da dimensão dos lotes de produção, os quais se repetem em maior número.

Os dados necessários são: o número total de estágios  $j$ ,  $M$ , e de produtos  $i$ ,  $N$ ; o número de dimensões discretas  $s$  dos equipamentos em cada estágio  $j$ ,  $NS$ , os seus correspondentes valores discretos normalizados,  $dv_{js}$ , e os custos respeitantes,  $c_{js}$ ; as quantidades procuradas,  $Q_i$ , ao longo do horizonte temporal,  $H$ ; as especificações técnicas, relativas aos factores de dimensão,  $S_{ij}$ , e aos tempos de ciclo,  $CT_i$ , estes últimos decorrentes dos tempos de processamento  $\tau_{ij}$ . Do cálculo de optimização efectuado, o qual pretende a minimização do total dos custos de investimento, determinam-se os tempos de produção em campanha SPC para cada

produto,  $t_i$ , bem como decorre o dimensionamento discreto de cada processo (único em cada estágio), associado às variáveis binárias,  $y_{js}$ .

Apresenta-se na *Tabela 4.3* os números de variáveis, binárias e contínuas, bem como de restrições, para os exemplos numéricos (EX1A a EX6A) nos quais se avaliou o comportamento do modelo em *batch\_milp\_fssz*, referindo-se o tempo médio necessário à resolução óptima.

Para resolução de referência em ambiente GAMS, utilizou-se o otimizador OSL (“OPTION SOLVER = OSL;”). Observa-se que os tempos de resolução (valores médios, relativos a cada instância aleatória) necessários à obtenção da solução óptima vão aumentando, sendo esse aumento mais significativo quando o número de variáveis binárias e restrições se aproxima e ultrapassa a centena, pois neste modelo o número de variáveis contínuas não é relevante. No caso do exemplo EX3A, ocorre uma descontinuidade desta observação (tempo de execução apresentado em *itálico*), que se explica pelo facto de numerosas instâncias consideradas neste exemplo não serem admissíveis, vendo-se terminada a execução do *solver* sem ser desenvolvido todo o cálculo necessário até à garantia da optimalidade. A limitada experiência conduzida para este modelo resulta do facto de que os últimos dois exemplos (EX5A e EX6A) descritos na *Tabela 4.3* já não serem resolúveis em tempo útil (3600 segundos) para o modelo em processos múltiplos por estágio, como se confirma da *Tabela 4.5*.

**Tabela 4.3** Variáveis e restrições em exemplos numéricos (série A) para o problema *batch\_milp\_fssz*.

Exemplo numérico	Parâmetros (N, M, NS)	Variáveis binárias	Variáveis contínuas	Restrições	Tempo <i>solver</i> (s)
EX1A	2, 3, 4	12	2	9	0.23
EX2A	3, 4, 5	20	3	17	0.36
EX3A	5, 4, 6	24	5	25	<i>0.28</i>
EX4A	6, 4, 10	40	6	29	0.76
EX5A	8, 5, 12	60	8	46	1.03
EX6A	10, 5, 15	75	10	56	2.17

Descreve-se o desenvolvimento sistemático de procedimentos de aproximação, de carácter heurístico, que visam a obtenção de soluções admissíveis de qualidade satisfatória em instâncias não extremas, sendo reportada a usual análise probabilística aos desvios (na função objectivo, na solução binária, aos tempos de resolução) resultantes do procedimento heurístico.

A aproximação trivial desenvolvida (conforme procedimento APROX0, na descrição do algoritmo em Apêndice 4-A), de proposta inicial de volume mínimo necessário em cada estágio seguida de um número limitado (até 3) de alterações, não se mostrou suficientemente robusta: para o exemplo EX1A funcionava bem (68% de soluções óptimas na aproximação inicial, aumentando para 96% após 1 alteração sobre essa aproximação, sendo as restantes soluções admissíveis mas não óptimas); para o exemplo EX2A revelava-se relativamente satisfatória (9% óptimos na aproximação inicial, aumentando para 31% após 1 alteração, havendo ainda 3% de soluções admissíveis subóptimas); para o EX3A (apenas 1% de valores óptimos na aproximação inicial) e seguintes (da aproximação inicial, apenas configurações

inadmissíveis, sempre aproximadas por defeito), o desacerto era sistemático, com o número de desvios nas variáveis binárias a suplantarem significativamente o número de alterações previstas (até 3).

De forma sistemática, pretende-se construir soluções admissíveis (procedimento APROX1 no algoritmo), pois a aproximação trivial conduzia genericamente a soluções inadmissíveis. Desenvolveram-se procedimentos alternativos para construção das desejadas soluções admissíveis, construção baseada em diferentes pressupostos, nomeadamente: na proporcionalidade entre os tempos de campanha de cada produto e as respectivas quantidades procuradas (APROX1A); partindo de solução inadmissível, selecção das dimensões a aumentar dirigida pela razão entre as variações nos tempos e nos custos que a alteração de dimensões provoca (APROX1B).

A primeira abordagem, APROX1A, baseava-se na selecção das dimensões discretas de forma a que os tempos de campanha estimados para cada produto fossem proporcionais às respectivas quantidades procuradas. Dividindo-se o tempo total disponível,  $H$ , pelas diferentes campanhas (de produto único), obtêm-se sempre soluções admissíveis, o que se comprova analiticamente na descrição do algoritmo, em Apêndice 4-A, bem como se verifica numericamente. Não obstante, estas soluções admissíveis apresentam desvios significativos, sempre por excesso de custos: para os exemplos numéricos analisados (EX1A a EX6A), os excessos de custos ocorriam entre 20 e 45%. Esta elevada discrepância deve-se à não consideração dos parâmetros técnicos na elaboração deste procedimento, os quais estabelecem a dificuldade relativa entre a produção das diferentes campanhas (produtos), especificamente, os valores dos tempos de processamento  $\tau_{ij}$  de cada produto em cada estágio, bem como os respeitantes factores de dimensão  $S_{ij}$ .

A aproximação alternativa APROX1B para construção de solução admissível enquadra-se no âmbito duma procura local iterada (*iterated local search*), pois: constitui um procedimento que altera sucessivamente a solução binária inadmissível por aumento da dimensão do equipamento num dado estágio; seleccionando esse estágio de acordo com avaliação efectuada das variações nos tempos e nos custos que a referida alteração implica; a alteração efectua-se no sentido do aumento de dimensão que melhor razão daquelas variações apresenta; o procedimento é finito, terminando com a obtenção de solução admissível ou após um número fixo de iterações. Os resultados obtidos para os exemplos numéricos em estudo são bem melhores do que os conseguidos através da aproximação proporcional APROX1A, sendo considerados genericamente satisfatórios, conforme Tabela 4.4, dado que: obtêm-se soluções admissíveis em todas as instâncias em que tal era expectável; obtêm-se os valores óptimos, da solução binária e do valor da função objectivo, numa parte significativa das instâncias, que varia principalmente entre 30% e 70% das instâncias; quanto ao valor da função objectivo (custo total de investimento a minimizar) das soluções subóptimas obtidas, os desvios por excesso são genericamente menores que 2.4%; relativamente às soluções binárias resultantes, enquanto uma fatia significativa corresponde à solução óptima, as soluções subóptimas usualmente diferem em apenas uma dimensão, mas em simultaneidade de estágios, ocorrendo assim situações de desvio duplo: um desvio binário positivo associado a um desvio binário negativo.

**Tabela 4.4** Desvios e tempos de execução para heurística no âmbito do problema *batch\_milp\_fssz*.

Exemplo numérico	Tempo <i>solver</i> (s)	Tempo heurística (s)	Desvios função objectivo	Desvios solução binária
EX1A	0.23	0.0006	91./ 0./ 6.8/ 4.4	91./ 2.22
EX2A	0.36	0.0028	72./ 8./ 2.4/ 1.7	72./ 1.96
EX3A	0.28	0.0005	73./ 56./ 1.2/ 0.9	73./ 1.58
EX4A	0.76	0.0016	50./ 4./ 1.5/ 1.4	50./ 1.79
EX5A	1.03	0.0020	30./ 0./ 1.5/ 1.5	30./ 2.43
EX6A	2.17	0.0056	53./ 0./ 1.1/ 0.9	53./ 2.00

Apresentam-se na *Tabela 4.4* os estimadores (formulados em Apêndice 4-B) relativamente aos desvios nos valores da função objectivo e na solução binária, resultantes da conjugação dos diversos procedimentos de aproximação descritos, respectivamente:

- na função objectivo — percentagem de instâncias admissíveis estimadas de forma exacta/ percentagem de instâncias inadmissíveis/ desvio percentual médio para as soluções subóptimas/ desvio padrão aos desvios percentuais médios;
- na solução binária — percentagem de instâncias com solução binária exacta/ média do número de desvios na solução binária para as soluções subóptimas.

Ora, tendo-se desenvolvido abordagens para a obtenção de soluções admissíveis, obviamente no âmbito das instâncias em que tal era expectável, e observando-se também a optimalidade de uma parte significativa dessas instâncias, não se preconiza a melhoria das soluções subóptimas obtidas, dado que o nível de aproximação atingido é considerado satisfatório.

Descrevem-se ainda alguns aspectos considerados implicitamente na formulação do constrangimento temporal,  $H$ , directamente do tempo de produção  $t_i$  de cada produto, bem como da sua relação com o tempo de ciclo respectivo,  $CT_i$ : desprezaram-se os tempos de folga ( $SL_{ikj}$ ) que pudessem existir entre as campanhas de produção dos diferentes produtos; desprezou-se a existência de “cabeça” (*head*) e “cauda” (*tail*) em cada campanha de produto único; supõe-se a não sobreposição das produções de campanhas consecutivas, de diferentes produtos  $i$  e  $k$ , iniciando-se a produção do produto  $k$  apenas depois de finalizar a produção de  $i$ . Estes aspectos revelam-se de importância fundamental, nomeadamente quando se pretende comparar ou equivaler instâncias entre as diferentes modelações aqui analisadas.

Refira-se que o modelo em estudo *batch\_milp\_fssz* (considerando unitários tanto o número de produtos em cada campanha como o número de processos em cada estágio) apresenta algumas insuficiências, como: a suposição de processo único por estágio, que provoca que numerosas instâncias se mostrem impossíveis de satisfazer, nomeadamente, logo que o número  $N$  de produtos  $i$  seja mais elevado, ou que as quantidades requeridas  $Q_i$  sejam superiores; a desconsideração dos tempos de limpeza, especialmente aqueles necessários às mudanças de campanhas de produção, que embora tenham apenas um efeito relativo de “cauda”, poderiam influir

criticamente no caso de uma solução próxima dos limites temporais; o facto de não se explicitar (intermediamente) o número de ciclos de produção  $n_i$  para cada produto, bem como não exigindo a sua integralidade, poderá traduzir-se marginalmente na apresentação de soluções óptimas impraticáveis, mas de efeito pouco significativo.

Do exposto, concluiu-se da significância relativa deste modelo, embora seja importante a sua consideração aqui, pois dele são generalizados os modelos que a seguir são estudados. Também, a solução deste modelo pode servir como estimativa inicial para a obtenção de modelos mais complexos, dado ser caracterizada por um tipo de resolução relativamente acessível.

#### 4.2.2 O modelo com processos múltiplos e em campanha de produto único

O dimensionamento óptimo da unidade de produção descontínua, com processos múltiplos em cada estágio (*multiple machine*), operando em campanha de produto único (SPC) e seguindo uma política de espera nula (ZW), com uma modelação discreta das dimensões dos processos, pode obter-se por resolução do modelo *batch\_milp\_fmsz* (*Flowshop/ Multiple machine/ SPC/ ZW*).

Os dados para a formulação deste problema são coincidentes com os do modelo anterior, de processo único, aos quais se acrescenta: o número máximo  $NP$  de processos (de iguais dimensões) em paralelo nos estágios; os limites superiores (*upper bounds*) para as variáveis semicontínuas  $nc_{ijsp}$ , que poderão ser estimados a partir dos parâmetros técnicos do problema.

O número de variáveis binárias,  $y_{jsp}$ , onde  $p$  representa o número de equipamentos implementados em cada estágio, aumenta directamente com os números de dimensões discretas,  $NS$ , e de unidades permitidas em paralelo,  $NP$ . O número de variáveis semicontínuas  $nc_{ijsp}$  é substancial, tal como é significativo o número de restrições introduzidas para correcta especificação dessas variáveis. Consequentemente, este modelo é de uma dimensão muito superior ao do correspondente modelo a processo único.

O modelo em *batch\_milp\_fmsz* permite instalar unidades em paralelo nos diferentes estágios, pressupondo que os tempos de ciclo dos produtos possam diminuir significativamente com a redução dos tempos de processamento nos estágios em que tal aconteça. Este aspecto revela-se importante, caso exista uma disparidade significativa entre os tempos de processamento de um produto ao longo dos diversos estágios, especialmente se ocorrer um valor elevado relativamente ao conjunto dos outros tempos de processamento.

Este modelo permite balancear os custos entre a implementação em cada estágio de um só processo com elevada capacidade, ou dois ou mais processos de menor dimensão, assumindo a especificação dos custos das dimensões normalizadas. Deste modo, a solução apresentando um só processo em cada estágio também é considerada, mas também comparada com outras configurações alternativas, sendo seleccionada aquela que apresentar o menor custo.

Uma configuração trivial para o sistema descontínuo supõe a implementação de estágios simples, assegurando-se a produção das quantidades procuradas com o dimensionamento efectuado através das capacidades discretas que observem aqueles constrangimentos. Não obstante, o crescimento futuro das procuras pode diminuir a significância desta abordagem, requerendo-se a implementação em paralelo de unidades processuais nos estágios sujeitos a maior constrangimento temporal. Ainda, tal configuração poderá constituir-se como uma solução inicial a ser melhorada em passos sucessivos.

Repare-se que o problema em processo único *batch\_milp\_fssz* pode constituir-se como um caso específico do problema de processos múltiplos *batch\_milp\_fmsz*, supondo neste último que o número de processos em cada estágio se restringe a  $NP = 1$ . Neste caso, comprova-se em Apêndice 4-A a equivalência das variáveis, função objectivo e restrições nos dois modelos, pelo que o pressuposto de que o modelo *batch\_milp\_fssz* seja NP-hard implicará que o modelo *batch\_milp\_fmsz* também seja NP-hard.

No sentido de comparar numericamente os modelos em análise, procedeu-se à seguinte analogia entre os modelos *batch\_milp\_fssz* (*Flowshop/ Single machine/ SPC/ ZW*) e *batch\_milp\_fmsz* (idem, mas *multiple machine*): restringiu-se, neste último modelo, o número de processos por estágio a  $NP = 1$ ; efectuaram-se duas séries de 100 amostras aleatórias, com ambos os modelos; compararam-se os resultados obtidos, que se revelaram iguais. Deste modo, não só se verificou a indicação teórica, como se certificou a validade da resolução numérica.

Ora, seja a situação problemática de dimensionamento e sequenciamento de um sistema com processos descontínuos de dimensões discretas, operando com as políticas SPC e ZW. Desta forma, o tempo de campanha de cada produto,  $t_i$ , estará directamente relacionado com o respectivo tempo de ciclo,  $CT_i$ . Pretende-se esclarecer em que casos haverá interesse pela implementação de processos em paralelo, determinando os estágios onde tal ocorrerá.

Um caso acontece na hipótese de que o número  $N$  de produtos  $i$  se alargue, ou também que as respectivas quantidades procuradas  $Q_i$  aumentem; assim, a necessidade de capacidade instalada crescerá e, para o caso de processo único, o problema *batch\_milp\_fssz* será impossível quando a selecção da maior dimensão de equipamento em todos os estágios,  $dv(j, NS)$ , não permitir satisfazer o horizonte temporal  $H$ . A adopção de processos múltiplos nos diversos estágios, conforme modelada em *batch\_milp\_fmsz*, permite responder satisfatoriamente a este caso hipotético, apresentando soluções óptimas admissíveis relativamente aos exemplos numéricos estudados. Mesmo perante um caso extremo de grande necessidade de produção, a generalização do número de processos  $NP$  implementados em paralelo poderá dar resposta adequada. Nos exemplos numéricos analisados, limitou-se o número de processos ao máximo de 3 processos, pois verificou-se que tal valor não era limitativo para a obtenção da solução óptima: nos exemplos resolvidos, o número óptimo de processos em paralelo nunca ultrapassou 2 processos.

Outro caso hipotético ocorre quando é admissível considerar uma solução de processo único, mas a solução em processos múltiplos apresentar um menor custo: especificando uma dimensão menor para um dado estágio, implementa-se um processo duplo, nesse ou noutro estágio, desde que o diferencial de custos seja vantajoso, verificando-se obviamente a satisfação das restrições do modelo perante os parâmetros da instância; a redução de dimensões pode ocorrer simultaneamente em mais do que um estágio, até mesmo em todos os estágios, observando-se a condicionante descrita (diferencial de custos entre as soluções admissíveis).

Apresenta-se na *Tabela 4.5* os parâmetros caracterizadores das instâncias numéricas submetidas a optimização, analisando-se subsequentemente os resultados obtidos.

**Tabela 4.5** Variáveis e restrições em exemplos numéricos (série A) para o problema *batch\_milp\_fmsz*.

Exemplos numéricos	Parâmetros ( $N, M, NS, NP$ )	Variáveis binárias	Variáveis contínuas	Restrições	Tempo solver (s)
EX1A	2, 3, 4, 2	24	52	70	0.30
EX2A	3, 4, 5, 3	60	186	221	10.53
EX3A	5, 4, 6, 3	72	370	425	101.67
EX4A	6, 4, 10, 3	120	732	797	405.99
EX5A	8, 5, 12, 3	180	1456	1566	(4098.31)
EX6A	10, 5, 15, 3	225	2270	2406	(15142.36)

Dado o elevado tempo de resolução exigido para a obtenção da solução inteira óptima para cada amostra, em alguns exemplos numéricos, decidiu impor-se como limite máximo para o tempo de execução uma hora, especificando-se “OPTION RESLIM = 3600;”. Também, o número de amostras considerado em cada exemplo numérico passou a ser de 30 amostras, em vez das 100 amostras inicialmente consideradas. Quanto aos exemplos não resolúveis naquele limite temporal, eles são relevados colocando-se os tempos de execução entre parênteses na coluna respeitante. Comparando com os tempos relativos à optimização destes exemplos segundo o modelo de processo único, *batch\_milp\_fmsz*, verifica-se que ocorre um aumento desmesurado do tempo de resolução para problemas com número de variáveis binárias comparáveis (o modelo de processos múltiplos apresenta, tipicamente e para o mesmo exemplo numérico, o triplo de variáveis binárias que o modelo a processo único). Não obstante tal disparidade, considerando-se problemas com semelhante número de variáveis binárias, infere-se que a elevada diferença existente se deve ao elevado número de variáveis contínuas e de restrições consideradas no modelo em *multiple machine* (da ordem dos milhares e dezenas de milhares, para grande parte dos exemplos).

Dada a impraticabilidade na resolução de instâncias de grande dimensão para o problema *batch\_milp\_fmsz*, procede-se ao desenvolvimento de procedimentos heurísticos, com a finalidade de obter soluções admissíveis de boa qualidade para as instâncias usuais, relatando-se a análise probabilística efectuada quanto aos desvios resultantes (na função objectivo e na solução binária, reportando-se os tempos de resolução).

A heurística desenvolvida integra sucessivamente diferentes procedimentos de aproximação, concretamente: procedimento prévio, para garantir a existência de solução; procedimento para estimar uma solução inicial, estimando os números e volumes mínimos necessários para os equipamentos a implementar; procedimento para construção de solução admissível, quer versando uma aproximação através da heurística para *single machine* (anteriormente desenvolvida), quer através de uma procura local iterada; procedimento para melhoria de solução admissível, também através de procura local.

O procedimento prévio implementado pretende garantir a existência de solução admissível, isto é, supõe-se que devem ser satisfeitas as restrições temporais com os equipamentos de máxima dimensão,  $dv(j, NS)$ , e em número máximo,  $NP(j)$ , em cada estágio  $j$ . Se tal não se verificar, o problema considerado no enquadramento

de processos múltiplos deve ser generalizado, eventualmente, aumentando-se os valores máximos considerados  $NP(j)$  e/ou  $NS(j)$ .

O procedimento para estimativa inicial (APROX0\_fmsz) é similar ao desenvolvido para o problema de processo único, pois também se supõe que a soma dos tempos de campanha  $t_i$  deverá satisfazer o horizonte disponível  $H$  em todos os estágios. Por conseguinte, estima-se o número de processos e os volumes mínimos necessários em cada estágio  $j$ , sendo estes valores contínuos adequados aos valores discretos superiores (especificados nos dados do problema), considerando-se valor unitário apenas para as variáveis binárias respeitantes. Esta solução é então verificada num procedimento específico (VERIFICA\_SOLUÇÃO\_fmsz) para garantir a sua admissibilidade. Este procedimento produz uma estimativa inicial em cada instância testada, verificando-se a ocorrência de inadmissibilidade (*infeasibility*) na generalidade dos exemplos numéricos (EX2A a EX6A), exceptuando o primeiro exemplo (EX1A) onde produz cerca de 30% de soluções binárias admissíveis e óptimas. De facto, esta aproximação permite estimar uma solução binária que associa uma *lower bound* para os custos de investimento e, caso admissível (*feasible*), essa solução binária constitui a solução óptima.

Para a construção de soluções admissíveis, efectuaram-se duas abordagens diferentes: inicialmente e caso seja admissível, o procedimento APROX1A\_fmsz estima uma solução utilizando o enquadramento de processo único por estágio (*single machine*); caso não seja admissível, aplica-se APROX1B\_fmsz que altera a solução inadmissível, no intuito de aumentar a dimensão e/ou número dos processos.

O procedimento APROX1A\_fmsz, no âmbito de estágios com um só processo, é composto por subprocedimentos (verificação de admissibilidade; se admissível, aplicar aproximação inicial e construção de solução admissível) já desenvolvidos para o modelo batch\_milp\_fssz. A qualidade das soluções estimadas através deste procedimento foi avaliada anteriormente, tendo-se obtido numerosas soluções binárias exactas, ou existindo desvios pouco significativos nas soluções subóptimas (conforme *Tabela 4.4*). Nas instâncias em que tal procedimento produzia soluções admissíveis, confirmou-se o grau de qualidade obtido para *single machine*, mas com uma ligeira deterioração dos estimadores da aproximação: redução em cerca de 10% do número de soluções óptimas obtidas; este aspecto contribui para aumentar o desvio percentual médio para cerca de 3%. Todavia, verificou-se um número significativo de instâncias inadmissíveis, no enquadramento de processo único.

O procedimento APROX1B\_fmsz constrói uma solução admissível a partir de solução inadmissível, prevendo o aumento da dimensão/número do equipamento nos diversos estágios, sendo esse aumento seleccionado de acordo com a melhor razão obtida entre as variações, nos tempos e nos custos, associadas ao aumento referido. A estimativa é verificada, no procedimento VERIFICA\_SOLUCAO\_fmsz, iterando-se o procedimento (de aumento da dimensão/número) até se obter uma solução admissível, constituindo-se este como um procedimento de procura local iterada (*iterated local search*): define-se a vizinhança de procura; estima-se a função de avaliação; procede-se à alteração (aumento) seleccionada; o critério de terminação engloba os objectivos de se atingir uma solução admissível, ou o número limite de iterações. Constatou-se que todas as soluções propostas utilizando este procedimento são admissíveis, embora os desvios percentuais apresentados sejam relativamente superiores aos decorrentes do procedimento anterior. Por conseguinte, torna-se

necessário proceder a uma melhoria da solução admissível aqui construída, cujo procedimento se apresenta em seguida.

No procedimento APROX2\_fmsz pretende-se melhorar a solução admissível anterior, analisando a eventualidade de diminuição da dimensão (ou número) dos processos em cada estágio, seleccionando-se os estágios que não fossem “críticos” (Apêndice 4-A): ou seja, foram seleccionados os estágios onde a diminuição da dimensão (ou número) dos processos não acarretasse aumento dos tempos de campanha, para os diversos produtos. Este procedimento também se constitui um procedimento de busca local iterada, pois define como: vizinhança, os estágios que não fossem “críticos” para a definição dos tempos de campanha dos produtos; função de avaliação, a razão entre a variação nos custos e nos tempos eventuais; alteração, a diminuição da dimensão (número) no estágio seleccionado através da melhor razão de variações; critério de paragem, o esgotamento de alternativas (diminuições admissíveis) ou de número de iterações. Os resultados parciais deste procedimento afirmam a sua utilidade, diminuindo os desvios percentuais das soluções subóptimas para uma margem satisfatória (cerca de 1%).

Apresentam-se na *Tabela 4.6* os estimadores referentes aos desvios nos valores da função objectivo e na solução binária, obtidos dos diversos procedimentos conjugados, respectivamente:

**Tabela 4.6** Desvios e tempos de execução para heurística no âmbito do problema *batch\_milp\_fmsz*.

Exemplo numérico	Tempo <i>solver</i> (s)	Tempo heurística (s)	Desvios função objectivo	Desvios solução binária
EX1A	0.30	0.0027	84./ 0./ 7.6/ 4.9	84./ 2.19
EX2A	10.53	0.0038	68./ 0./ 3.4/ 3.8	68./ 2.03
EX3A	101.67	0.0042	42./ 0./ 7.0/ 6.4	42./ 2.48
EX4A	405.99	0.0072	48./ 0./ 1.7/ 2.1	48./ 1.88
EX5A	(4098.31)	0.0166	37./ 0./ 1.5/ 1.5	37./ 2.58
EX6A	(15142.36)	0.0250	53./ 0./ 0.9/ 0.9	53./ 2.00

Assim, verificando-se a optimalidade de uma fatia importante das instâncias aproximadas, bem sendo considerado globalmente satisfatório o nível de aproximação obtido as soluções subóptimas, não se entende oportuna a continuação da melhoria do procedimento heurístico. De facto, embora nas instâncias de menor dimensão surjam desvios médios na função objectivo superiores a 2%, tal resulta do mesmo tipo de desvio na solução binária: um desvio duplo simultâneo, configurando por excesso a dimensão num estágio, por defeito num outro estágio. Em instâncias de maior dimensão, tal desvio duplo resulta num valor comparativo menor (da ordem de 1%), mas cujo tratamento envolverá uma análise combinatória de grande exigência. Além disso, as soluções obtidas pela conjugação dos procedimentos de aproximação exigem um tempo de resolução praticamente desprezável em comparação com a resolução exacta, constituindo-se como excelentes estimativas iniciais para a resolução enumerativa exacta do método do *branch-and-bound*.

Numa perspectiva de síntese quanto ao modelo *batch\_milp\_fmsz* em análise, o qual considera campanhas de produto único em estágios com várias unidades de produção, saliente-se que: generaliza o modelo com processo único, permitindo

apresentar aqui soluções que eram inadmissíveis face ao modelo anterior (conforme os valores nulos obtidos quanto à fracção de instâncias inadmissíveis); em variadas instâncias, as soluções agora apresentadas são melhores (menores custos de investimento) do que as do modelo a processo único, reduzindo-se naquelas sempre que o número de processos em cada estágio seja unitário; tal como em *batch\_milp\_fssz*, mantém-se a desconsideração relativamente, quer aos tempos de limpeza, quer quanto à integralidade do número de ciclos de produção, cuja gravidade é relativa.

A presente formulação pode ser melhorada, incorporando o tratamento de campanhas de produtos múltiplos (MPC), que generaliza a situação anteriormente tratada de política de campanha de produto único (SPC), sendo estudado em seguida tal aspecto.

#### 4.2.3 O modelo com processo único e em campanha de produtos múltiplos

A adopção de campanhas de produção com produtos múltiplos (*multiple product campaign*, MPC) envolve o sequenciamento dos lotes individuais dos diferentes produtos, no intuito de reduzir tempos “mortos” ( $SL_{ikj}$ , *slacks* ou *idle times*), folgas ou não utilização do equipamento, o que poderá ser vantajoso caso os tempos envolvidos nas transições (por exemplo, limpezas ou *cleanup*) entre diferentes produtos sejam diminutos.

Relativamente aos modelos anteriores, os dados considerados são equivalentes, aos quais haverá a adicionar valores para as referidas folgas  $SL_{ikj}$ , ocorrendo na transição de processamento do produto  $i$  para o produto  $k$ , no estágio  $j$ . Esses valores são calculados através de um algoritmo recorrente, considerando-se a existência de tempos de limpeza  $CL_{ikj}$  (dados a fornecer) associados a cada transição.

No entanto, a dificuldade do cálculo de optimização será bastante superior quando se considera a restritiva política de transferência de espera nula (ZW), especialmente, se os tempos de transição não forem negligenciáveis. O fundamento para tal aumento de dificuldade reside na determinação do tempo de ciclo, pois para uma política de armazenagem intermédia não-limitada (*unlimited intermediate storage*, UIS) com tempos de limpeza nulos ( $CL_{ikj} = 0$ .) esse valor é calculável analiticamente, mas noutras situações (UIS com tempos de limpeza não nulos, ZW) requer também a determinação da sequência de produção no ciclo (*cyclic scheduling*).

Sendo conhecido o número  $n_i$  de lotes a produzir para cada produto, permite-se a determinação do tempo de ciclo, bem como o decorrente sequenciamento da produção através de LP [Birewar e Grossmann, 1989a]. Não obstante, o número  $n_i$  de lotes de cada produto  $i$  é neste modelo uma variável a optimizar, atendendo às procuras e parâmetros técnicos respeitantes, pelo que se encontra amplificado o problema parcial de determinação do tempo de ciclo, o qual era então numericamente tratado de uma forma atraente (como referido, por LP). Ainda, a integralidade do número  $n_i$  de lotes é condição necessária para a integralidade do número de permutas ou transições de produção,  $Nch_{ik}$ , pelo que a não ocorrência da primeira situação impossibilitará a existência desta segunda situação, obviamente desejável para o carácter real da solução apresentada. De facto, requer-se ainda a análise dos valores assim obtidos,  $Nch_{ik}$ , para obtenção do sequenciamento das campanhas e sua posterior composição, pelo que se torna essencial que estes sejam valores inteiros. Um aspecto também a ter em conta, é a possibilidade de ocorrência de subciclos neste enquadramento, pelo que terá de se recalculer o modelo melhorado com a introdução

das restrições de eliminação de subciclos (*subtour elimination constraints*, no âmbito do TSP agregado).

Repare-se que o problema em campanha de produto único, *batch\_milp\_fssz*, pode constituir-se como um caso específico do problema de campanha de produtos múltiplos, *batch\_milp\_fsmz*, supondo neste último que o número de transições para cada produto  $i$  se restringe a (restrição já descrita em b.37):

$$Nch_{ii} = n_i - 1, \quad \forall i$$

Consequentemente, a solução óptima para a modelação em campanha de produto único, SPC, constituirá uma *upper bound* para a situação generalizada de campanha de produtos múltiplos, MPC, dado que esta segunda situação apresenta um espaço de soluções admissíveis bem mais alargado, pois generaliza a primeira.

Em Apêndice 4-A comprova-se a equivalência das variáveis, função objectivo e das restrições entre os modelos considerando campanhas de produto único (*batch\_milp\_fssz*) ou de produtos múltiplos (*batch\_milp\_fsmz*), associando-se então a complexidade computacional dos dois modelos. Testam-se numericamente múltiplas instâncias dos dois modelos, onde se verifica a coincidência de resultados, comprovando-se tanto a equivalência obtida analiticamente, como a implementação computacional aqui efectuada.

Definiu-se uma nova série de exemplos numéricos (série B), condicionada pela geração aleatória sequencial dos dados amostrais, a qual não foi possível aplicar à diversidade dos modelos estudados, pois estes apresentavam conjuntos de dados com dimensões diversas. Nomeadamente, embora os modelos considerando MPC utilizem os mesmos dados que os modelos em SPC, têm de ser complementados com os tempos de limpeza, as folgas nas transições de produção, o que não se coaduna com a repetição da geração aleatória, necessária para tornar comparável os diferentes modelos perante uma mesma instância.

Descrevem-se na *Tabela 4.7* os números de variáveis e restrições associados a cada exemplo numérico testado, reportando-se também o tempo de resolução médio. Dado que os referidos números, definidores da dimensão dos diferentes exemplos, são relativamente modestos, em especial se os compararmos com os do modelo anterior (*batch\_milp\_fsmz*), os tempos necessários também são limitados. Um factor que torna mais eficiente a resolução deste modelo é a já descrita formulação das transições de produção como um TSP agregado, resolúvel através de LP. Dados os tempos apresentados, não se considerou necessário o desenvolvimento de procedimentos de aproximação ou heurísticas.

**Tabela 4.7** Variáveis e restrições em exemplos numéricos (série B) para o problema *batch\_milp\_fsmz*.

Exemplos numéricos	Parâmetros ( $N, M, NS$ )	Variáveis binárias	Variáveis contínuas	Restrições	Tempo <i>solver</i> (s)
EX1B	3, 4, 5	20	12	29	0.46
EX2B	6, 4, 10	40	42	50	1.42
EX3B	8, 5, 12	60	72	74	2.38
EX4B	12, 6, 15	90	156	120	2.89
EX5B	20, 7, 17	119	420	214	*
EX6B	30, 9, 20	180	930	378	*

Todavia, deve referir-se a não exigência de integralidade do número de *batches*,  $n_i$ , a qual poderia aumentar significativamente a dificuldade de resolução do presente modelo. Concretamente, utilizando GAMS/OSL não é possível otimizar directamente problemas com variáveis inteiras de valor superior a 100.

Outro aspecto limitativo por inerência, decorre da consideração de processo único em cada estágio, e como as dimensões discretas se encontram por inerência também limitadas, observa-se a inadmissibilidade de soluções do modelo quando o número de produtos,  $N$ , ou as quantidades procuradas,  $Q$ , se tornam mais exigentes em termos de capacidades (instâncias de EX5B e EX6B, assinaladas com \*).

Apresentam-se na *Tabela 4.8* os estimadores (formulados em Apêndice 4-B) referentes aos desvios nos valores da função objectivo e na solução binária, resultantes da comparação entre os modelos configurando campanhas de produtos múltiplos (MPC) e campanhas de produto único (SPC), respectivamente:

- na função objectivo — percentagem do número de melhorias (desvios negativos nos custos), no âmbito das instâncias admissíveis/ percentagem de instâncias inadmissíveis/ desvio percentual médio para as melhorias/ desvio padrão para os desvios percentuais médios;
- na solução binária — percentagem de instâncias com soluções coincidentes/ média do número de desvios na solução binária, para as instâncias admissíveis comparáveis.

**Tabela 4.8** Desvios resultantes da modelação em *batch\_milp\_fmsz* (MPC) relativamente a *batch\_milp\_fssz* (SPC).

Exemplo numérico	Parâmetros (N, M, NS)	Melhorias função objectivo	Solução binária
EX1B	3, 4, 5	43./ 6.7/ 4.9/ 2.7	57./ 1.6
EX2B	6, 4, 10	97./ 0./ 5.7/ 3.6	3./ 2.5
EX3B	8, 5, 12	100./ 13.3/ 5.2/ 3.1	0./ 3.1
EX4B	12, 6, 15	100./ 66.7/ 5.8/ 0.8	0./ 4.4
EX5B	20, 7, 17	* / 100. / * / *	*
EX6B	30, 9, 20	* / 100. / * / *	*

Da análise da *Tabela 4.8*, evidencia-se que o aumento da necessidade de capacidades, especialmente decorrente do aumento do número de produtos ( $N$ ) considerados em cada exemplo, resulta em: **i)** melhoria das soluções deste modelo relativamente ao *batch\_milp\_fssz* (de SPC), na quase totalidade das instâncias, rondando essa melhoria um valor estável de cerca de 5% do valor-objectivo (custos de investimento); **ii)** tal melhoria decorre de uma maior eficiência de produção, através de optimização das transições efectuadas, permitindo obter o mesmo nível produtivo com dimensões menores, numa parte significativa dos estágios (para EX4B, em cerca de 4 estágios dos 6 estágios considerados); **iii)** o contínuo aumento da necessidade de capacidades, decorrente de mais de duas dezenas de produtos, conduz a situações de inadmissibilidade, conforme se ilustra dos exemplos EX5B e EX6B.

Em consequência, para situações realistas em que se pretenda considerar um número de produtos mais elevado, ou em quantidades  $Q$  mais elevadas, este

enquadramento de um só processo por estágio revela-se limitativo, embora a hipótese de campanhas de produtos múltiplos (MPC) permita obter algumas melhorias relativamente à de campanhas de produto único (SPC).

Complementou-se este estudo, repetindo os exemplos EX5B e EX6B com um número sucessivamente menor de produtos,  $N$ , verificando-se que ocorria admissibilidade caso este número se encontrasse na gama de 15 a 20, considerando-se processo único por estágio e diminuindo a procura por produto,  $Q_i$ , em cerca de 25% do valor médio.

No respeitante ao modelo *batch\_milp\_fsmz* aqui analisado, modelando campanhas com vários produtos mas apenas um processo por estágio, repare-se que: generaliza o modelo de campanha de produto único, o que lhe permite apresentar soluções de melhor qualidade (menores custos de investimento) relativamente a esse modelo, bem como até tornar admissíveis algumas instâncias que eram inadmissíveis no âmbito do modelo de produto único; as soluções apresentadas não contemplam a integralidade do número de *batches*,  $n_i$ , logo também não se verifica a correspondente integralidade do número de permutas,  $Nch_{ik}$ , pelo que a formação das respeitantes campanhas não se encontra assegurada; esta formulação pode também ser generalizada, alargando-se o número de processos considerados em cada estágio,  $NP$ , a valores superiores à unidade, configurando-se assim o modelo seguinte, *batch\_milp\_fmmz*.

#### 4.2.4 O modelo com processos múltiplos e em campanha de produtos múltiplos

Este modelo considera processos múltiplos em cada estágio, bem como campanhas de produtos múltiplos, sendo o modelo mais generalizado desta série aqui em estudo. Por conseguinte, tendo os anteriores modelos (SPC com processo único por estágio; SPC com processos múltiplos; MPC com processo único) considerado situações específicas aqui englobadas, as suas soluções nunca apresentarão qualidade superior, mas podem constituir valores de referência (iniciais, de corte) para a resolução desta formulação mais complexa.

As variáveis envolvidas são as usuais para os modelos analisados, bem como os dados e parâmetros são correspondentes aos já descritos. Dada a generalização aqui formulada, em termos de multiplicidade de processos por estágio e de produtos por campanha, esperam-se soluções de qualidade superior.

Pretendendo comparar os modelos *batch\_milp\_fsmz* (*Flowshop/ Single machine/ MPC/ ZW*) e *batch\_milp\_fmmz* (idem, mas aqui considera-se *multiple machine*), restringem-se o número de processos por estágio a  $NP = 1$  e consideram-se folgas nulas,  $SL_{ikj} = 0.0$ , conformando assim os pressupostos destes dois modelos; efectuam-se duas séries de 30 amostras, observando-se que os resultados de ambos os modelos são iguais. Assim, é possível garantir a coerência numérica da implementação computacional destes dois modelos. No entanto, a suposição de folgas nulas desvirtua o enquadramento em que tal hipótese é colocada, pois a adopção de campanhas de produtos múltiplos visa otimizar, justamente, as folgas que ocorrem entre cada transição de *batches*. Tal contradição nos pressupostos, poderá originar resultados anómalos, como se irá constatar adiante.

O problema considerando campanhas de produtos múltiplos (MPC) em processo único, *batch\_milp\_fsmz*, constitui um caso específico do problema de processos múltiplos, *batch\_milp\_fmmz*, atendendo a que neste último o número de processos em cada estágio se pode restringir a  $NP = 1$ . Consequentemente, generalizando o número de processos em cada estágio, alarga-se o espaço de soluções

admissíveis, pelo que se poderão obter melhores resultados. Assim, a solução óptima do modelo de um só processo representa uma *upper bound* para o modelo de processos múltiplos, podendo constituir um valor de corte ou uma solução inicial para este último modelo.

Em Apêndice 4-A mostra-se a equivalência das variáveis, função objectivo e restrições entre estes dois modelos de campanha de produtos múltiplos, bem como as condicionantes a verificar para que tal ocorra, pelo que a classe de complexidade computacional do problema em *single machine*, implica a mesma classe para o problema em *multiple machine*.

Ainda, repare-se que o problema em SPC, *batch\_milp\_fmsz*, constitui um caso específico do problema em MPC, *batch\_milp\_fmmz*, ambos considerando processos múltiplos em cada estágio, supondo neste último que o número de transições  $Nch_{ikj}$ , para cada produto  $i$ , satisfaz a condição descrita em b.37. Poder-se-ia, caso necessário, mostrar também a equivalência entre estas duas formulações configurando processos múltiplos.

Na *Tabela 4.9* são apresentados os números de variáveis, binárias e contínuas, bem como o número de restrições envolvidas nos exemplos testados (série B), além do tempo médio de optimização.

**Tabela 4.9** Variáveis e restrições em exemplos numéricos (série B) para o problema *batch\_milp\_fmmz*.

Exemplos numéricos	Parâmetros (N, M, NS, NP)	Variáveis binárias	Variáveis contínuas	Restrições	Tempo solver (s)
EX1B	3, 4, 5, 3	60	12	29	1.65
EX2B	6, 4, 10, 3	120	42	50	3.49

Ressalta a descrição de tempos de resolução apenas para os primeiros dois exemplos desta série B, o que se justifica devido à incongruência verificada nos resultados obtidos. De facto, para os exemplos testados, verificaram-se melhorias de relevo face às soluções apresentadas considerando MPC e *single machine*: as instâncias então inadmissíveis são, agora em *multiple machine*, já admissíveis; para as instâncias então admissíveis, observa-se agora menores custos. Contudo, uma análise aos valores óptimos revela que, para as instâncias comparáveis (com um só processo por estágio), os custos menores decorrentes de configurações de menor dimensão só são assim obtidos tendo em conta que neste modelo *batch\_milp\_fmmz* não se consideram as folgas associadas às transições. Isto é, perante a mesma instância, com iguais parâmetros e dados, este modelo apresenta igualmente configurações de um só processo por estágio, mas menores dimensões (e custos), pois não trata as folgas, já que considera simultaneamente as políticas de espera nula (*zero wait*) e de armazenagem intermédia ilimitada (*unlimited intermediate storage*, UIS).

Esta contradição deriva de uma limitação de índole teórica: para sequenciar campanhas de produtos múltiplos considerando *single machine*, utiliza-se a abordagem de Pekny e Miller [1991] que trata esta situação através de LP, considerando um TSP (*Traveling Salesman Problem*) agregado, descrito na Sec. 2.2. Em termos de *multiple machine*, seria necessário enquadrar um problema de TSP para cada um dos processos em paralelo, o que manifestamente não aparenta ser praticável.

Relativamente ao modelo em estudo *batch\_milp\_fmmz*, que formula campanhas com produtos múltiplos e processos múltiplos em cada estágio, note-se que: generaliza o modelo anterior de processo único (com campanha de produtos múltiplos, MPC), apresentando melhores soluções (menores custos de investimento); torna admissíveis instâncias que o não eram no âmbito do modelo de processo único, devido ao inerente limite para as capacidades de produção; são numerosas as instâncias onde não se verifica a integralidade do número de *batches*,  $n_i$ , não se observando também a integralidade respeitante ao número de permutas,  $N_{ch_{ik}}$ ; não contempla os custos associados à armazenagem intermédia, sejam custos de investimento ou de operação, pressuposto necessário para a coerência real do presente modelo.

Conclui-se que, embora o modelo *batch\_milp\_fmmz* possa apresentar (incoerentemente) soluções de menores custo, a respeitante aplicação prática encontra-se comprometida dada a contradição nos pressupostos em que este modelo foi desenvolvido.

## Sinopse

Na Sec. 4.1 deste Capítulo dedicado à análise dos modelos descritos no Capítulo 2 e das heurísticas aqui desenvolvidas, procede-se a um estudo teórico dos modelos de interesse para o planeamento de processos.

Resumidamente, comprovou-se a dificuldade de resolução dos modelos de planeamento de processos abordados, na medida em que se procedeu à classificação da complexidade computacional. O trabalho aqui desenvolvido inicia-se, como referido oportunamente, com um estudo de complexidade similar ao realizado em Ahmed *et al.* [2000]: tal necessidade decorre de o modelo matemático ali considerado, embora similar na sua globalidade ao aqui analisado, diferir em alguns grupos de restrições. A restante abordagem teórica desenrola-se de forma autónoma, no sentido da conduzida por Furman e Sahinidis [2001] dirigida a modelos de redes de permutadores de calor (HEN).

De uma forma geral, verificou-se que estes modelos de planeamento de processos constituem problemas NP-hard, como expectável, já que se trata de modelos MILP, embora tal formulação não seja condição suficiente, pois existem diversos modelos com restrições de integralidade e que são polinomialmente resolúveis (por LP, através de Programação Dinâmica, ou reformulações, como se observa nos problemas ELS ou TSP).

Do exposto, ressalta a necessidade de se desenvolverem heurísticos, caso se pretendam tratar instâncias de grande dimensão (mas sendo praticável a resolução de problemas de pequena dimensão com as técnicas usuais), dado o progresso computacional que se tem verificado, quer em termos de equipamentos, quer em termos de implementações e métodos.

Esta abordagem teórica permitiu um conhecimento detalhado destes modelos de planeamento de processos, das suas vantagens e limitações. Assim, torna-se necessário realizar uma abordagem dirigida à optimalidade, para garante da robustez da configuração do sistema de processos, face à incerteza inerente ao planeamento de médio e longo prazo.

Na Sec. 4.2, estudam-se e comparam-se modelos dirigidos ao dimensionamento e sequenciamento de processos descontínuos, seleccionados e descritos no Capítulo 2.

Tais modelos consideram política de espera nula (ZW), mas variam entre si quanto ao modo de campanhas de produção (SPC *vs.* MPC) e ao número de processos por estágio (processo único *vs.* processos múltiplos). Constituem assim quatro modelos, sendo possível mostrar relações de equivalência perante instâncias específicas (Apêndice 4-A) desses modelos, o que implica idêntica classe de complexidade computacional. Para os modelos que o justificavam, foram aqui desenvolvidas heurísticas com resultados satisfatórios (frequentemente óptimos), segundo o estudo probabilístico associado.

Em seguida, sintetizam-se para cada modelo o trabalho desenvolvido e as limitações encontradas:

- modelo *batch\_milp\_fssz* — considerando SPC e processo único, constitui o modelo mais limitado, mas pode utilizar-se como base para estimativa inicial; desenvolveram-se diversas heurísticas, realizando-se a sua análise probabilística a desvios perante instâncias médias, cujos resultados globais se revelam de boa qualidade (fracção significativa de soluções óptimas, desvios de cerca de 1% para as soluções subóptimas, desvio combinatório mínimo na solução binária);
- modelo *batch\_milp\_fmsz* — supondo SPC mas em processos múltiplos, apresenta-se como um modelo de difícil resolução, devido ao elevado número de variáveis contínuas e de restrições; também se desenvolveram heurísticas, com a sua análise probabilística, obtendo-se resultados de boa qualidade, similar à do modelo anterior;
- modelo *batch\_milp\_fsmz* — para MPC e processo único, revela-se de resolução relativamente eficiente, atendendo à agregação do TSP que lhe permite o tratamento por LP, e à não exigência de integralidade do número de *batches*; relativamente ao modelo *batch\_milp\_fssz* (directamente comparável mas versando SPC), as instâncias testadas permitem prever uma diminuição dos custos de investimento de cerca de 5%; no entanto, a limitação de processo único coloca entraves à obtenção de soluções admissíveis em face de instâncias realistas (aumento do número de produtos ou das quantidades procuradas);
- modelo *batch\_milp\_fmmz* — perante MPC e processos múltiplos, apresenta o maior potencial de aplicação dos modelos aqui analisados; todavia, contradições nos pressupostos conduzem a resultados numéricos incoerentes, quando comparados com os resultados de outros modelos para as mesmas instâncias.





# Estudo numérico da optimalidade

Neste Capítulo, elaboram-se estudos da optimalidade dos casos equacionados (planeamento de processos, dimensionamento e sequenciamento de processos descontínuos), adequando-se os modelos aos parâmetros incertos e promovendo-se a robustez, a qual é quantificável através dos devidos estimadores. A robustez dos modelos generalizados e das soluções obtidas em ambiente de incerteza é valorizada e, não obstante a consideração de amostras com pequena dimensão, os resultados revelam a valia da abordagem proposta.

Os modelos de optimização considerados para aqueles casos apresentam duas componentes: uma componente estrutural (em primeira fase), independente da variação nos seus dados, onde se observam as variáveis de projecto que são independentes de qualquer instância dos parâmetros incertos; e uma componente de recurso ou controlo (em segunda fase), que é susceptível de perturbação nos dados, onde se encontram as variáveis de controlo que são ajustadas assim que os parâmetros incertos são conhecidos, bem como após se definirem os valores das variáveis de projecto: este enquadramento é o de Programação Estocástica em Duas fases (*Two-Stage Stochastic Programming*, 2SSP).

O planeamento de expansão de capacidades é significativo para muitas indústrias e caracteriza-se pela afectação de recursos substanciais, em investimentos com longos períodos de retorno, observando-se economias de escala devido à tecnologia utilizada (por exemplo, nas indústrias da química pesada, da metalurgia, ou outras). Então, a incerteza na procura poderá causar dois tipos de desvios: por excesso de capacidade, que representa recursos de capital que não são utilizados eficientemente, podendo afectar uma expansão futura; e por procura não-satisfeita, com as consequências da imposição de penúria. Estes dois tipos de desvios são utilizados para penalização do valor objectivo, no intuito de promover a robustez.

Um modelo robusto permite obter planos de expansão de capacidades caracterizados por:

- robustez da solução, quando a solução óptima (ou valor esperado) obtida permanece “próxima” da solução óptima para qualquer instância dos cenários de procura; interessa então minimizar a variabilidade (por exemplo, a variância ou desvio absoluto) do valor objectivo de segunda fase, penalizando-a; desta forma, a solução óptima do modelo constitui também uma solução de boa qualidade para qualquer um dos cenários considerados;
- robustez do modelo, pois a solução óptima apresenta uma redução do excesso de capacidade, ou da procura não-satisfeita, considerando todos os cenários; penalizam-se então os estimadores associados; por conseguinte, num problema de planeamento de produção (“produção de capacidades”) com incerteza nas previsões da procura, o modelo robusto não deve apresentar nem valores elevados para a procura não-satisfeita, nem para o excesso de produção.

Variando os parâmetros de penalização, permite-se que o decisor analise o balanço da robustez do modelo e da solução *versus* o custo dessa robustez. Esses parâmetros podem ser vistos como reflectindo a atitude face ao risco do decisor, podendo mesmo apresentar comportamentos conflitantes: porventura, aumentando-se as capacidades para satisfazer uma maior procura previsível, também se aumentará o excesso de capacidades.

Os modelos de longo prazo, utilizando dados incompletos ou incertos, são comuns neste tipo de problemas, sendo tratados de forma reactiva, através de análise de sensibilidade, ou de forma pró-activa, através de formulações de SP, enquadrando este que permite promover a robustez.

A usual análise de sensibilidade consiste num estudo de optimalidade reactiva, pois apenas se debruça sobre o impacto da incerteza dos dados nas variáveis de decisão. Frequentemente, assumem-se análises ao valor médio (por ex., o *expected value problem*, EVP) ou ao pior caso possível (*worst case*): a abordagem de valor médio estabelece limites com elevado erro, enquanto a abordagem ao pior caso possível produz formulações conservadoras e soluções onerosas. Este tipo de abordagem reactiva não permite controlar a incerteza, pelo que é necessário adequar os procedimentos.

A abordagem por SP é pró-activa, pois o decisor pode basear-se na flexibilidade das variáveis de recurso ou controlo, em segunda fase, onde as recomendações do modelo podem ser ajustadas às instâncias dos dados. No entanto, otimiza-se apenas o valor esperado do valor objectivo, ignorando-se a respectiva variabilidade e desprezando também as preferências do decisor perante o risco.

A distinção entre SP e a optimização robusta (*robust optimization*) apresenta-se no aspecto da função objectivo, onde a optimização robusta controla a variabilidade da solução, enquanto que em SP apenas se optimiza um valor esperado. Outro aspecto diferenciador é o tratamento das restrições do modelo de optimização: em SP, quando não se encontra uma solução admissível das variáveis de projecto que permita a existência de variáveis de controlo que satisfaçam as restrições para todos os cenários (*complete recourse*), o modelo é declarado inadmissível; em optimização robusta, esta contingência é explicitada, possibilitando-se a existência de termos de penalidades e desvios que permitam encontrar uma solução que viole as restrições com menor gravidade.

As soluções obtidas por um modelo de otimização robusta podem ser comparadas com as de um modelo de SP, sendo estas soluções de SP calculadas utilizando-se valores nulos para os parâmetros de penalização. Exemplificando com um problema de expansão de capacidades, a solução de SP apresentará um menor custo esperado de investimento, dado que não são consideradas quaisquer penalizações, mas resultará em excesso de capacidade para a grande maioria dos cenários previsíveis de procura. De facto, as formulações de SP apresentam menores custos esperados pois aproveitam as economias de escala presentes em unidades de grande dimensão, mas os planos assim obtidos são sensíveis à incerteza na procura e sobrevalorizam os cenários de baixa probabilidade, os quais dirigem a solução.

Todavia, no âmbito da otimização robusta ocorrem algumas limitações, nomeadamente, quanto à especificação dos parâmetros de penalização, assim como não se providenciam meios para seleccionar os cenários, embora estes constituam apenas um conjunto de instâncias admissíveis dos dados do problema (questão que também ocorre em SP).

Um dilema na otimização baseada em cenários consiste no número destes a incluir para encontrar uma solução robusta para todos os eventos possíveis da população que os cenários pretendem estimar. Laguna [1998] realizou experiências numéricas para seleccionar o número de cenários óptimo, sendo a robustez testada em função da população de cenários. Aplicando a solução de expansão óptima obtida através da amostragem, ao longo da população de cenários, verificou trivialmente que quanto maior a dimensão da amostra, menor o desvio relativamente ao resultado da população. Contudo, também constatou que amostras de pequena dimensão já produzem soluções próximas do óptimo. Assim, nos casos de otimização aqui estudados, define-se um conjunto plausível de instâncias como cenários discretos.

Quanto à implementação computacional de problemas de SP e com robustez, caracterizados pela grande dimensão das suas instâncias, note-se que a sua resolução requer usualmente a utilização de computadores de elevado desempenho —não disponíveis—, com a aplicação de procedimentos em paralelo ou distribuídos.

Nas seguintes secções, propõem-se modelos generalizadores das vertentes estocástica e de robustez, bem como de aspectos operacionais de interesse, os quais são dirigidos aos casos de otimização descritos, nomeadamente, para o planeamento de processos, e para o dimensionamento e sequenciamento de processos descontínuos.

Na Sec. 5.1, No âmbito do planeamento de processos, desenvolve-se um modelo de otimização que valoriza a robustez, com variáveis binárias associadas a decisões de investimento de longo prazo, além de considerar as decisões operacionais sobre a rede de processos, no âmbito da 2SSP. O modelo robusto proposto congrega aspectos de produção flexível, considerando diferentes esquemas de produção, característicos da indústria de processos químicos —por exemplo, em termos de matérias-primas e de produtos intermédios a utilizar, conforme os aspectos tratados em Apêndice 5-A—, com o tratamento da incerteza intrínseca a um horizonte temporal dilatado.

Na Sec. 5.2, descreve-se e avalia-se a utilidade de um modelo robusto para o dimensionamento e sequenciamento de processos descontínuos, que permite generalizar: *i*) o enquadramento determinístico e uniperíodo —do sequenciamento multiproducto (*flowshop*), com processos múltiplos em cada estágio (*multiple machine*), campanhas de produto único (SPC) e política de espera nula (*zero wait, ZW*)— a um enquadramento estocástico com robustez, mas ainda uniperíodo;

*ii)* este tratamento uniperíodo (estático) a um tratamento multiperíodo (dinâmico), para optimização do VLA robusto correspondente ao investimento em processos descontínuos com dimensões discretas.

## 5.1 Abordagem generalizada ao planeamento de processos: flexibilidade e robustez

No âmbito do planeamento de processos, desenvolve-se um modelo de optimização multiperíodo de MILP, que, por um lado, englobe a optimização das decisões de investimento de longo prazo e, por outro, considere as decisões operacionais sobre a rede de processos. O dimensionamento e a temporização da expansão de capacidades de unidades de processo são importantes, dado que exigem um substancial investimento associado a um longo tempo de retorno. A natureza determinística dos modelos não acompanha a incerteza inerente ao largo horizonte de planeamento, no qual haverá que estimar procuras de produtos e seus preços, o que é uma primeira dificuldade. Preconiza-se, então, o enquadramento da 2SSP, onde: as variáveis de projecto estão associadas apenas aos constrangimentos estruturais, os quais não dependem dos parâmetros incertos da procura (decisões binárias, capacidades, expansão de capacidades); as variáveis de controlo podem ser ajustadas assim que os parâmetros incertos são conhecidos, bem como são afinadas de acordo com as variáveis de controlo (quantidades compradas e vendidas, fluxos).

Outras dificuldades subsistem, nomeadamente, quanto à flexibilidade nos processos de produção, de forma a considerar a produção de um mesmo produto a partir de diferentes matérias-primas ou componentes. Desta forma, considera-se a possibilidade de um dado processo apresentar configurações alternativas, integrando diferentes esquemas de produção.

De seguida, apresenta-se a nomenclatura aplicável ao modelo estocástico que se desenvolveu:

- $NC, NM, NP, NR$ , números de componentes ou materiais  $j$ , de mercados de
- $NS, NT$  – materiais  $l$ , de processos de produção  $i$ , de cenários previsíveis  $r$ , de esquemas de produção  $s$ , de períodos de tempo  $t$ ;
- $prob$  – probabilidade associada a cada cenário;
- $\xi$  – valor líquido actual (VLA);
- $\lambda_{dsv}, \lambda_{zp}$  – parâmetros de penalização;
- $Z_p$  – erros ou desvios de impossibilidade, em cada cenário;
- $\gamma, \Gamma$  – preços unitários actualizados, de venda e de compra, dos produtos químicos;
- $\delta$  – custos unitários actualizados de produção, relativos ao componente principal de cada processo;
- $\alpha, \beta$  – custos actualizados para expansão de capacidades, de natureza variável e fixa;
- $a, d$  – disponibilidades e procuras dos produtos químicos;
- $\mu$  – constantes características, associando os fluxos de entrada e de saída de cada componente a um fluxo genérico  $W$  do processo;

- $\rho$  – taxa de produção adimensional relativa, definida a partir da taxa de produção e do factor de capacidade  $Q$ ;
- $\theta$  – quantidade (do produto principal  $j$ ) em produção no esquema  $s$  do processo  $i$  e no período  $t$ , para cada cenário  $r$ ;
- $H$  – valor máximo da fracção do tempo de produção disponível, pelo processo  $i$  ao longo do período  $t$ ;
- $PS(i)$  – conjunto dos esquemas de produção  $s$  alocáveis ao processo flexível  $i$ ;
- $QE_{it\tau}^{accum}$  – limite superior (UB) para a expansão de capacidades, respeitante à “procura de capacidade” acumulada desde o período  $t$  até ao período  $\tau$ .

O modelo desenvolvido para optimização considera uma função objectivo estocástica, a qual visa a maximização do valor esperado, condicionado por parcelas de penalização, devidas seja à variabilidade perante os cenários amostrais, seja por subutilização das capacidades de produção implementadas, de forma a traduzir a robustez do modelo:

$$\text{sp.1} \quad [\max] \Phi = \sum_{r=1}^{NR} \text{prob}_r \xi_r - \lambda dsv \sum_{r=1}^{NR} \text{prob}_r \cdot dsvn_r - \lambda zp \sum_{i=1}^{NP} \sum_{t=1}^{NT} \sum_{r=1}^{NR} Zp_{itr}$$

O objectivo determinístico corresponde ao VLA, cuja avaliação decorre em cada cenário  $r$ , definindo-se como o montante das vendas a menos dos montantes das compras de componentes, dos custos de operação dos processos, dos custos de investimento, fixos e variáveis, em expansão de capacidades:

$$\text{sp.2} \quad \xi_r = \sum_{j=1}^{NC} \sum_{t=1}^{NM} \sum_{t=1}^{NT} (\gamma_{jlt} \text{Sal}_{jltr} - \Gamma_{jlt} \text{Pur}_{jltr}) - \sum_{i=1}^{NP} \sum_{s \in PS(i)} \sum_{t=1}^{NT} (\delta_{ist} \rho_{is} \theta_{istr}) - \sum_{i=1}^{NP} \sum_{t=1}^{NT} (\alpha_{it} QE_{it} + \beta_{it} y_{it}), \quad \forall r$$

Um modelo de optimização apresenta uma solução designada como robusta caso o custo actual do cenário especificado permaneça próximo do valor esperado óptimo para todos os cenários. Usualmente, minimiza-se a variância dos custos de segunda fase simultaneamente com a minimização da respectiva esperança dos custos. Aqui introduz-se uma medida de robustez que penaliza os valores de segunda fase que se encontrem abaixo do valor esperado respectivo, medida essa formulada linearmente de maneira possibilitar a aplicação de técnicas de LP. A métrica linear para penalizar a variabilidade, em cada cenário aleatório  $r$ , é dada pelo desvio negativo,  $dsvn$ :

$$\text{sp.3} \quad dsvn_r \geq \sum_{r'=1}^{NR} (\text{prob}_{r'} \xi_{r'}) - \xi_r, \quad \forall r$$

É conveniente que se evite uma significativa subutilização das unidades em expansão, obtendo-se um melhor ajuste das capacidades às necessidades de produção, para o que se introduziu o factor de penalização  $\lambda zp$ . O desvio por defeito ao constrangimento de limite superior para a capacidade de produção, o qual representa a subutilização das unidades de processo implementadas e que se quantifica nas variáveis de folga  $Zp$ , obtém-se de:

$$\text{sp.4} \quad \sum_{s \in PS(i)}^{NS} \theta_{istr} + Zp_{itr} = H_{it} Q_{it}, \quad \forall i, t, r$$

Uma das principais limitações decorre da disponibilidade financeira para investir no sistema de processos em expansão. A ocorrência de um limite superior  $CI(t)$  para o montante disponível para investimento em cada período  $t$ , é modelada através do seguinte grupo de restrições:

$$\text{sp.5} \quad \sum_{i=1}^{NP} (\alpha_{it} QE_{it} + \beta_{it} y_{it}) \leq CI(t), \quad \forall t$$

A restrição de carácter lógico, que caracteriza as variáveis para expansão de capacidades  $QE$ , será apenas incorporada como limite superior, dado que a implementação de unidades de pequena capacidade não representa qualquer economia de escala, sendo mesmo contra-indicada pela modelação efectuada:

$$\text{sp.6} \quad QE_{it} \leq QE_{it}^{Upp} y_{it}, \quad \forall i, t$$

As restrições quanto às variáveis de projecto incorporam os balanços multiperíodo para as capacidades de produção  $Q$ , para cada processo  $i$  no período  $t$ :

$$\text{sp.7} \quad Q_{it-1} + QE_{it} = Q_{it}, \quad \forall i, t$$

Além de não se considerar o correspondente limite inferior lógico para as capacidades de expansão, também se entende que o usual constrangimento ao número de expansões de cada processo apresenta apenas um relativo interesse numérico, não traduzindo qualquer benefício em termos de aproximação à realidade, pelo que não se inclui nesta formulação. De facto, caso tal restrição preveja um número de expansões excessivamente diminuto, porventura eliminaria soluções de interesse económico, enquanto que um valor excessivo para tal limite superior tornaria a restrição redundante.

A consideração dos fluxos desagregados de produção e dos respectivos coeficientes  $\mu$  (com sinal positivo para produção, negativo para consumo), no âmbito do balanço mássico global a cada componente  $j$ , origina a igualdade:

$$\text{sp.8} \quad \sum_{l=1}^{NM} Pur_{jltr} + \sum_{i=1}^{NP} \sum_{s \in PS(i)}^{NS} \mu_{ijs} \rho_{is} \theta_{istr} = \sum_{l=1}^{NM} Sal_{jltr}, \quad \forall j, t, r$$

Atendendo a que a delimitação das quantidades (procuras e disponibilidades) dos diversos materiais interessa sobretudo para definir a gama de variação numérica, aqui apenas se restringirá o valor superior respeitante, no intuito de que não seja eliminada qualquer solução decorrente do espaço probabilístico possível:

$$\text{sp.9} \quad Pur_{jltr} \leq a_{jltr}^{Upp}, \quad \forall j, l, t, r$$

$$\text{sp.10} \quad Sal_{jltr} \leq d_{jltr}^{Upp}, \quad \forall j, l, t, r$$

A consideração das anteriores relações no modelo *spplan\_milp\_flex* congrega aspectos de flexibilidade com os de robustez: considera-se flexibilidade através de diversos esquemas de produção alocáveis aos processos; simultaneamente, trata-se a incerteza associada ao investimento a longo prazo, pois pondera o valor esperado penalizado pela sua variabilidade e pelos desvios à modelação.

**Modelo *spplan\_milp\_flex*:**

$$\text{sp.11-a} \quad [\max] \Phi = \sum_{r=1}^{NR} \text{prob}_r \xi_r - \lambda \text{dsv} \sum_{r=1}^{NR} \text{prob}_r \cdot \text{dsvn}_r - \lambda \text{zp} \sum_{i=1}^{NP} \sum_{t=1}^{NT} \sum_{r=1}^{NR} Zp_{itr}$$

sujeito a

Relações *sp.2—sp.10*

$$\text{sp.11-b} \quad \xi_r, \text{dsvn}_r, \text{Sal}_{jltr}, \text{Pur}_{jltr}, \theta_{isrt}, Zp_{itr}, QE_{it}, Q_{it} \geq 0, \quad \forall i, j, l, s, t, r$$

$$\text{sp.11-c} \quad y_{it} \in \{0,1\}, \quad \forall i, t$$

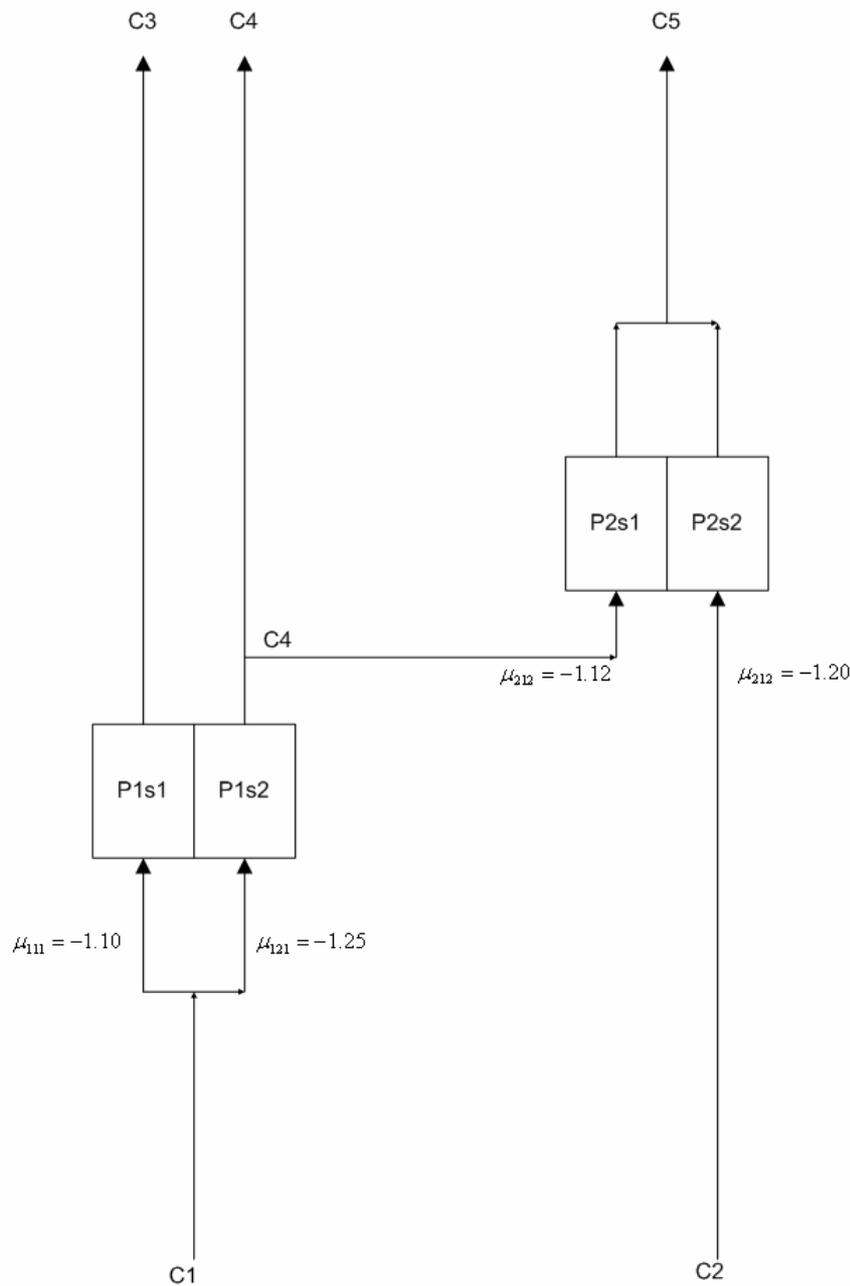
O modelo *spplan\_milp\_flex* apresenta:  $(2.NR+2.NM.NC.NT.NR+NP.NS.NT.NR+NP.NT.NR+2.NP.NT)$  variáveis contínuas, correspondentes, respectivamente, ao VLA e desvios, quantidades vendidas e compradas dos componentes, quantidades produzidas, variáveis de folga de capacidades, expansão de capacidades e capacidades;  $(NP.NT)$  variáveis discretas, respeitantes às variáveis de decisão binárias em cada período de tempo e para cada processo;  $(2.NR+NT+NP.NT.NR+2.NP.NT+NC.NT.NR+2.NC.NM.NT.NR)$  restrições, respectivamente, associadas às restrições de definição do VLA e desvios, limite ao investimento, definição das folgas de capacidades, balanços e restrições lógicas às capacidades, balanço global aos componentes, restrições de limite superior às procuras e disponibilidades.

**5.1.1 Aplicação numérica: sensibilidade a penalizações (*spplan\_ex1*)**

Considere-se a superestrutura da rede de processos representada na *Figura 5.1* e constituída por dois processos contínuos flexíveis, cada um composto por dois esquemas de produção. O processo 1 demonstra flexibilidade no produto, pois a partir do mesmo componente reagente C1, tanto produz o componente C3 como o componente C4. O processo 2 revela flexibilidade nas matérias reagentes, pois origina o componente produto C5, quer a partir de C2 ou de C4. Os materiais reagentes C1 e C2 são comprados e utilizados nos processos químicos, enquanto os produtos C3 e C5 são produzidos e vendidos; o componente C4, produzido no esquema 2 do processo 1, tanto pode ser vendido, como utilizado como reagente no esquema 1 do processo 2.

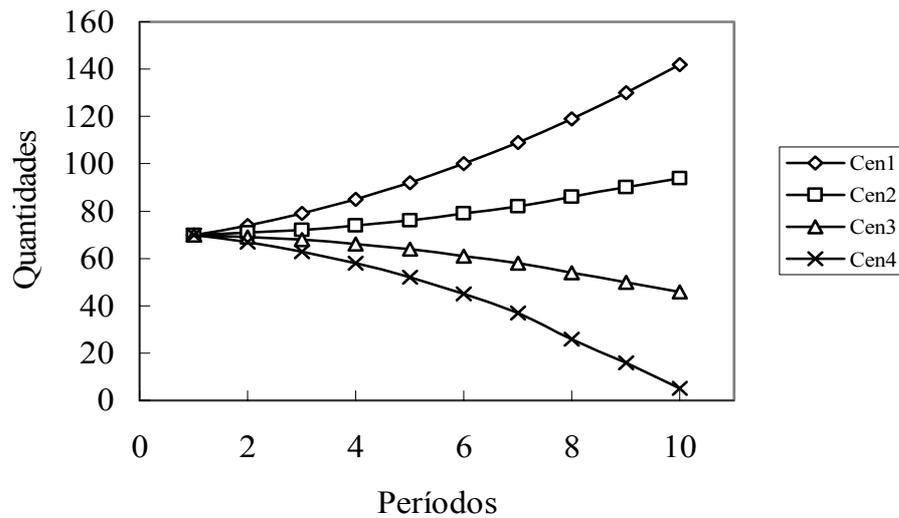
Para o modelo estocástico, supõe-se incerteza na procura do componente C5, a qual é descrita através de um número  $(NR = 4)$  discreto de cenários. Os dados numéricos para o exemplo *spplan\_ex1* encontram-se no Apêndice 5-B, contemplando um horizonte temporal de  $(NT = 10)$  períodos anuais.

Apresentam-se na *Figura 5.2* os 4 cenários (Cen.) da procura do componente C5 ao longo do horizonte de 10 períodos anuais prevendo-se: uma subida acentuada (Cen. 1); uma subida moderada (Cen. 2); uma descida moderada (Cen. 3); uma descida acentuada (Cen. 4). Desta forma, apresentam-se estruturas de procura com características diferentes, as quais são devidamente ponderadas pelas probabilidades de ocorrência (*prob*). Estas probabilidades podem corresponder, em termos relativos, a uma hierarquização dos cenários segundo a presunção da sua ocorrência, o que contribui para um maior cuidado na sua elaboração.



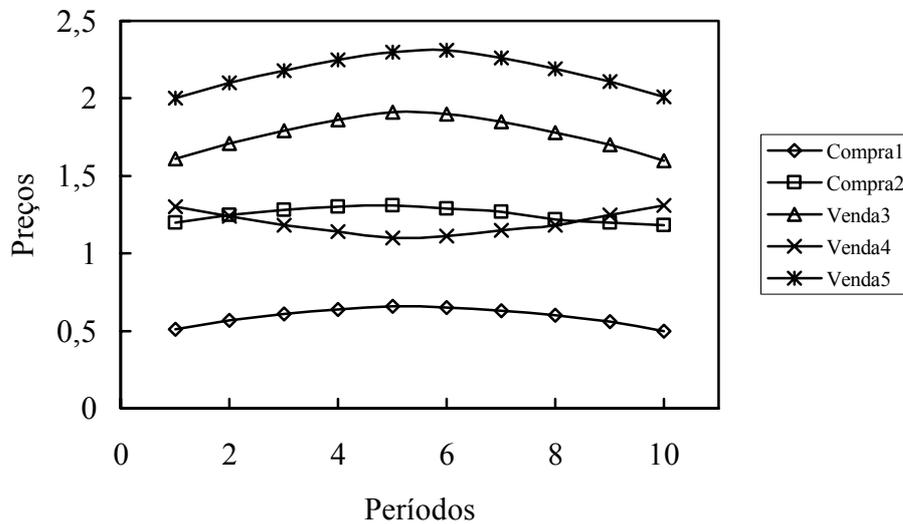
**Figura 5.1** Superestrutura da rede de processos para o exemplo numérico *spplan\_ex1*.

Desta forma, especifica-se um conjunto discreto de cenários plausíveis para este exemplo numérico, cenários que podem ser hierarquizados de acordo com uma postura mais agressiva ou otimista, em contraponto com uma postura mais conservativa ou pessimista. Anotem-se as observações de Laguna [1998] para adequar o número de cenários, que se descrevem novamente: quanto maior a dimensão da amostra, menor o desvio relativamente ao resultado da população inteira; no entanto, amostras de pequena dimensão também produzem soluções próximas do ótimo. Então, prevêem-se resultados de boa qualidade, com um esforço computacional que não exceda os limites do razoável, dado o crescimento exponencial do tempo de resolução para este tipo de problema.



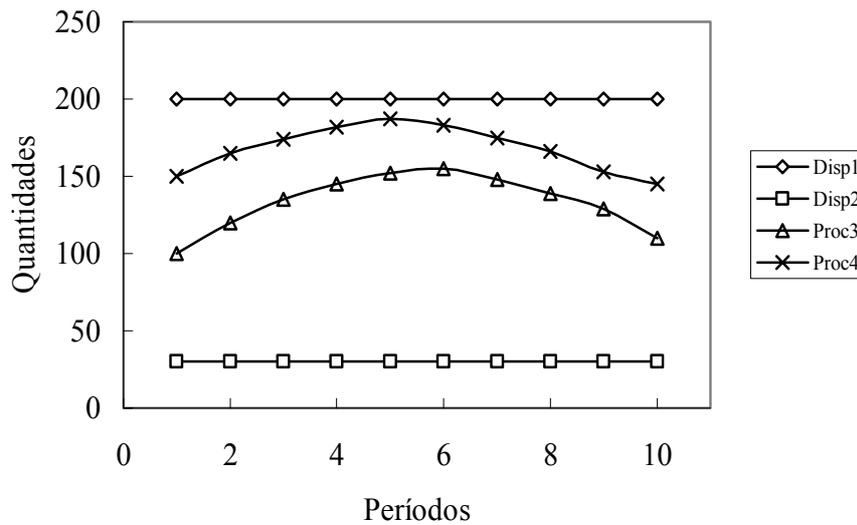
**Figura 5.2** Cenários da procura do componente C5 ao longo do horizonte temporal.

Na *Figura 5.3* observa-se a variação dos preços de compra e venda de componentes ao longo do horizonte temporal, verificando-se os valores mais elevados para os componentes produtos C5 e C3, enquanto os componentes reagentes C1 e C2 são mais acessíveis. O componente C4, com a sua dupla utilização (como reagente e produto), apresenta valores similares aos do reagente C2, mas revela uma variação convexa (com a concavidade voltada para cima). Desta maneira, será competitiva a utilização de quer um, quer outro dos componentes C2 e C4, como reagentes nos esquemas do processo 2.



**Figura 5.3** Variação dos preços de compra e venda de componentes ao longo do horizonte temporal.

As disponibilidades e procuras dos componentes C1 a C4 são apresentadas na *Figura 5.4*, onde se verifica: uma grande disponibilidade do reagente C1; uma disponibilidade limitada do reagente 2 (motivando daí a sua possível substituição com o componente C4); procuras variáveis dos componentes C3 e C4.



**Figura 5.4** Disponibilidades e procuras dos componentes C1 a C4

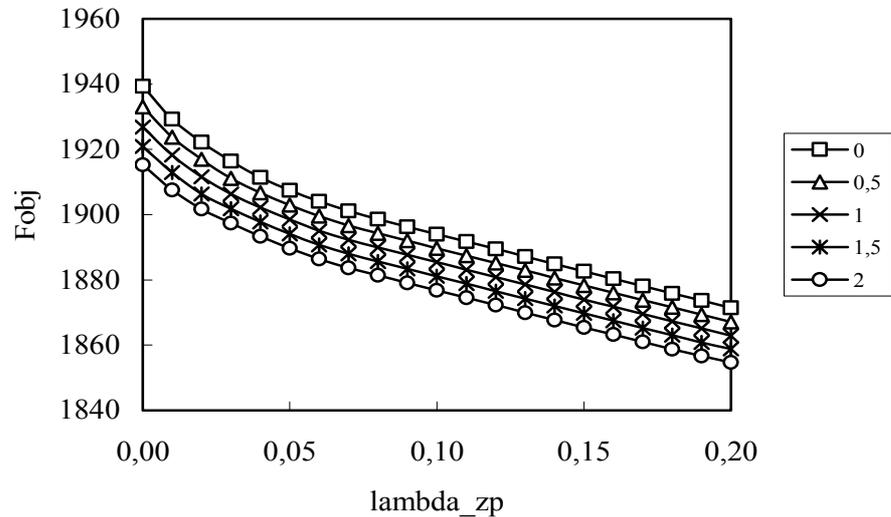
São apresentados graficamente os resultados do caso ilustrativo para o qual, na *Tabela 5.1* se listam valores representativos. Tomando como referência a situação de parâmetros de penalização,  $\lambda_{dsvn}$  e  $\lambda_{zp}$ , nulos (situação estocástica não-robusta), verifica-se que: uma variação nestes parâmetros, respectivamente, 1.0 e 0.1, provoca apenas uma ligeira diminuição da função objectivo robusta (2.78%), enquanto que a variabilidade diminui acentuadamente (32.07%) e as folgas de capacidade diminuem fortemente (89.30%); uma variação mais pronunciada dos parâmetros de penalização, respectivamente, 2.0 e 0.2, provoca um decréscimo pouco significativo da função objectivo (4.37%), mas não acentua as variações, quer na variabilidade (34.91%), quer no excesso de capacidade (90.51%).

**Tabela 5.1** Valores significativos da optimização estocástica do problema-exemplo *spplan\_ex1*.

Parâmetros $\lambda_{dsvn}; \lambda_{zp}$	Esperança da função objectivo	Esperança da variabilidade, $E_{dsvn}$	Esperança das folgas de capacidade, $E_{zp}$
0.0; 0.0	1939.33	12.66	304.35
1.0; 0.1	1885.38	8.60	32.56
2.0; 0.2	1854.64	8.24	28.88

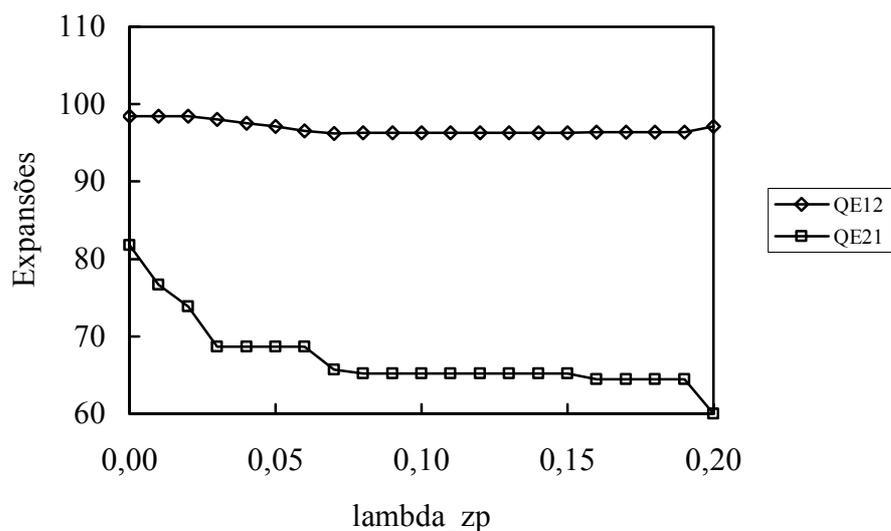
A razão para a diferença de ordens de grandeza dos parâmetros de penalização,  $\lambda_{dsvn}$  e  $\lambda_{zp}$ , deve-se ao facto de o primeiro apenas ser afectado pelos *NR* cenários, enquanto que o segundo depender de (*NP.NT.NR*) parcelas de folgas de capacidade. Desta forma, para o exemplo numérico testado, enquanto que na variabilidade apenas constam 4 parcelas (cenários), no excesso de capacidade consideram-se 80 parcelas de folgas (2 processos em 10 períodos e em 4 cenários). Por conseguinte, para uma harmonização das penalidades, o factor de penalização correspondente à variabilidade deverá ser cerca de uma ordem de grandeza superior ao factor de penalização das folgas de capacidade.

Observa-se a variação do valor da função objectivo estocástica com os valores dos parâmetros de penalização  $\lambda_{zp}$  e  $\lambda_{dsvn}$  na *Figura 5.5*, constatando-se que um aumento do parâmetro de excesso de capacidade  $\lambda_{zp}$  provoca uma diminuição continuada da função objectivo. Também, as linhas são praticamente paralelas para os diferentes valores do parâmetro de desvio, com  $\lambda_{dsvn}$  variando de 0 a 2, verificando-se a importância de carácter constante deste parâmetro.



**Figura 5.5** Variação do valor da função objectivo estocástica com  $\lambda_{zp}$  e  $\lambda_{dsvn}$

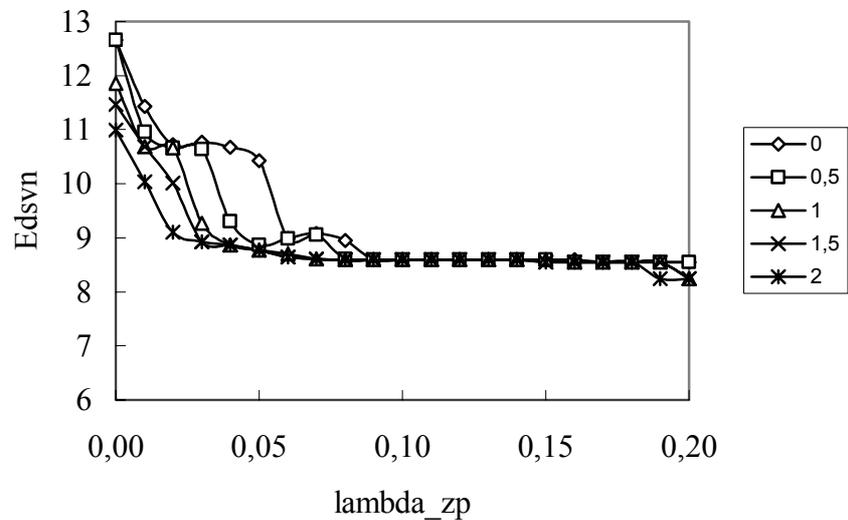
Na *Figura 5.6* representa-se a variação das expansões do processo 1 no período 2, QE12, e do processo 2 no período 1, QE21, com  $\lambda_{zp}$  (para  $\lambda_{dsvn} = 1.0$ ). Refira-se que ocorre uma expansão máxima do processo 1 no período 1 (QE11 = 100), para todos os valores dos parâmetros de penalização experimentados. Enquanto que a expansão QE12 é praticamente constante para os diferentes valores de  $\lambda_{zp}$ , a expansão QE21 sofre o efeito do aumento deste parâmetro no intuito de diminuir o excesso de capacidade, ajustando-o à procura incerta do componente C5.



**Figura 5.6** Variação das expansões com  $\lambda_{zp}$  (para  $\lambda_{dsvn} = 1.0$ )

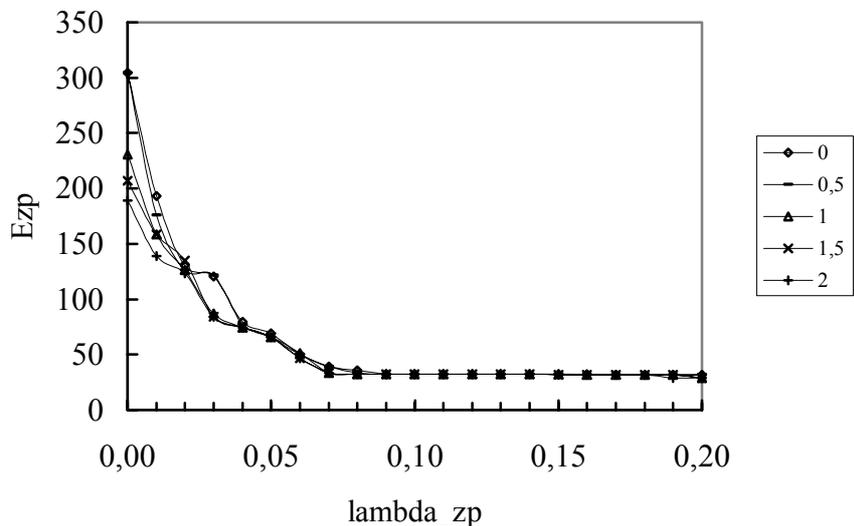
Atente-se na *Figura 5.7*, onde se pode observar a robustez da solução: através da variação do valor esperado do desvio parcial negativo com  $\lambda_{dsvn}$  (variando de 0 a

2) e  $\lambda_{zp}$ , ocorre uma diminuição relevante desta métrica com uma ligeira ponderação dos parâmetros de penalização. Essa diminuição é tanto mais rápida quanto maior for o valor do parâmetro  $\lambda_{dsvn}$ , mas atinge depois um patamar de estabilidade.



**Figura 5.7** Robustez da solução: variação do valor esperado do desvio parcial negativo com  $\lambda_{zp}$  e  $\lambda_{dsvn}$

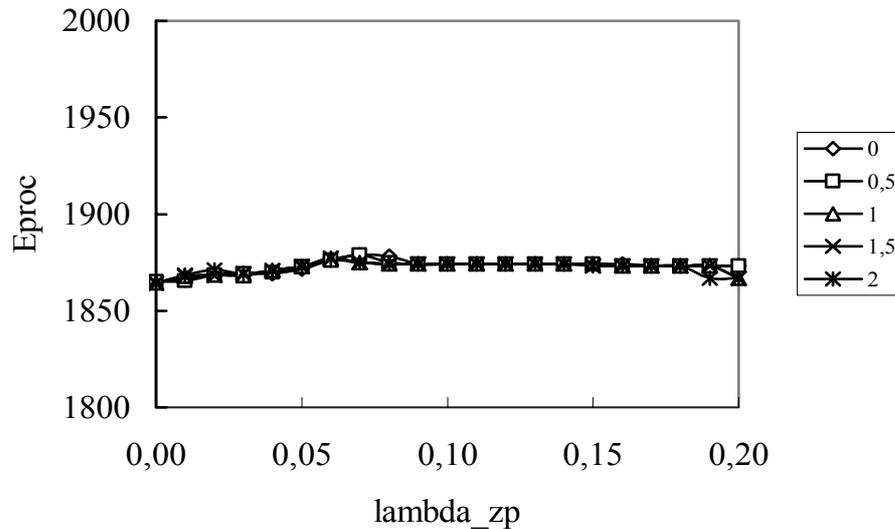
Também, na *Figura 5.8*, se verifica uma rápida diminuição do valor esperado das folgas (excessos de capacidade) com o aumento de  $\lambda_{zp}$  e  $\lambda_{dsvn}$  (de 0 a 2), constatando-se assim a robustez desejada do modelo. As folgas são penalizadas pelo parâmetro de desvio  $\lambda_{dsvn}$ , mas a sua diminuição é acentuada com a progressão do parâmetro  $\lambda_{zp}$ , atingindo-se depois uma zona de constância permanente (sensivelmente, a partir de  $\lambda_{zp} = 0,1$ ).



**Figura 5.8** Robustez do modelo: variação do valor esperado das folgas com  $\lambda_{zp}$  e  $\lambda_{dsvn}$

Na *Figura 5.9* representa-se a variação do valor esperado da procura total não-satisfeita com  $\lambda_{dsvn}$  (de 0 a 2) e  $\lambda_{zp}$ , observando-se que esta métrica não sofre um aumento superior a 1,0% para os valores dos parâmetros de penalização aqui apresentados. Por conseguinte, aumentando significativamente a robustez do planeamento efectuado, com diminuição dos valores esperados,  $Edsvn$  e  $Ezp$ , como

se comprova das figuras anteriores, não acarreta uma diminuição do serviço prestado, representado pela métrica da procura não-satisfeita.

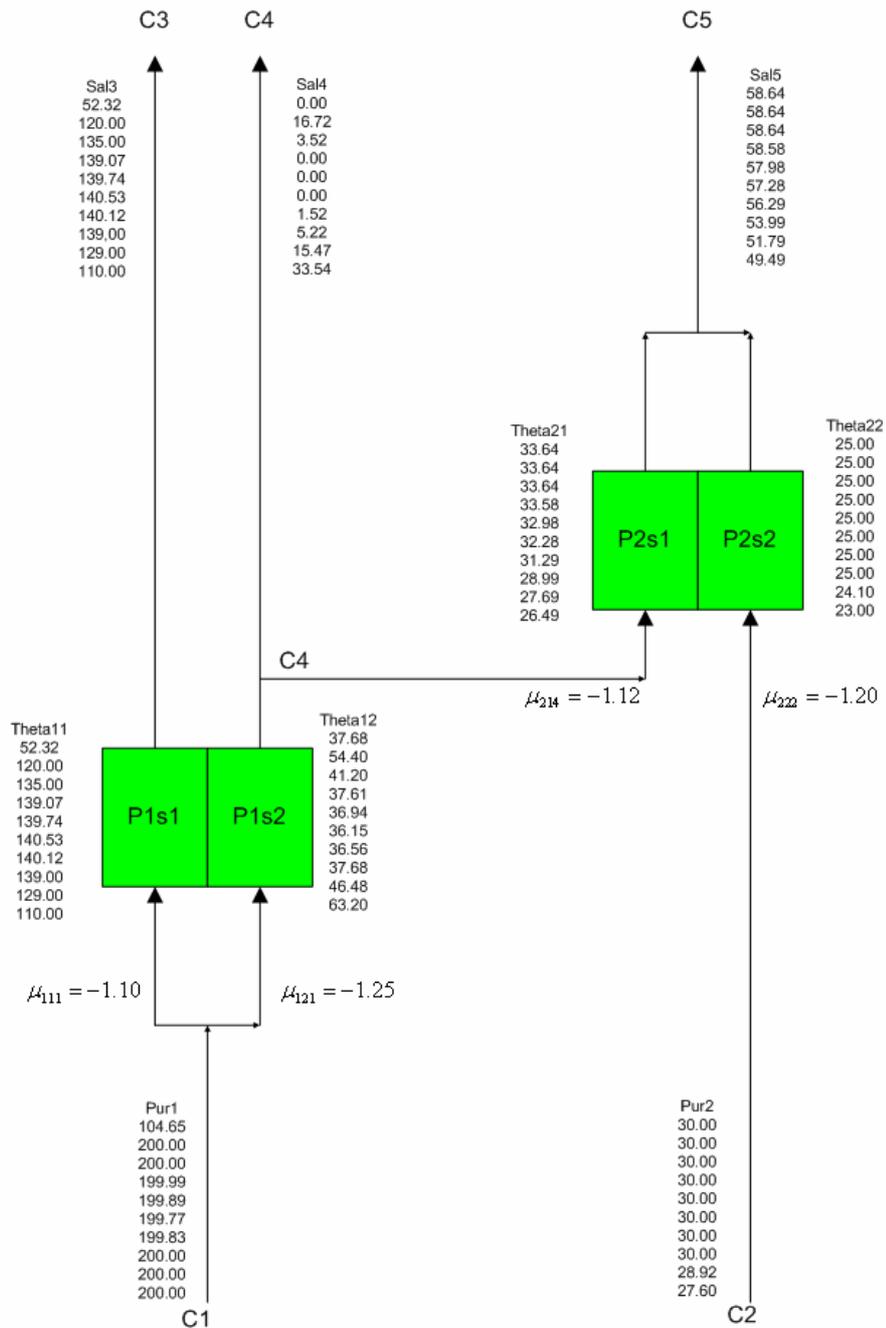


**Figura 5.9** Variação do valor esperado da procura total não-satisfeita com  $\lambda_{zp}$  e  $\lambda_{dsvn}$

Considerando que a solução obtida com uma ligeira ponderação dos parâmetros de penalização apresentará valores satisfatórios para a esperança dos desvios, a esperança dos excessos de capacidades e a esperança da procura total não-satisfeita, apresenta-se na *Figura 5.10* a estrutura da rede resultante com  $\lambda_{zp} = 0.1$  e  $\lambda_{dsvn} = 1.0$ , indicando-se também os fluxos (médios) de interesse. Refira-se que, para estes valores dos parâmetros de penalização, se verifica apenas uma ligeira diminuição da função objectivo robusta (cerca de 2.78%).

Procede-se a uma breve caracterização dos valores (médios) obtidos para os diversos fluxos.

- Os valores de compra do componente C1 encontram-se limitados: no período inicial, pela capacidade do processo 1; nos restantes períodos, pela sua disponibilidade (procura) máxima; para as compras do componente C2, estas atingem o máximo disponível nos primeiros 8 períodos de tempo, diminuindo ligeiramente nos dois últimos períodos, devido à diminuição do valor do componente C5, que acarreta uma menor utilização do componente C2 no esquema 2 do processo 2.
- Os valores das vendas do componente C3 encontram-se limitados: no primeiro período, pela capacidade do processo 1; nos períodos 2, 3, 8, 9 e 10, pela sua procura máxima; nos períodos intermédios, 4 a 7, não acompanha o aumento da sua procura, por utilização preferencial do esquema 2 no processo 1; quanto às vendas do componente C4, estas são diminutas ou nulas até à parte final do horizonte temporal (devido à diminuição inicial do seu valor de venda), onde atingem algum significado devido ao aumento do valor deste componente; as vendas do componente C5 são inicialmente limitadas pela capacidade do processo 2, diminuindo depois em virtude da diminuição do seu valor (e da venda preferencial do componente C4, em detrimento da sua utilização para produzir C5, pois o valor de C4 aumenta no final do horizonte).



**Figura 5.10** Estrutura e fluxos (médios) da rede de processos resultante para o exemplo *spplan\_ex1*, com  $\lambda_{zp} = 0.1$  e  $\lambda_{dsvn} = 1.0$ .

- No respeitante aos fluxos de produção no processo 1, estes exaurem a capacidade deste processo, havendo uma preponderância inicial do esquema 1 devido ao aumento de valor do componente C3 e diminuição do valor do componente C4; na parte final do horizonte temporal, ocorre a situação inversa, com aumento da utilização do esquema 2, pois o componente C4 aumenta de valor e o componente C3 diminui; quanto aos fluxos no processo 2, observa-se uma utilização máxima possível do esquema 2, devido ao limite superior da disponibilidade do componente C2, finalizando com uma ligeira diminuição pois o valor do componente

C5 também acaba por diminuir; o esquema 1 deste processo utiliza preferencialmente quase todo o componente C4 produzido no esquema 2 do processo 1, acabando por diminuir essa utilização na parte final do horizonte, por efeito conjugado do menor valor do componente C5 e aumento da venda do componente C4 (o seu valor sobe).

Quanto às expansões efectuadas (a sombreado): para o processo flexível 1, ocorrem duas expansões consecutivas, nos períodos 1 e 2, sendo a primeira máxima (100.0) e a segunda (96.31) adequada de forma a que a utilização deste processo fosse quase integral (mais de 98%, tendo como referência os fluxos médios); quanto ao processo flexível 2, propõe-se uma única expansão de capacidade, de nível tal (65.16) que a utilização é total nos primeiros períodos, diminuindo depois devido ao menor valor do componente C5, em detrimento do aumento de valor do componente C4. Refira-se que os valores obtidos são óptimos no respeitante à medida de penalização assumida, quer para a variabilidade, quer para os excessos da capacidade instalada.

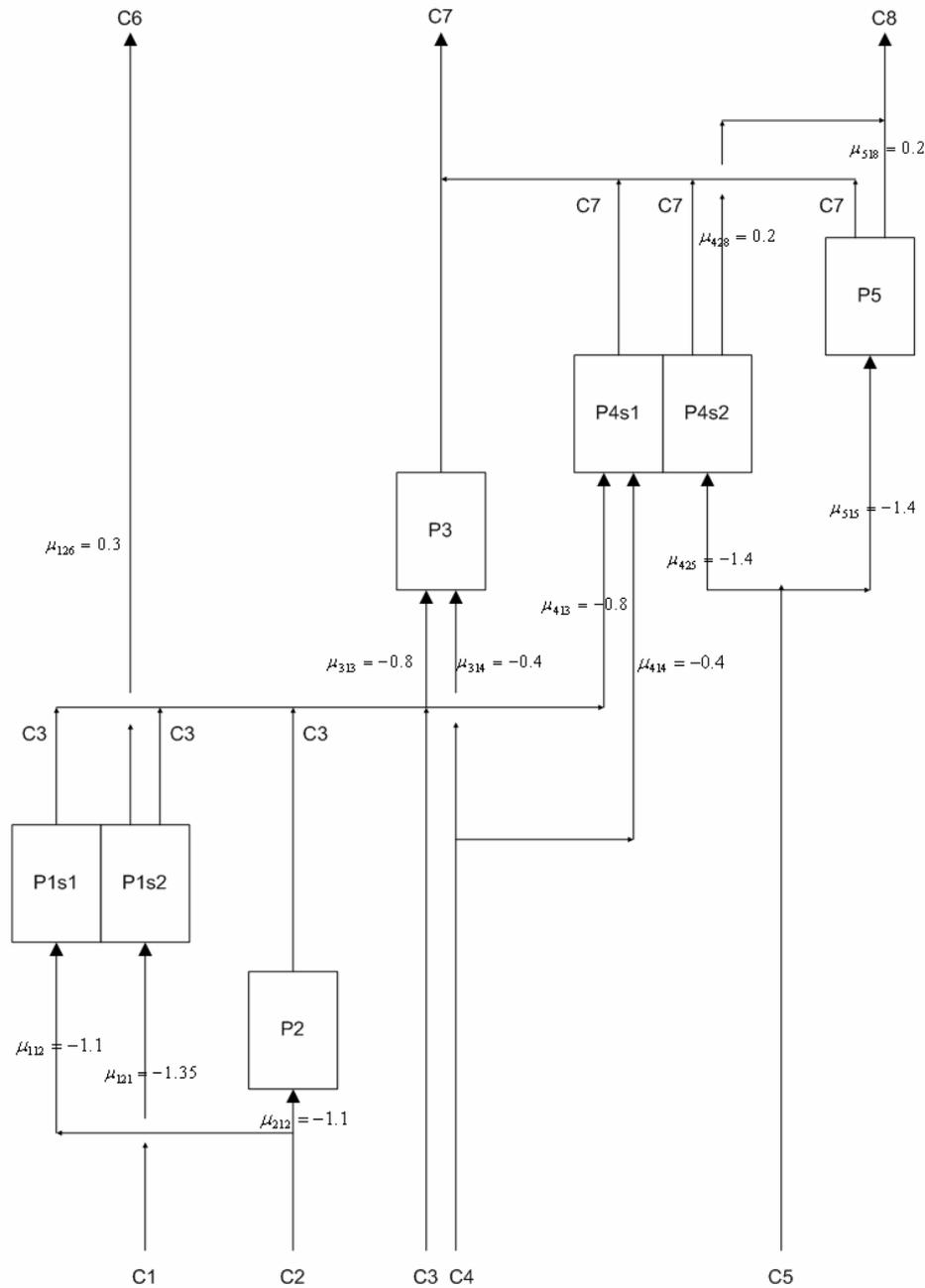
Assim, constata-se que o modelo implementado se apresenta sensível às variações dos preços de venda e compra dos diversos componentes, repercutindo essas variações na diferente utilização dos componentes pelos diversos processos. Também se verifica alguma robustez na solução apresentada, com as capacidades expandidas a acompanhar moderadamente as variações das procuras, não assumindo valores demasiado penalizadores para os respectivos excessos de capacidade. Por exemplo, a procura dos diversos componentes não é exaurida, embora as expansões efectuadas procurem acomodar a sua evolução.

### 5.1.2 Aplicação numérica: reagente crítico (*splan\_ex2*)

Seja a situação problemática representada na superestrutura da rede de processos constituída por cinco processos contínuos: o processo dedicado 3, existente inicialmente, produz o componente de elevado valor C7, utilizando os componentes C3 e C4, sendo que o componente C3 apresenta também um elevado valor e uma disponibilidade diminuta.

Deste modo, é de equacionar a produção de C3, a qual poderá ser realizada através dos componentes C1 ou C2, reagentes de compra: quer utilizando o processo flexível 1 com dois esquemas de produção, quer utilizando o processo dedicado 2; os parâmetros de produção no processo 2 e no esquema 1 do processo 1 são similares, mas o processo flexível 1 apresenta ainda a possível produção do componente C6, através do respectivo esquema 2.

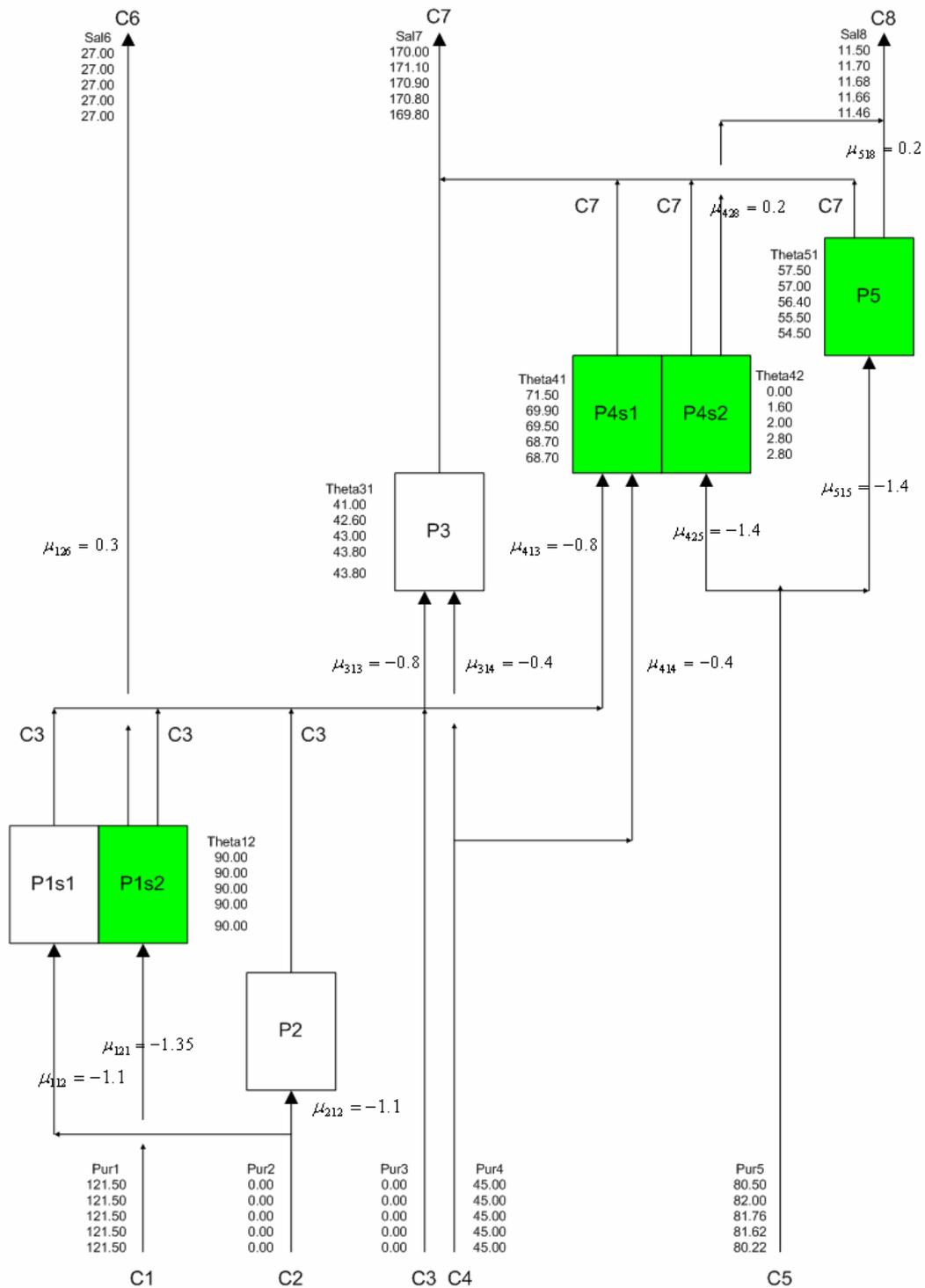
No entanto, também se equaciona a possível expansão do processo dedicado 3, surgindo como alternativa possível a implementação do processo flexível 4, com dois esquemas de produção: o esquema 1, similar ao processo 3; o esquema 2, que utiliza como reagente substituto o componente C5 (em vez dos componentes C3 e C4). Para equilibrar a possível expansão dos processos 3 e 4, supõe-se ainda a existência de um processo dedicado 5, possuindo os mesmos parâmetros que o esquema alternativo 2 do processo 4: assim se contrapõe as expansões do processo flexível 4, com as expansões conjuntas dos processos dedicados 3 e 5.



**Figura 5.11** Superestrutura da rede de processos para o exemplo *spplan\_ex2*.

Os componentes reagentes C1 a C5 possuem disponibilidade elevada e valor diminuto, excepto o componente C3 com baixa disponibilidade e elevado valor. Os componentes de venda, C6 a C8, possuem elevado valor, sendo a procura elevada para o componente C7 e relativamente mediana para os componentes C6 e C8. Para o modelo estocástico, supõe-se incerteza na procura do componente C7, a qual é descrita através de um número ( $NR = 4$ ) discreto de cenários. Os dados numéricos para o exemplo *spplan\_ex2* encontram-se no Apêndice 5-B, contemplando um horizonte temporal de ( $NT = 5$ ) períodos anuais.

Na *Figura 5.12* apresenta-se a estrutura da rede resultante para a situação problematizada no exemplo *spplan\_ex2*, referindo-se os fluxos médios de interesse para  $\lambda_{zp} = 0.1$  e  $\lambda_{dsvn} = 1.0$ .



**Figura 5.12** Estrutura e fluxos (médios) da rede de processos resultante para o exemplo *spplan\_ex2*, com  $\lambda zp = 0.1$  e  $\lambda dsvn = 1.0$ .

Seguidamente, procede-se a uma breve descrição dos resultados obtidos, sejam fluxos (médios) dos diferentes componentes e nos diversos processos, bem como dos valores óptimos relativos à expansão de capacidades.

- Quanto aos fluxos de compra dos componentes C3 e C2, eles são sempre nulos: o componente reagente C3 é relativamente oneroso e com pequena disponibilidade, pelo que é preferivelmente produzido através do esquema

2 do processo 1; o componente reagente C2 seria utilizado para produzir o referido componente C3, mas é preterido em favor da utilização do componente C1, o qual permite também a produção do componente C6, que apresenta um valor significativo; o componente reagente C1 é utilizado no esquema 2 do processo 1, sendo limitado pela capacidade máxima deste processo; os componentes C4 e C5 são utilizados complementarmente, de forma a satisfazer a procura do componente C7.

- Relativamente aos fluxos de venda, observa-se que: a procura do componente C7 é (em termos médios) satisfeita integralmente, através da utilização dos diversos processos 3, 4 e 5; o componente C6 é valorizado, sendo preferencialmente seleccionando o esquema 2 do processo 1, em detrimento do respectivo esquema 1 e do processo dedicado 2, que não prevêm a produção de C6; o componente C8 apresenta também alguma valorização, pois prefere-se também a sua produção através do processo dedicado 5, em vez de apenas a produção de C7.
- No respeitante aos fluxos de produção nos diferentes processos, verifica-se que: o processo flexível 1 utiliza única e integralmente o esquema 2, para produzir o componente de venda C6 além do reagente C3; o processo dedicado 2 não é implementado, bem como não é utilizado o esquema 1 do processo 1, com condicionamentos similares (apenas produção de C3); o processo dedicado 3, preexistente, não utiliza a totalidade da sua capacidade; em contraponto, o processo flexível 4 é utilizado na totalidade, de forma preferencial no esquema 1 (produz apenas C7) e de forma marginal no esquema 2 (produz também C8); o processo dedicado 5 é utilizado quase na totalidade, complementando a produção de C7 e ainda produzindo também C8.

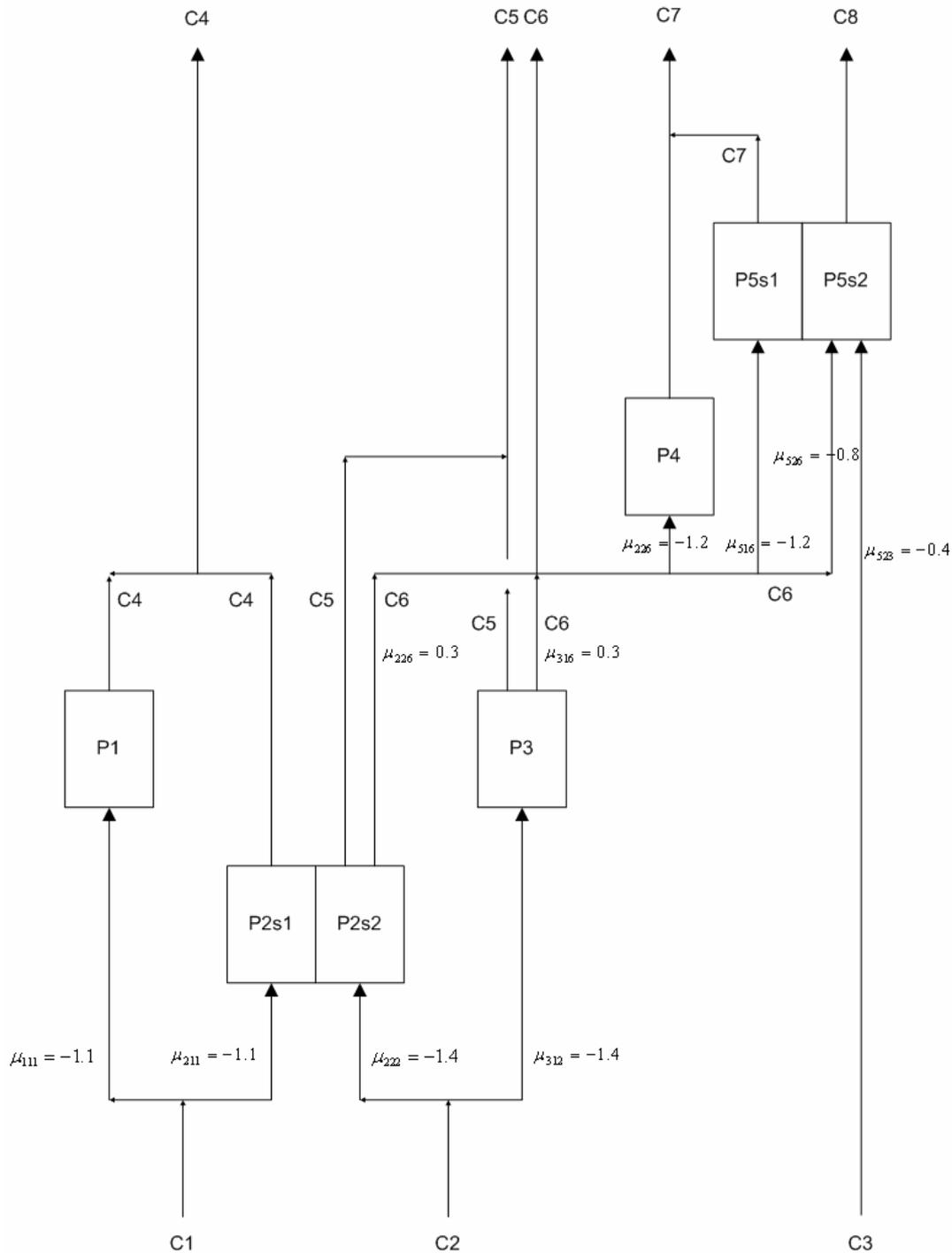
As expansões propostas (a sombreado, na *Figura 5.12*) ocorrem todas no primeiro período do horizonte temporal: para o processo flexível 1, ela é máxima (100.0) e integralmente utilizada, mas não se supõe uma segunda expansão; os processos 2 e 3 não são expandidos; para o processo flexível 4, a capacidade (79.44) é também integralmente utilizada (atendendo aos fluxos médios) mas o esquema 2 apenas aproveita a não operação do esquema 1, este preferível para a produção do componente C7; para o processo dedicado 5, propõe-se a expansão de capacidade (63.89) no sentido de contribuir para a produção de C7, mas também do produto de venda C8 e utilizando o reagente alternativo C5.

Da descrição acima se conclui pelo interesse numa via alternativa para a produção do componente de interesse C7, nomeadamente, utilizando o reagente substituto C5 e produzindo também C8. Também se revela economicamente favorável a produção do reagente limitativo C3, preferencialmente através da via que produz também C6, em detrimento da via que produz apenas C3. Assim, desenvolvendo um estudo comparativo das diferentes vias de produção, poderão estimar-se quais os parâmetros desses processos que os tornam economicamente preferíveis.

### 5.1.3 Aplicação numérica: produto crítico (*spplan\_ex3*)

Suponha-se a problematização decorrente da superestrutura da rede de processos na *Figura 5.13*, constituída por cinco processos contínuos: o processo

dedicado 3, existente inicialmente, produz o componente de elevado valor C5 utilizando o reagente C2, sendo que o produto C6 não é de interesse económico: apresenta um diminuto valor e uma procura também diminuta (ou mesmo nula, se assim se entender).



**Figura 5.13** Superestrutura da rede de processos para o exemplo *spplan\_ex3*.

Desta forma, deseja-se equacionar a utilização de C6, a qual poderá ser realizada através processos 4 e 5, a implementar: seja utilizando o processo flexível 5, com o esquema de produção 1 que produz C7, e com o esquema 2 que utiliza também o componente reagente C3 para produzir o produto C8; seja utilizando o processo dedicado 4, com os mesmos parâmetros de produção que o esquema 1 do processo 5.

Por conseguinte, balanceia-se a utilização do (sub) produto C6 através da utilização de um processo dedicado ou flexível.

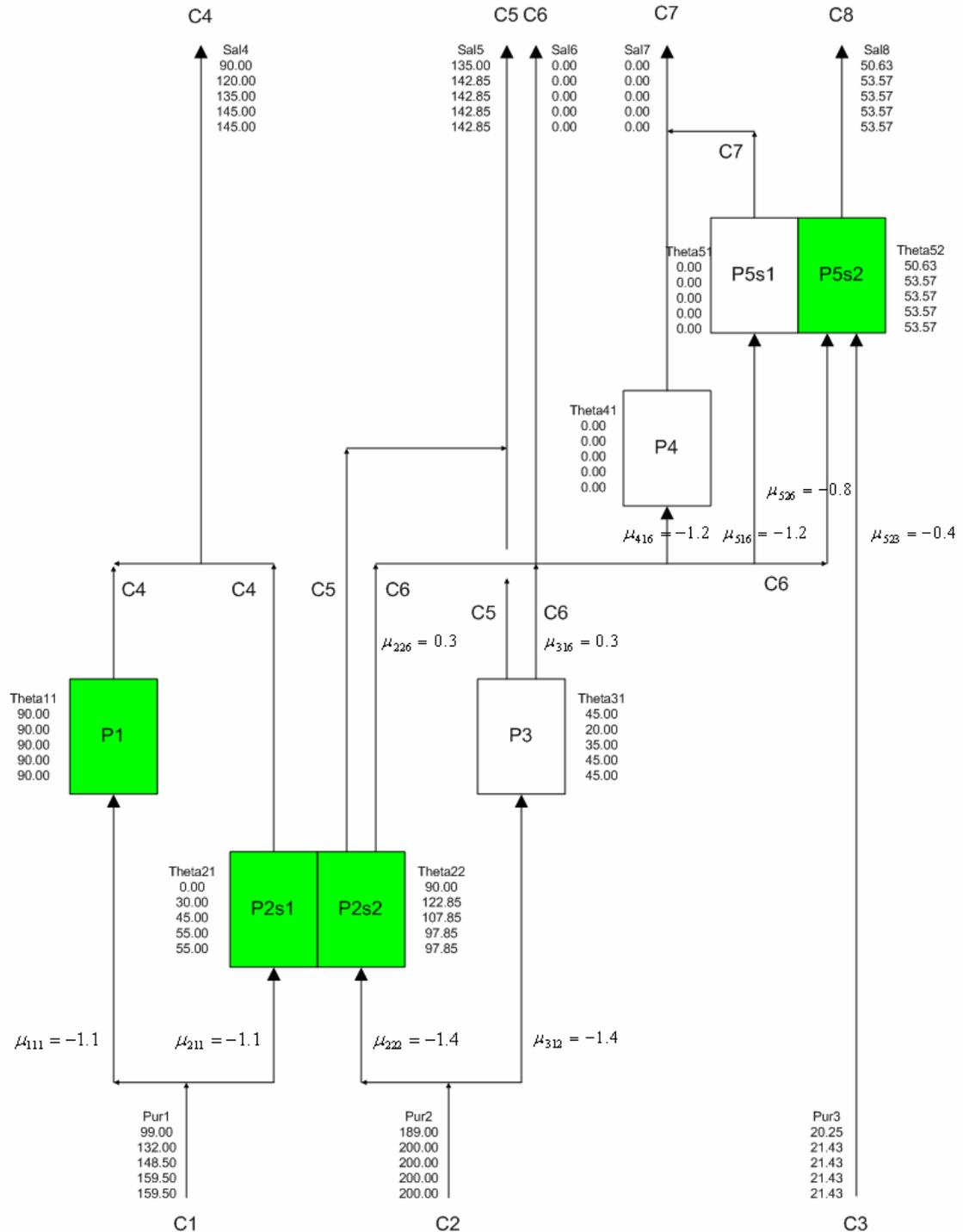
Não obstante, supõe-se também a expansão do processo dedicado 3, surgindo como alternativa possível a implementação do processo flexível 2, com dois esquemas de produção: o esquema 1, que utiliza o reagente C1 para produzir C4; o esquema 2, similar ao processo 3. Para equilibrar a possível expansão dos processos 2 e 3, equaciona-se também a existência do processo dedicado 1, possuindo similares parâmetros aos do esquema 1 do processo 2. Desta maneira, contrapõem-se as expansões do processo flexível 2 com as expansões conjuntas dos processos dedicados 1 e 3.

Os componentes de venda, C4 a C8, possuem elevado valor, sendo a sua procura mediana ou elevada, excepto para o componente C6, o qual apresenta valor e procura diminutos. Os componentes reagentes, C1 a C3, possuem disponibilidade elevada e valor diminuto. No modelo estocástico, apresenta-se incerteza na procura do componente 5, a qual é descrita através de um número ( $NR = 4$ ) discreto de cenários. Os dados numéricos para o exemplo *spplan\_ex3* encontram-se em Apêndice 5-B, contemplando um horizonte temporal de ( $NT = 5$ ) períodos anuais.

Apresenta-se, na *Figura 5.14*, a estrutura da rede resultante para o exemplo *spplan\_ex3*, com  $\lambda_{zp} = 0.1$  e  $\lambda_{dsvn} = 1.0$ , indicando-se ainda os fluxos médios de interesse. Procede-se a uma descrição sumária dos resultados óptimos obtidos para este terceiro exemplo estocástico.

- Quanto aos fluxos de compra: do componente C1 são delimitados pela utilização máxima possível, quer no processo dedicado 1, quer no esquema 1 do processo 2, no intuito de produzir tanto quanto possível do componente de interesse C4; do componente C2 são estrangidos pela disponibilidade máxima deste componente, excepto para o primeiro período onde se impõe a restrição de utilização máxima decorrente das capacidades dos processos 2 e 3, onde o componente C2 é utilizado; do componente C3 derivam-se directamente da sua utilização no esquema 2 do processo 5, onde se exaure a capacidade instalada (excepto no primeiro período, onde ainda não se produz todo o componente C6 possível, devido a uma expansão do processo 2 no segundo período).
- Relativamente aos fluxos de venda: do componente C4, como combinação da utilização máxima do processo 1, com a utilização crescente do esquema 1 do processo 2, devido ao valor do componente em questão; do componente C5, delimitado pela disponibilidade máxima do componente C2, excepto no primeiro período onde a restrição decorre das capacidades disponíveis nos processos 2 e 3; dos componentes C6 e C7 que são nulos, devido ao seu evidente desinteresse económico; do componente C8, que exaure a capacidade instalada no processo 5, por sua vez decorrente da disponibilidade do componente C6 produzido.
- No respeitante aos fluxos de produção: no processo dedicado 1, a capacidade é integralmente utilizada na produção de C4; no processo flexível 2, a capacidade também é integralmente utilizada, preferencialmente no esquema 2 para produzir C5 e C6, mas com interesse crescente ao longo do horizonte pelo esquema 1, onde se produz C4; no processo 3, dedicado e preexistente, a capacidade nem sempre é

totalmente utilizada; os fluxos são nulos no processo dedicado 4 e no esquema 1 do processo flexível 5, pois a transformação do subproduto C6 no componente de interesse C7 não é seleccionada; é economicamente preferível o esquema 2 do processo 5 para produção de C8, onde apenas não exaure completamente a capacidade instalada no primeiro período, pois não dispõe de C6 suficiente.



**Figura 5.14** Estrutura e fluxos (médios) da rede de processos resultante para o exemplo *spplan\_ex3*, com  $\lambda_{zp} = 0.1$  e  $\lambda_{dsvn} = 1.0$ .

As expansões propostas (a sombreado, na *Figura 5.14*) são as seguintes: para o processo dedicado 1, expansão máxima (100) única no período inicial, com utilização total dessa capacidade ao longo do horizonte temporal; no processo flexível 2, expansão máxima (100) no primeiro período com utilização total, seguida de uma segunda expansão (69.84) no segundo período, verificando-se sempre utilização total e uma preferência crescente pelo esquema 1, onde se produz o componente C4; quanto ao processo 3, dedicado e preexistente, não ocorre qualquer expansão, embora nalguns períodos a capacidade deste processo seja completamente utilizada, já que se utiliza preferencialmente o esquema 2 do processo 2 para produzir C5; o processo dedicado 4 não é de interesse, pelo que não ocorre qualquer expansão, já que a utilização do componente C6 na produção de C7 não é seleccionada; no processo flexível 5 ocorre uma expansão (59.52) no período inicial, utilizando apenas o esquema 2 para a produção de C8 pois, como já se viu, a produção de C7 no esquema 1 é negligenciada.

Por conseguinte, mostra-se de interesse a utilização do componente C6, de preferência pela via que produz C8, sendo possível analisar os parâmetros desta via que a tornam preferível relativamente à da produção de C7. Quanto à expansão do processo 3, dedicado e preexistente, não se supõe qualquer expansão, sendo preferível equacionar a instalação do processo 2, flexível e que permite também a produção do componente C4. Os parâmetros relativos a este componente C4 são tão favoráveis que se revela de interesse a instalação de um processo dedicado à sua produção.

## 5.2 Abordagem generalizada ao dimensionamento e sequenciamento de processos descontínuos: robustez e períodos de tempo

No âmbito do dimensionamento e sequenciamento de processos descontínuos, apresenta-se uma abordagem em ambiente incerto, generalizadora da usual perspectiva determinística: prevê-se variabilidade dos cenários (procuras e retornos dos produtos) que se trata de forma estocástica. Assim, enquanto a perspectiva determinística se encontra subjacente, pois corresponde à consideração de um cenário único, propõe-se que a configuração óptima das dimensões dos processos em cada estágio enquadre diferentes cenários prováveis.

O modelo proposto engloba a optimização das decisões de investimento (de longo prazo), considerando o sequenciamento (de curto prazo) na rede de processos descontínuos. Então, é curial considerar que se exige um elevado investimento associado a um longo prazo de retorno, além de que os modelos determinísticos não tratam o risco inerente a um alargado horizonte temporal, pois é habitual considerar-se apenas um único período temporal no sequenciamento. Esta limitação decorre da dificuldade de resolução do problema de sequenciamento, o qual é inserido no problema de dimensionamento.

É utilizado o enquadramento de 2SSP, com promoção da robustez pois penalizam-se desvios na solução e na modelação, relaxando-se as restrições de integralidade sobre as variáveis de segunda fase. Concretamente, ocorrem variáveis de controlo (*recourse*) de carácter inteiro e binário relacionadas com o sequenciamento e restrições de precedências, que inviabilizariam o tratamento do

problema de segunda fase. De facto, a Programação Estocástica Inteira (*Stochastic Integer Programming*, SIP) com variáveis inteiras em segunda fase encontra-se ainda num estágio de desenvolvimento bastante incipiente, conhecendo-se da literatura (Caroe e Schultz [1999], Ahmed *et al.* [2003]) métodos que funcionam apenas com problemas de teste, de pequena dimensão. Em consequência, as variáveis de segunda fase assumem-se como contínuas (por exemplo, o número de *batches*), ocorrendo apenas variáveis binárias em primeira fase, as discriminadoras das decisões quanto às dimensões seleccionadas.

O modelo estocástico com robustez visa a maximização de uma medida robusta do VLA, para o que se deverá seleccionar as dimensões discretas dos processos descontínuos, bem como o número de processos em paralelo operando desfasadamente (*out-of-phase*). Para a correcta avaliação do VLA, é pressuposto básico a consideração de diversos períodos de tempo, onde os retornos são avaliados de forma probabilística, dada a incerteza associada às quantidades procuradas e aos retornos unitários de cada produto.

O modelo determinístico generalizado a este enquadramento estocástico consiste no *batch\_milp\_fmsz* (*Flowshop/ Multiple machine/ SPC/ Zero wait*), analisado na Sec. 4.2, dado que.

- para aplicações realistas, com um número e quantidade de produtos adequada, deve considerar-se a opção dos processos múltiplos, pois em contrário ocorrem situações de inadmissibilidade;
- a selecção de campanhas de produto único (SPC) decorre da impraticabilidade de aplicar campanhas de produtos múltiplos (MPC) em ambiente *multiple machine*, quer por insuficiência do modelo correspondente (*batch\_milp\_fmz*, ver Sec. 4.2), quer por impossibilidade de índole teórica (aplicação múltipla de um problema de TSP a cada processo em paralelo);
- decorrendo da análise efectuada na Sec. 4.2, prevê-se que a configuração a seleccionar segundo SPC exceda a correspondente configuração em MPC, em cerca de 5% dos custos de investimento, correspondentes a cerca de metade dos estágios com acréscimo de uma ordem de dimensão (valores obtidos no contexto de processo único por estágio); ou seja, a configuração seleccionada com SPC estará já sobredimensionada, o que permitirá absorver a introdução de novos produtos, ou mesmo o crescimento de procuras dos produtos existentes.

Por conseguinte, o enquadramento de suporte é similar ao do modelo *batch\_milp\_fmsz*, em termos de definição de parâmetros e índices, variáveis e dados, bem como os pressupostos em que foi desenvolvido. Generaliza-se em termos de incerteza (cenários aleatórios) e de multiplicidade de períodos temporais, ocorrendo também a inversão de designação entre os índices dos produtos (era  $i$ , passa a ser  $j$ ) e dos estágios (era  $j$ , passa a  $i$ ). Justifica-se esta inversão de ordem relativa no sentido em que o propósito principal da optimização é a configuração dos estágios (definição das dimensões e número de processos), bem como se uniformiza a nomenclatura com a do modelo robusto para planeamento de processos flexíveis, proposto na Sec. 5.1.

Quanto à nomenclatura utilizada, ela é similar à considerada para o correspondente modelo determinístico *batch\_milp\_fmsz*, bem como para o modelo

robusto *spplan\_milp\_flex*, introduzindo-se alguns novos parâmetros, dados e variáveis, que se listam em seguida:

- $M, NC, NP(i), NR, ,$  números de estágios de produção  $i$ , de componentes ou produtos  $j$ , de processos de produção  $p(i)$  por estágio, de cenários discretos  $r$ , de dimensões discretas  $s(i)$  dos processos em cada estágio, de períodos de tempo  $t$ ;
- $NS(i), NT$  – quantidades procuradas não-satisfeitas pela produção do produto  $j$ , no período  $t$  e cenário  $r$ ;
- $\lambda dsv, \lambda flg, \lambda qns$  – parâmetros de penalização de desvios de variabilidade, de folgas de capacidades, de procuras não-satisfeitas;
- $ret$  – valores (incertos) unitários de retorno (valores líquidos) dos produtos, em cada período  $t$  e cada cenário  $r$ ;
- $Qns$  – quantidades procuradas não-satisfeitas pela produção do produto  $j$ , no período  $t$  e cenário  $r$ ;
- $folg$  – folgas de capacidades em cada estágio  $i$ , referentes à globalidade dos *batches* de cada produto  $j$ , em todos os períodos  $t$  e cenários  $r$ ;
- $W$  – quantidades globais produzidas do produto  $j$ , no período  $t$  e cenário  $r$ ;
- $tcamp$  – tempos de campanha (em SPC) relativos a cada produto  $j$ ;
- $H$  – horizonte temporal para sequenciamento da produção, constante para todos os períodos.

Neste modelo estocástico propõe-se maximizar o VLA esperado,  $\xi$ , definido através do retorno esperado a menos dos custos de investimento, os quais ocorrem apenas no primeiro período; incorpora-se robustez no modelo, sujeitando-se a penalizações (em termos dos valores médios), quer por variabilidade,  $dsv_n$ , no âmbito dos diversos cenários discretos, no sentido de que o valor associado a cada cenário se mantenha na proximidade do valor esperado óptimo, quer por não-satisfação das quantidades procuradas,  $Qns$ , ou por penalização das folgas de capacidades,  $folg$ :

$$\text{sb.1} \quad [max] \Phi = \sum_{r=1}^{NR} prob_r \xi_r - \lambda dsv \sum_{r=1}^{NR} prob_r . dsv_n_r - \lambda qns \sum_{r=1}^{NR} \frac{prob_r}{NC.NT} \left( \sum_{j=1}^{NC} \sum_{t=1}^{NT} Qns_{jtr} \right) - \lambda flg \sum_{r=1}^{NR} \frac{prob_r}{M.NC.NT} \left( \sum_{i=1}^M \sum_{j=1}^{NC} \sum_{t=1}^{NT} folg_{ijtr} \right)$$

Cada parcela probabilística  $\xi$  corresponde ao VLA calculado perante cada cenário  $r$ , definindo-se como o montante actualizado dos retornos das vendas (valores líquidos, a menos dos montantes das compras de matérias e dos custos de operação dos processos) menos os custos de investimento (discretos, de acordo com a discretização das capacidades em cada estágio):

$$\text{sb.2} \quad \xi_r = \sum_{j=1}^{NC} \sum_{t=1}^{NT} ret_{jtr} W_{jtr} - \sum_{i=1}^M \sum_{s=1}^{NS(i)} \sum_{p=1}^{NP(i)} c_{isp} y_{isp}, \quad \forall r$$

Utiliza-se uma métrica linear de penalização da variabilidade, que é definida pelo desvio negativo,  $dsv_n$ , do VLA (em termos efectivos, do retorno) relativamente ao seu valor esperado, em cada cenário aleatório  $r$ :

$$\text{sb.3} \quad dsvn_r \geq \sum_{r'=1}^{NR} (prob_{r'} \xi_{r'}) - \xi_r, \quad \forall r$$

A procura não-satisfeita de um qualquer produto  $j$ ,  $Qns_{jtr}$ , é definida em cada período de tempo,  $t$ , e para cada cenário,  $r$ , como a folga da restrição correspondente (pela diferença entre as quantidades procuradas  $Q_{jtr}$  e produzidas  $W_{jtr}$ ), modelando-se através do seguinte grupo de restrições:

$$\text{sb.4} \quad W_{jtr} + Qns_{jtr} = Q_{jtr}, \quad \forall j, t, r$$

$$\text{sb.5} \quad Qns_{jtr} \geq 0, \quad \forall j, t, r$$

As quantidades globalmente produzidas de cada produto,  $W_{jtr}$ , em cada cenário aleatório e para cada período de tempo, associam-se aos correspondentes números de *batches* (agregados),  $nc_{ijsptr}$ ,

$$\text{sb.6} \quad S_{ij} W_{jtr} \leq \sum_{s=1}^{NS(j)} \sum_{p=1}^{NP(j)} dv_{is} nc_{jisptr}, \quad \forall j, i, t, r$$

Pelo que, a definição das folgas globais ( $folg_{ijtr}$ ) das capacidades implementadas decorre directamente de:

$$\text{sb.7} \quad S_{ij} W_{jtr} + folg_{ijtr} = \sum_{s=1}^{NS(i)} \sum_{p=1}^{NP(i)} dv_{is} nc_{ijsptr}, \quad \forall i, j, t, r$$

Note-se que o número de *batches*,  $nc_{ijsptr}$ , corresponde ao produto-agregação das variáveis ( $n_{jtr} \cdot y_{isp}$ ), para o que se consideram também os seguintes grupos de restrições:

$$\text{sb.8} \quad nc_{ijsptr} - nc_{ijspp}^{Upp} y_{isp} \leq 0, \quad \forall i, j, s, p, t, r$$

$$\text{sb.9} \quad \sum_{s=1}^{NS(i)} \sum_{p=1}^{NP(i)} y_{isp} = 1, \quad \forall i$$

$$\text{sb.10} \quad \sum_{s=1}^{NS(i)} \sum_{p=1}^{NP(i)} nc_{ijsptr} - n_{jtr} = 0, \quad \forall i, j, t, r$$

Associados aos números de *batches*, deve também determinar-se os tempos de campanha,  $tcamp_{jtr}$ , bem como verificar a satisfação do horizonte temporal,  $H$ :

$$\text{sb.11} \quad \sum_{s=1}^{NS(i)} \sum_{p=1}^{NP(i)} \left( \frac{\tau_{ij}}{p(i)} nc_{ijsptr} \right) - tcamp_{jtr} \leq 0, \quad \forall i, j, t, r$$

$$\text{sb.12} \quad \sum_{j=1}^{NC} tcamp_{jtr} \leq H, \quad \forall t, r$$

Da conjugação das diferentes condições, obtém-se o modelo estocástico *sbatch\_milp\_fmsz* para dimensionamento e sequenciamento de processos descontínuos, visando a maximização do VLA com robustez, na solução e no modelo, configurando campanhas de produto único (SPC) e processos múltiplos (*multiple*

*machine*) em cada estágio, em ambiente *flowshop* com política de espera nula (*zero wait, ZW*):

**Modelo *spbatch\_milp\_fmsz*:**

$$\text{sb.13-a} \quad \begin{aligned} [\max] \Phi = & \sum_{r=1}^{NR} prob_r \xi_r - \lambda dsv \sum_{r=1}^{NR} prob_r . dsv_n_r - \lambda qns \sum_{r=1}^{NR} \frac{prob_r}{NC.NT} \left( \sum_{j=1}^{NC} \sum_{t=1}^{NT} Qns_{jtr} \right) \\ & - \lambda flg \sum_{r=1}^{NR} \frac{prob_r}{M.NC.NT} \left( \sum_{i=1}^M \sum_{j=1}^{NC} \sum_{t=1}^{NT} folg_{ijtr} \right) \end{aligned}$$

sujeito a,

Relações *sb.2—sb.4*

Relações *sb.7—sb.12*

$$\text{sb.13-b} \quad folg_{ijtr}, nc_{ijsptr}, n_{jtr}, Qns_{jtr}, tcamp_{jtr}, W_{jtr} \geq 0, \quad \forall i, j, s, p, t, r$$

$$\text{sb.13-c} \quad y_{isp} \in \{0,1\}, \quad \forall i, s, p$$

O modelo *spbatch\_milp\_fmsz* apresenta um número relativamente económico de variáveis binárias (*M.NS.NP*), dado que apenas são deste tipo as variáveis associadas à selecção da dimensão e número de equipamentos, em cada estágio. Contudo, o número de variáveis contínuas é muito elevado ( $NT.NR.(4.NC + M.NC.NS.NP + M.NC)$ ), considerando-se aqui, para cada período de tempo e cenário discreto: os 4 grupos de variáveis referentes aos tempos de campanha, aos números de *batches*, às quantidades produzidas, às procuras não-satisfeitas; os números de *batches* desagregados, relativos também a cada combinação estágio, produto, dimensão e número de processos; as folgas de capacidades (globais), em cada estágio e relativas a cada produto. Ainda elevado é o número de restrições ( $M + 2.NR + NT.NR.(NC + M.NC.NS.NP + 3.M.NC + 1)$ ), sendo respectivamente: consideradas as restrições relativas à selecção, única em cada estágio, dum dado número de processos com uma dada dimensão; dois grupos de restrições, onde são definidos os VLAs e os desvios lineares negativos; definidas as quantidades procuradas não-satisfeitas; as restrições lógicas quanto aos números semicontínuos de *batches*; três grupos de restrições, respeitando às folgas de capacidade (globais), à associação entre os números de *batches*, semicontínuos e contínuos, à determinação dos tempos de campanha; finalmente, a satisfação do horizonte temporal disponível.

Então, trata-se da aplicação deste modelo generalizado, para o dimensionamento e sequenciamento de processos descontínuos, ilustrando-se numericamente duas situações problemáticas: considerando o dimensionamento de forma a satisfazer procuras incertas num só período de tempo (“estático”), generalizando-se o problema determinístico de minimizar os custos de investimento a uma forma estocástica e robusta; também, assumindo as procuras incertas num ambiente multiperíodo, dito “dinâmico”, maximizando-se o VLA robusto do investimento realizado.

### 5.2.1 Aplicação numérica: robustez e período único (*spbatch\_ex1*)

Seja a situação problemática onde se pretende o dimensionamento dos processos descontínuos, considerando o respectivo sequenciamento com campanhas

de produto único (SPC) e política de espera nula (ZW), mas de forma a satisfazer a procura incerta dos componentes a produzir, num só período ou horizonte temporal (problema “estático” mas estocástico). Esta procura é modelada através de um conjunto discreto de cenários, aos quais é associada uma probabilidade de ocorrência, sendo penalizada a procura não-satisfeita.

A finalidade não consiste em maximizar o VLA, já que se considera apenas um período temporal, sendo o VLA usualmente considerado no âmbito multiperíodo (problema “dinâmico”), dados os elevados custos de investimento envolvidos e que não possibilitarão o retorno do investimento (*payback*) num só período temporal. Então, não se considerando os valores de retorno (ver Apêndice 5-D), formulando-se  $ret_{jtr} = 0$ . em conformidade com a finalidade acima descrita, obtém-se o mesmo valor para qualquer variável  $\xi$  e não se verificarão desvios nesta variável, pelo que  $dsvn_r = 0$ . Dado que o propósito desta optimização robusta é a satisfação das procuras incertas, não se penalizam as folgas de capacidade,  $\lambda flg = 0.0$ , pelo que a função objectivo em *sb.13-a* se reduz à minimização robusta dos custos de investimento, incluindo-se apenas as parcelas de penalidade pela procura não-satisfeita,

$$sb.14 \quad [min] \Psi = \sum_{j=1}^M \sum_{s=1}^{NS(i)} \sum_{p=1}^{NP(i)} c_{isp} \mathcal{Y}_{isp} + \lambda qns \sum_{r=1}^{NR} \frac{prob_r}{NC \cdot NT} \left( \sum_{j=1}^{NC} \sum_{t=1}^{NT} Qns_{jtr} \right)$$

Assim, utiliza-se o modelo definido pelo conjunto das equações em *sb.13*, onde se pode negligenciar: o índice dos períodos temporais,  $t$ , já que se considera apenas um período de tempo; os grupos de restrições referentes à definição das variáveis do VLA,  $\xi$ , que apresentarão valor cenário-independente, e dos respectivos desvios,  $dsvn$ , que consequentemente serão nulos.

Pretende-se analisar o efeito da utilização de um diferente número de cenários no âmbito deste modelo estocástico, o qual se reduz ao problema determinístico quando se considere apenas um cenário.

Os dados utilizados nas diversas instâncias são apresentados, de forma compreensiva, em Apêndice 5-E, encontrando-se descritos na *Tabela 5.2* os números de variáveis, binárias e contínuas, de restrições, bem como os tempos médios de resolução para cada uma das instâncias tratadas (EX1A a EX1F).

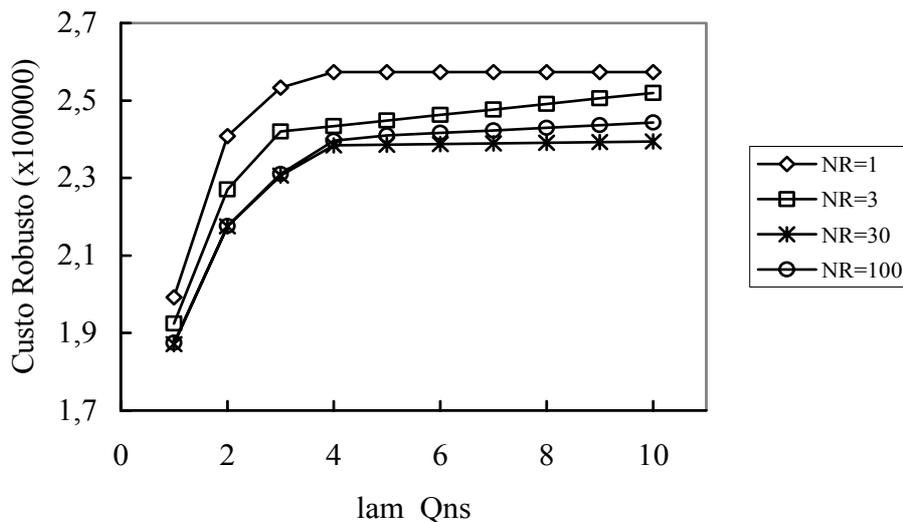
**Tabela 5.2** Parâmetros, variáveis e restrições das instâncias para o exemplo numérico *spbatch\_ex1*, configurando:  $NC = 4$ ;  $M = 3$ ;  $NS = 5$ ;  $NP = 3$ ;  $NT = 1$ .

Exemplos numéricos	Parâmetros $NR$	Variáveis binárias	Variáveis contínuas	Restrições	Tempos execução (s)
EX1A	1	45	209	226	0.33
EX1B	3	45	603	672	1.68
EX1C	7	45	1391	1564	7.48
EX1D	15	45	2967	3348	21.46
EX1E	30	45	5922	6693	91.60
EX1F	100	45	19712	22303	940.31

Embora o número de variáveis binárias se mantenha constante para as diversas instâncias, constata-se que tanto o número de variáveis contínuas, como o

número de restrições, varia de forma linear com o número de cenários,  $NR$ ; a variação do tempo médio de resolução é aproximadamente quadrática.

Representa-se graficamente a variação dos diferentes estimadores (custo robusto,  $\psi$ , valor esperado da procura não-satisfeita,  $Eproc$ , valor esperado das folgas de capacidade,  $Efolg$ , custo não-robusto,  $Ecsi$ ), com o aumento conjugado do número de cenários,  $NR$ , e de penalização por procura não-satisfeita,  $\lambda qns$ , procedendo-se à sua análise. Devido a uma questão de quase coincidência das diferentes linhas representadas, o que dificulta a sua correcta visualização, em alguns dos gráficos podem não se encontrar as seis linhas representativas das diferentes instâncias, de  $NR = 1$  (EX1A) a  $NR = 100$  (EX1F). Para os referidos estimadores, associados à presente modelação, encontram-se as suas definições em Apêndice 5-D.



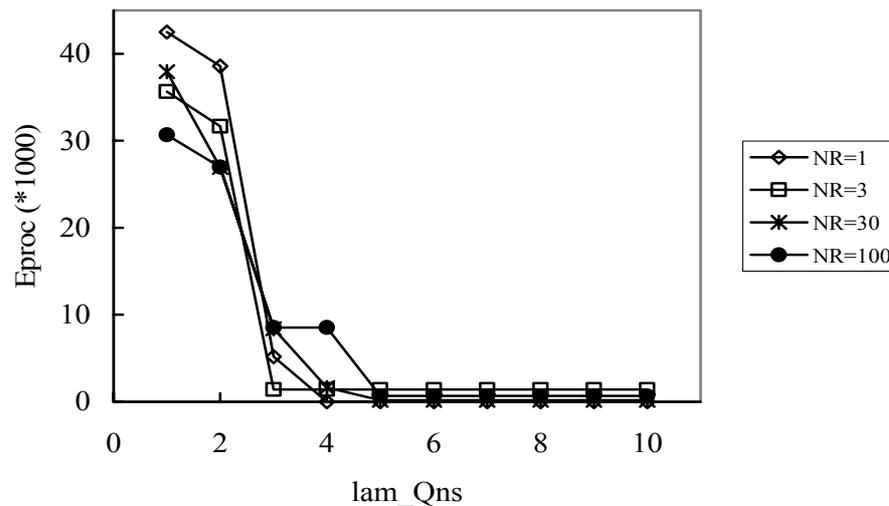
**Figura 5.15** Variação dos custos robustos em função de  $\lambda qns$  ( $lam\_Qns$ ), para as diversas instâncias amostrais

Da observação da *Figura 5.15*, verifica-se que os valores do custo robusto ( $VLArob$ ,  $\psi$ ) variam significativamente na gama de  $\lambda qns$  desde 1 a 4, a qual se apresenta como a zona mais sensível e na qual iremos atentar.

O custo de investimento determinístico ( $NR = 1$ ), considerando a devida penalização da procura não-satisfeita, verifica-se ser significativamente superior (desde 5 até 10%) aos restantes custos robustos, na gama de  $\lambda qns$  apresentada. Entende-se que este resultado é consequência da disseminação do efeito penalizador da procura não-satisfeita de um dado cenário pelos restantes cenários, o que não pode acontecer quando o cenário é único e apresenta probabilidade unitária.

Embora não representada nesta figura, reporta-se que quando  $\lambda qns$  aumenta na gama de 20 até 40, os custos robustos ( $NR > 1$ ) tendem para o valor do custo determinístico: os valores esperados da procura não-satisfeita,  $Eproc$ , já são diminutos naquela gama, logo só após um forte incremento de  $\lambda qns$  é que a parcela respeitante (a penalizar) apresenta valor significativo.

Na *Figura 5.16* representa-se o valor esperado da procura não-satisfeita,  $Eproc$ , versus o parâmetro de penalização associada,  $\lambda qns$ , verificando-se uma pronta diminuição daquele estimador.



**Figura 5.16** Variação do valor esperado da procura não-satisfeita,  $E_{proc}$ , em função de  $\lambda_{qns}$ , para as diversas instâncias amostrais

Registe-se que, para  $\lambda_{qns} = 0$ , o valor esperado em  $E_{proc}$  (valor não representado) é o resultante da configuração mínima: um só processo em cada estágio e com dimensão mínima, à qual corresponderá um valor virtual de procura não-satisfeita elevado (cerca de  $83 \times 10^3$  kg). Para o primeiro valor representado graficamente,  $\lambda_{qns} = 1$ ,  $E_{proc}$  varia entre  $30 \times 10^3$  kg e  $40 \times 10^3$  kg, representando cerca de 23% da procura média ( $150 \times 10^3$  kg). Também se observa que quando  $\lambda_{qns}$  aumenta,  $E_{proc}$  diminui rapidamente, nomeadamente:

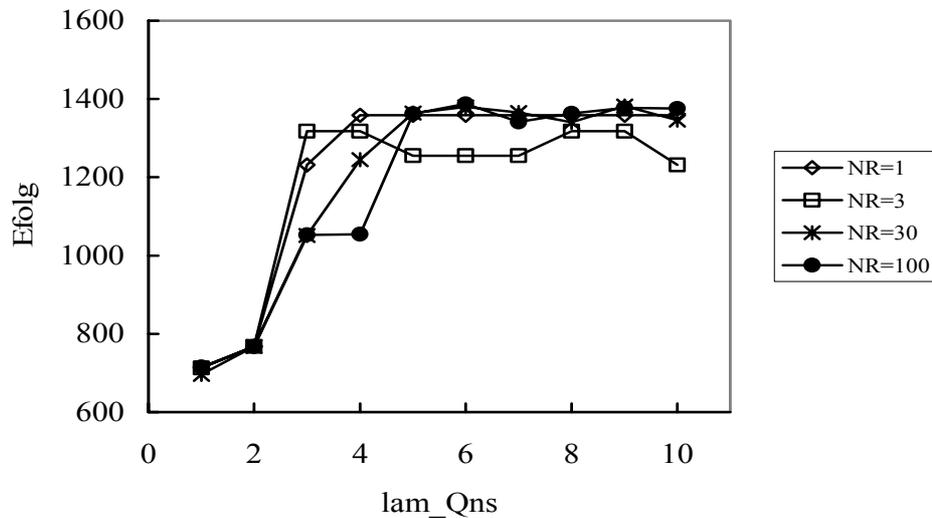
- para  $\lambda_{qns} = 3$ , os valores de  $E_{proc}$  representam menos de 7% do valor médio da procura incerta;
- para  $\lambda_{qns} = 4$ , os valores de  $E_{proc}$  já representam menos de 1% do valor médio da procura, se exceptuarmos  $NR = 100$ ;
- para  $\lambda_{qns} = 5$ , os valores de  $E_{proc}$  representam menos de 1% do valor médio da procura, para todas as instâncias numéricas.

Note-se que a procura não-satisfeita se concentra quase na totalidade em situações bem específicas, merecedoras de uma análise direccionada: num ou noutro produto, para uma ou outra instância. De facto, da observação dos valores óptimos para as variáveis  $Q_{ns,j}$  ressalta que estas variáveis são nulas em quase todas as instâncias e para quase todos os produtos. Será então curial verificar se a não-satisfação da procura de um específico produto terá consequências de outra índole (cumprimento de acordos comerciais, serviço ao cliente, etc.). Neste caso, poderão ser introduzidos parâmetros de penalização específicos para o produto em análise.

A Figura 5.17 relaciona a variação da esperança das folgas de capacidade,  $E_{folg}$ , com o aumento da penalização pela procura não-satisfeita,  $\lambda_{qns}$ , realçando-se um crescimento inicial, a que se segue uma tendência de estabilização na vizinhança de um valor majorante.

Observa-se que  $E_{folg}$  tende para um valor limite de majoração de cerca de 1400 litros, supondo configuração de volumetria óptima máxima, o que representa

um subutilização percentual (agregando as capacidades discretas de todos os estágios, obtêm-se 10929 litros) de até 13%.



**Figura 5.17** Variação do valor esperado das folgas de capacidades,  $E_{folg}$ , em função de  $\lambda_{qns}$ , para as diversas instâncias amostrais

Porém,  $E_{folg}$  representa um valor de carácter relativo, pois é dependente da instância que se analisa, mas se atendermos à sua percentagem em função da capacidade total em cada instância, observam-se (conforme Tabela 5.4, para  $NR = 7$ ) valores estabilizados na gama de 11 a 13 %. De facto, para  $\lambda_{qns} = 0$ , associa-se a configuração mínima de cerca de 3451 litros e na qual se verifica uma subutilização percentual (agregada para todos os estágios) de 6.6%; para os primeiros valores representados graficamente ( $\lambda_{qns} = 1$ ),  $E_{folg}$  é cerca de 712 litros, representando 6.5% de subutilização da capacidade máxima (10929), mas correspondendo a 11.7% da capacidade óptima (6087 litros) para esta instância (Tabela 5.3 e Tabela 5.4).

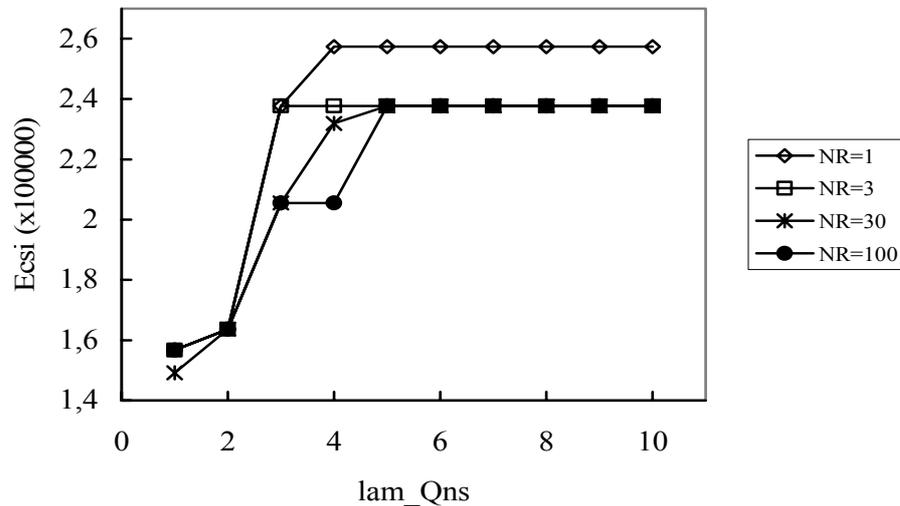
Por conseguinte, quando  $\lambda_{qns}$  aumenta,  $E_{folg}$  aumenta rapidamente para o majorante atrás referido (cerca de 1400 litros), mas as percentagens de subutilização da capacidade em cada instância mantêm-se sensivelmente entre 11 e 13%.

A variação do custo não-robusto,  $E_{csi}$ , em termos da variação do parâmetro penalizador da procura não-satisfeita,  $\lambda_{qns}$ , é representada na Figura 5.18, obtendo-se uma variação similar à da esperança das folgas de capacidade,  $E_{folg}$ : um crescimento inicial forte, que depois tende a estabilizar num valor majorante.

Esse majorante do estimador  $E_{csi}$  correspondente ao custo de investimento determinístico ( $NR = 1$ ), um valor de cerca de  $2.6 \times 10^5$  (euros), significativamente superior (cerca de 8%) aos restantes, cerca de  $2.4 \times 10^5$  (euros), na gama de  $\lambda_{qns}$  apresentada. Quando  $\lambda_{qns}$  aumenta (desde cerca de 20 e até 40), os valores do custo de investimento estocásticos ( $NR > 1$ ) tendem para o valor determinístico (Figura 5.19): o valor de  $E_{proc}$  é diminuto nessa gama, logo só após um forte incremento de  $\lambda_{qns}$  é que a parcela respectiva apresenta valor significativo; no entanto, para valores de  $\lambda_{qns}$  superiores a 3, a diminuição de  $E_{proc}$  é pouco significativa

Repare-se que os custos de investimento, dados por  $E_{csi}$ , tendem para um valor limite de majoração, de cerca de  $2.6 \times 10^5$  (euros), correspondente à satisfação integral de todas as instâncias da procura incerta. Todavia,  $E_{csi}$  apresenta um valor “permanente” (para  $\lambda_{qns} = 0$ , configuração mínima) de cerca de  $1.25 \times 10^5$  (euros), o

qual representa uma percentagem de 48%, relativamente à configuração óptima máxima; para os primeiros valores ( $\lambda qns = 1$  ou 2) representados graficamente,  $Ecsi$  é cerca de  $1.6 \times 10^5$  (euros), representando cerca de 62% do custo da configuração máxima, isto é, uma tal configuração conduziria a uma diminuição do custo de investimento esperado de cerca de 38%.



**Figura 5.18** Variação do valor esperado dos custos de investimento,  $Ecsi$ , em função de  $\lambda qns$ , para as diversas instâncias amostrais

Quando  $\lambda qns$  aumenta, o custo esperado de investimento  $Ecsi$  aumenta rapidamente para o valor atrás referido de  $2.4 \times 10^5$ , pelo que:

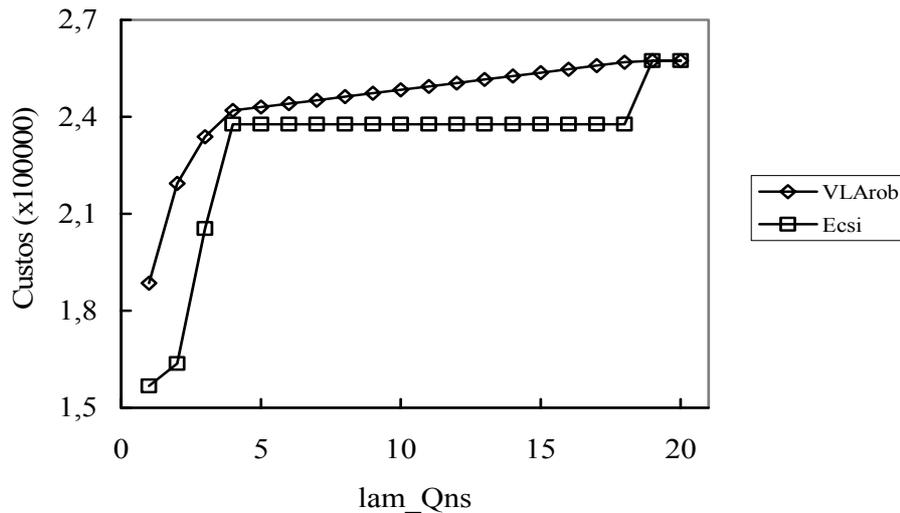
- para  $\lambda qns = 3$ , os valores de  $Ecsi$  (exceptuando  $NR = 3$ ) são cerca de  $2.0 \times 10^5$ , sendo uma diminuição percentual do custo esperado de cerca de 23%;
- para  $\lambda qns = 4$ , os valores de  $Ecsi$  (se exceptuarmos  $NR = 100$ ) são cerca de  $2.4 \times 10^5$ , representando uma diminuição percentual do custo esperado de cerca de 8%;
- para  $\lambda qns = 5$ , todos os valores de  $Ecsi$  coincidem no valor referido anteriormente, conducente à diminuição descrita de cerca de 8%.

Na gama de variação do  $\lambda qns$  de 0 a 20, compara-se a variação do custo robusto ( $VLArob$ ,  $\psi$ ) com a do custo não-robusto ( $Ecsi$ ) na Figura 5.19 para as instâncias com 15 cenários (EX1D), sendo esta variação perfeitamente similar às das outras instâncias tratadas.

Recorde-se que a diferença entre estes dois estimadores decorre da consideração (ou não) da penalidade sobre as procuras não-satisfeitas. Também, o custo não-robusto  $Ecsi$  corresponde neste exemplo *spbatch\_ex1* ao custo de investimento, associado às dimensões discretas seleccionadas.

Numa fase inicial de crescimento da penalização, ambos os custos aumentam fortemente, fruto do aumento de dimensões destinado a diminuir a procura não-satisfeita. Embora o custo robusto,  $VLArob$ , seja sempre superior ao custo não-robusto,  $Ecsi$ , sensivelmente na gama entre valores de  $\lambda qns$  de 5 a 18, essa diferença

deve-se apenas ao aumento deste factor de penalização, pois a dimensão seleccionada permanece imutável, as quantidades produzidas e as procuras não-satisfeitas idem, donde se deduz esse aumento da penalização como o causador do crescimento do custo robusto. Apenas quando esta penalidade ronda o valor de 19, é que a parcela correspondente adquire ponderação suficiente para implicar um novo aumento da dimensão, obviamente porque tal penalização ultrapasse o custo inerente ao aumento de dimensão que aí se verifica. A partir desse valor, a procura encontra-se totalmente satisfeita, *Eproc* tende a anular-se, e os custos robusto e não-robusto coincidem.



**Figura 5.19** Variação dos custos robusto (*VLArób*) e não-robusto (*Ecsi*) em função de  $\lambda_{qns}$  (para  $NR = 15$ )

Nas tabelas seguintes (*Tabela 5.3* e *Tabela 5.4*), registam-se os valores significativos (isto é, associados a alterações na configuração ou solução óptima) para instâncias de EX1C, no qual se consideram instâncias com sete cenários, mas cuja variação com o crescimento da penalização sobre a procura não-satisfeita,  $\lambda_{qns}$ , é similar para todas as outras instâncias, com diferentes números de cenários.

Na *Tabela 5.3* reportam-se os valores esperados (custos robustos, custos não-robustos, *Edsvn*, *Eproc*, *Efolg*) de interesse, comprovando-se numericamente a tipologia das evoluções observadas nos gráficos anteriores:

- os custos, robusto e não-robusto, aumentam fortemente para valores diminutos de  $\lambda_{qns}$ , quando ainda ocorre procura não-satisfeita com alguma importância; esta importância vai diminuindo, pelo que os valores dos custos tendem a variar menos, até que por fim coincidem quando toda a procura é satisfeita;
- o valor esperado da procura não-satisfeita, *Eproc*, diminui fortemente sob influência do respeitante parâmetro de penalização, com tendência para se anular;
- o valor esperado da folga (global) de capacidades, *Efolg*, aumenta significativamente na fase inicial do crescimento de  $\lambda_{qns}$ , tendendo a estabilizar na vizinhança de um valor majorante; mesmo perante uma

configuração mínima (um só processo por estágio, com a menor dimensão possível), regista-se um valor residual, de carácter “permanente”, para a folga de capacidade, de cerca de 725 litros.

**Tabela 5.3** Valores significativos da optimização estocástica do problema-exemplo *spbatch\_ex1* (para  $NR = 7$ )

$\lambda_{qns}$	<i>Custos (rob)</i>	<i>Custos</i>	<i>Edsvn</i>	<i>Eproc</i>	<i>Efolg</i>
0.	124596.08	124596.08	0.0	82855.70	725.64
1.	188885.36	156737.60	0.0	32147.76	712.72
2.	220447.78	163661.91	0.0	28392.93	767.74
3.	234553.01	205467.00	0.0	9695.34	1042.93
4.	241546.56	237689.36	0.0	964.30	1404.58
5.	242510.86	237689.36	0.0	964.30	1233.89
	(...)		(...)		(...)
10.	247332.35	237689.36	0.0	964.30	1359.57
	(...)		(...)		(...)
20.	256975.34	237689.36	0.0	964.30	1305.95
21.	257332.14	257332.14	0.0	0.0	1448.17
	(...)		(...)		(...)
40.	257332.14	257332.14	0.0	0.0	1400.87

Quanto ao valor esperado para os desvios por variabilidade, *Edsvn*, comprova-se numericamente a previsão de índole teórica efectuada (valores de retorno nulos,  $ret_{jtr} = 0$ , implica que o VLA a maximizar corresponda à minimização dos custos de investimento, que dependem apenas das variáveis binárias). Tal ordem de resultados não comprova a robustez na solução, mas neste caso constata-se a respectiva invariabilidade: estando definidas as dimensões dos processos em primeira fase, a produção será sempre a maior possível, dado que se penaliza a procura não-satisfeita, o que equivale a promover a produção.

Porém, a pronta variação nos valores esperados da procura não-satisfeita, *Eproc*, com o aumento do respectivo parâmetro de penalização, originando mesmo que este estimador se anule, permite-nos afirmar a robustez da modelação efectuada.

Quanto à folga de capacidades, *Efolg*, deve notar-se que este estimador contém um carácter de permanência, pois a subutilização dos equipamentos encontra-se subjacente a este tipo de problemática: a subutilização ou a folga não existiria, no âmbito da política de espera nula (ZW), se e só se todos os produtos apresentassem iguais valores para os parâmetros técnicos (concretamente, para os factores de dimensão,  $S_{ij}$ , em litros/kg), nos diferentes estágios.

Complementa-se esta informação numérica sobre os valores esperados, com os valores inscritos na *Tabela 5.4*, referenciados pelos parâmetros de penalização  $\lambda_{qns}$  e pelos custos (não-robustos), discriminando:

- a ordem da dimensão discreta (*size*, *s*) seleccionada em cada estágio, *Ord(s)*; por exemplo, “1/ 3/ 4” indica que foi seleccionada a primeira dimensão no primeiro estágio, a terceira dimensão no segundo estágio, a quarta dimensão no terceiro estágio; estes valores decorrem directamente da solução binária;
- a soma das dimensões ou volumes discretos seleccionados, *Soma(dv)*; este valor é oportuno para a análise das folgas, dado o interesse de estimar *Efolg* em termos relativos a cada instância;
- a percentagem do valor esperado da procura não-satisfeita, *%Eproc*, tendo como base 100 o valor esperado da procura incerta, em cada instância;
- a percentagem do valor esperado da folga de capacidades, *%Efolg*, sendo a base de cálculo deste estimador a soma de volumes, *Soma(dv)*, em cada instância; assim, é possível referenciar de forma adequada e comparável este estimador.

Assim, na *Tabela 5.4*, observa-se o aumento gradual dos custos, associados ao aumento correspondente das ordens de dimensão, *Ord(s)*, e da soma de volumes, *Soma(dv)*.

**Tabela 5.4** Valores significativos da optimização estocástica do problema-exemplo *spbatch\_ex1* (para  $NR = 7$ )

$\lambda qns$	<i>Custos</i>	<i>Ord(s)</i>	<i>Soma(dv)</i>	<i>%Efolg</i>	<i>%Eproc</i>
0	124596.08	1/ 1/ 1	3451.47	21.0	54.1
1	156737.60	1/ 3/ 3	6087.68	11.7	21.0
2	163661.91	1/ 3/ 4	6659.83	11.5	18.6
3	205467.00	2/ 5/ 5	9366.59	11.1	6.3
4	237689.36	3/ 5/ 5	10298.62	13.6	0.6
5	237689.36	3/ 5/ 5	10298.62	12.0	0.6
	(...)		(...)		(...)
10	237689.36	3/ 5/ 5	10298.62	13.2	0.6
			(...)		(...)
20	237689.36	3/ 5/ 5	10298.62	12.7	0.6
21	257332.14	4/ 5/ 5	10929.24	13.3	0.0
	(...)		(...)		(...)
40	257332.14	4/ 5/ 5	10929.24	12.8	0.0

No entanto, em termos de percentagem do valor esperado da folga de capacidades, *%Efolg*, verifica-se que esta se mantém numa gama estrita de valores, entre 11% e 13%. Repare-se que, em termos absolutos, o valor esperado para a folga,

*Efolg*, aumentava bastante para os primeiros valores da penalidade,  $\lambda qns$ , mas depois atingia um valor de estabilidade. Tal permite-nos confirmar a inerência de um valor residual para as folgas de capacidade neste tipo de problema, com diversos produtos a serem processados num mesmo equipamento.

Atendendo ao efeito da penalização da procura não-satisfeita,  $\lambda qns$ , sobre o respectivo valor esperado,  $\%Eproc$ , constata-se que:

- se a penalização for muito elevada, toda a procura será satisfeita,  $\%Eproc$  (bem como  $Eproc$ ) tende a anular-se; para tal, as dimensões seleccionadas e os custos correspondentes também serão elevados; aqui, poderá suceder a situação indesejada de um cenário, mesmo com diminuta probabilidade de ocorrência, se tornar crítico para a definição da solução óptima;
- caso se relaxasse ligeiramente a imposição sobre a satisfação das procuras, por exemplo, até 1%, seriam diminuídos os custos de investimento em cerca de 8%; caso se optasse por uma solução mais flexível, relaxando a satisfação das procuras até 6%, a diminuição dos custos seria de 20%;
- estes argumentos tomam ainda maior preponderância caso se compare esta solução, em ambiente de SPC, com uma política de operação em MPC; como se observou na Sec. 4.2 e no âmbito de *single machine*, o dimensionamento via SPC traduz-se num sobredimensionamento relativamente àquele que seria necessário caso se operasse em MPC.

Juntamente com estas conclusões parcelares, de diminuição significativa em termos de custos perante uma ligeira relaxação das imposições sobre as procuras incertas, entende-se realçar a similitude de comportamento das instâncias tratadas perante diferentes número de cenários,  $NR$  (de 1 a 100), bem como a robustez da modelação efectuada, com os estimadores a responderem de forma adequada à variação do parâmetro de penalização da procura não-satisfeita,  $\lambda qns$ .

### 5.2.2 Aplicação numérica: robustez e períodos múltiplos (*sbatch\_ex2*)

Trata-se a situação generalizada do dimensionamento e sequenciamento dos processos descontínuos, visando não só o objectivo parcial da satisfação da procura incerta dos diferentes componentes, mas antes a usual maximização do valor líquido actual, VLA, considerando vários períodos temporais (problema “dinâmico” e estocástico). A procura incerta é definida por um conjunto discreto de cenários, com uma dada probabilidade, sendo penalizados: os desvios por variabilidade do VLA; as folgas por excesso de capacidade.

Quanto à penalização da procura não-satisfeita (conforme Apêndice 5-D), e atendendo a que os parâmetros aleatórios  $Q_{jtr}$  não são passíveis de optimização, pois são dados do problema sob incerteza, verifica-se que a penalização da procura não-satisfeita se converte num aumento efectivo dos valores de retorno  $ret_{jtr}$  envolvidos. Donde, optou-se por não penalizar inicialmente a procura não-satisfeita neste exemplo *sbatch\_ex2*, especificando  $\lambda qns = 0$ , embora se avalie a respeitante evolução no decorrer das diversas instâncias numéricas. Todavia, dada a importância deste parâmetro de penalização sobre a selecção da configuração óptima, a sua introdução será tratada oportunamente.

Os dados referentes às diversas instâncias deste exemplo numérico, *sbatch\_ex2*, são descritos de forma compreensiva no Apêndice 5-E, sendo

apresentados na *Tabela 5.5* os números de parâmetros, de variáveis binárias e contínuas, e de restrições, bem como os tempos médios (referentes a  $\lambda qns = 0$ ) para a resolução óptima, isto para cada instância analisada.

**Tabela 5.5** Parâmetros, variáveis e restrições das instâncias para o exemplo numérico *spbatch\_ex2*, nas diferentes configurações e com  $\lambda qns = 0$ .

Exemplos numéricos	Parâmetros <i>NC/M/NS/NP/NT/NR</i>	Variáveis binárias	Variáveis contínuas	Restrições	Tempos (s)
EX2A	4/ 3/ 5/ 3/ 5/ 5	45	4965	5538	65.5
EX2B	4/ 3/ 5/ 3/ 10/ 5	45	9925	11063	162.3
EX2C	7/ 5/ 6/ 3/ 5/ 5	90	16630	18590	(50000)

Repare-se que a dimensão do problema, em termos de variáveis contínuas e de restrições, cresce de forma aproximadamente linear com o número de períodos, *NT*, tal como já se tinha constatado para o número de cenários, *NR*, em *spbatch\_ex1*. No entanto, o tempo médio de resolução revela-se perfeitamente exponencial, com tempos razoáveis para as instâncias menores, mas já não sendo possível obter a solução óptima para a generalidade das instâncias de EX2C, pelo que se apresenta o valor médio respeitante à paragem do processo de cálculo entre parênteses. Tal paragem deve-se à parametrização interna do *solver* utilizado (OSL), a qual não foi possível manipular. Também se testaram instâncias ainda de maior dimensão, mas através de uma abordagem enumerativa, não se obtendo resultados devido ao carácter esparso do modelo, o qual apresenta uma fatia significativa de coeficientes e variáveis com valor nulo.

Na implementação deste segundo exemplo numérico, *spbatch\_ex2*, referente ao problema estocástico e multiperíodo, decidiu utilizar-se um número específico de cenários discretos ( $NR = 5$ ) para as diferentes instâncias. Tal decisão fundamenta-se na análise dos resultados obtidos para o exemplo anterior, *spbatch\_ex1*, bem como se comprova através da comparação entre duas instâncias do exemplo EX2A: uma, com 5 cenários; outra, com 10 cenários.

Então, verificou-se que a configuração do sistema de processos descontínuos, definida pela solução binária óptima, se manteve inalterável (ordem da dimensão *Ord(s)* em cada um dos três estágios: 4/ 3/ 4) em face da alteração no número de cenários, para a gama de valores dos parâmetros de penalização testada; pese embora uma ligeira melhoria nos valores esperados quando o número de cenários duplicou, tal melhoria não teve qualquer efeito sobre a descrita configuração e consumiu cerca do triplo do tempo de cálculo (o tempo médio de cerca de 60 segundos por instância, passou para cerca de 180 segundos).

Também se tinha verificado, no decorrer das diversas instâncias para o exemplo numérico *spbatch\_ex1*, onde se tratava o caso estocástico mas uniperíodo ( $NT = 1$ ), que o aumento do número de cenários discretos considerados no cálculo não conduzia a qualquer diferença significativa em termos de configuração óptima das dimensões nos diferentes estágios, embora pudessem deslocar ligeiramente cada configuração no âmbito da gama de variação da penalidade por procura não-satisfeita,  $\lambda qns$ . No entanto, os valores esperados para os estimadores técnicos (de custos, desvios por variabilidade, procuras não-satisfeitas, folgas de capacidades) são similares perante cada configuração seleccionada.

Contudo, a especificação do número de períodos  $NT$  ainda não foi esclarecida, pelo que se procedeu a uma análise comparativa de instâncias de EX2A e EX2B, de forma similar à realizada para o número de cenários: instâncias com 5 períodos temporais *versus* instâncias com 10 períodos, sendo que EX2A e EX2B coincidem nos números dos restantes índices (de estágios,  $M$ , produtos,  $NC$ , dimensões discretas,  $NS$ , limite de processos,  $NP$ , de cenários,  $NR$ ).

A utilização de um número menor de períodos de tempo, sendo uma atitude comum a agregação de períodos temporais de um horizonte alargado, tem vantagens não só em termos de resolução numérica, mas também em termos de uniformização da importância relativa dos diferentes períodos, especialmente se a taxa de actualização é elevada. Mesmo considerando períodos temporais com igual dimensão (seja, um ano), a indicação inequivocamente positiva dos estimadores económicos para um horizonte de médio prazo pode traduzir-se na suficiência da tomada de decisão. Caso uma abordagem inicial, de médio prazo, não seja conclusiva, então é conveniente considerar prazos mais longos, mas estando subjacente tanto o aumento dos riscos associados, como garantir os pressupostos por um horizonte mais alargado.

Na *Tabela 5.6* apresentam-se os valores dos estimadores usuais na presente modelação, para instâncias de EX2A, especificadas com 5 períodos de tempo. Similarmente, na *Tabela 5.7* reportam-se os valores respeitantes a instâncias EX2B, com 10 períodos temporais, procedendo-se à respectiva comparação.

**Tabela 5.6** Valores dos estimadores para o problema *spbatch\_ex2*, considerando instâncias de EX2A com  $\lambda qns = 0$ .

$\lambda dsv / \lambda flg$	$VLArob$	$Ecsi$	$Edsvn$	$Eproc$	$Efolg$
0.0/ 0.0	19235.08	19235.08	5102.04	10994.32	624.45
0.1/ 0.005	18159.31	19235.08	5102.04	10994.32	607.58
0.2/ 0.01	17083.54	19235.08	5102.04	10994.32	607.58

A definição da gama de variação dos parâmetros de penalização decorreu de forma a que o respectivo efeito conjugado provocasse uma diminuição de cerca de 10% no valor objectivo robusto ( $VLArob$ ), sendo avaliada a evolução dos correspondentes estimadores técnicos. Observou-se, de acordo com a *Tabela 5.6*, que os referidos estimadores se mantêm invariáveis ( $Ecsi$ ,  $Edsvn$ ,  $Eproc$ ), ou variam apenas ligeiramente ( $Efolg$ ). Tal decorre da invariabilidade da solução binária óptima, com  $Ord(s) = (4/ 3/ 3)$  para todas as instâncias testadas, sendo óptimo produzir a maior quantidade possível (igual) nessa configuração (igual), obtendo-se valores esperados também semelhantes.

Para as instâncias de EX2B reportadas na *Tabela 5.7*, foi seguida a mesma metodologia que para EX2A, verificando-se igual estabilidade nos estimadores. Todavia, aqui a configuração inalterável em todas as instâncias é ligeiramente superior, com  $Ord(s) = (4/ 3/ 4)$ , encarando-se então a alternativa de seleccionar uma dimensão superior para o terceiro estágio.

**Tabela 5.7** Valores dos estimadores para o problema *spbatch\_ex2*, considerando instâncias de EX2B com  $\lambda_{qns} = 0$ .

$\lambda_{dsv}/\lambda_{flg}$	<i>VLArób</i>	<i>Ecsi</i>	<i>Edsvn</i>	<i>Eproc</i>	<i>Efolg</i>
0.0/ 0.0	144795.64	144795.64	8668.59	3682.19	572.84
0.5/ 0.05	135602.69	144795.64	8668.59	3682.19	520.88
1.0/ 0.1	126409.74	144795.64	8668.59	3682.19	520.88

Relativamente às instâncias com apenas 5 períodos de tempo, verifica-se na *Tabela 5.7*: o aumento em uma ordem de grandeza dos valores esperados do VLA, robusto (*VLArób*) e não-robusto (*Ecsi*), deve-se a que os custos de investimento ocorrem apenas no período inicial, correspondendo os períodos adicionais, de 6 a 10, apenas a retornos sem investimentos complementares; a variabilidade do VLA, estimada por *Edsvn*, embora aumente em termos absolutos, revela-se até diminuta se a analisarmos em termos relativos ou percentuais; dado o aumento de capacidade proposto (da dimensão de ordem 3 para a de ordem 4, no terceiro estágio), o valor esperado para a procura não-satisfeita, *Eproc*, diminui significativamente, bem como diminui ligeiramente o valor esperado das folgas de capacidade, *Efolg*.

Esta diminuição do valor esperado das folgas, *Efolg*, apresenta uma interpretação dupla: é importante, pois esta diminuição ocorre num contexto de aumento de capacidade; é pouco significativa, em termos relativos ou percentuais, caso se considere o total das capacidades seleccionadas.

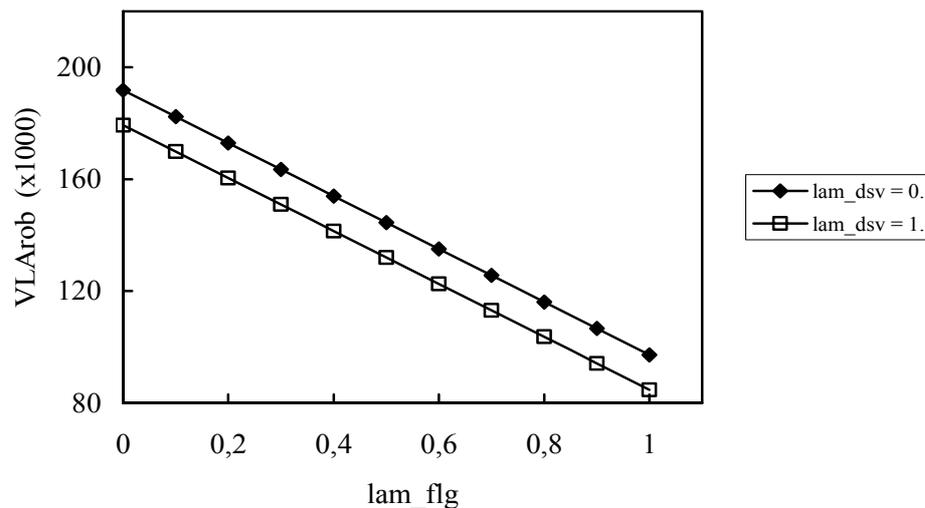
É conveniente mencionar a situação de não coincidência dos valores incertos nas diferentes instâncias para os períodos directamente comparáveis (os primeiros cinco períodos), devido à forma como é realizada a geração aleatória de valores. Isto é, embora os parâmetros das distribuições de probabilidade, como a média e o desvio padrão para distribuição gaussiana, apresentem os mesmos valores, não é exacto afirmar a que o exemplo EX2B, com 10 períodos temporais, corresponde integralmente ao EX2A, com apenas 5 períodos, aos quais se acrescentam 5 períodos complementares.

Nas instâncias anteriormente comparadas, observou-se que a variação nos parâmetros de penalização,  $\lambda_{dsv}$  e  $\lambda_{flg}$ , não provocava qualquer efeito significativo, nomeadamente, alterando a configuração óptima. Ora, a gama de variação das penalidades foi burilada tendo em vista apenas um efeito conjugado de cerca de 10% sobre o valor objectivo robusto, pelo que se obtém uma gama de valores relativamente diminuta. Para comprovar que a limitação dos efeitos não resulta da utilização de tal gama, alargou-se esta gama para o intervalo [0,1] em cada um dos parâmetros testados, de forma a ser observada (em *Tabela 5.8*) uma diminuição da ordem dos 50% no valor robusto, *VLArób*. Esta diminuição no valor robusto não se reflecte sobre os outros estimadores, nomeadamente, sobre o valor não-robusto (*Ecsi*) que reflecte directamente os custos de investimento, inalteráveis nestas instâncias.

**Tabela 5.8** Valores dos estimadores para o problema *spbatch\_ex2*, considerando instâncias de EX2A

$\lambda_{dsv}/\lambda_{flg}$	<i>VLArob</i>	<i>Ecsi</i>	<i>Edsvn</i>	<i>Eproc</i>	<i>Efolg</i>
0.0/ 0.0	191835.82	191835.82	12497.19	3163.86	603.54
1.0/ 0.0	179338.63	191835.82	12497.19	3163.86	605.99
0.0/ 0.5	144527.63	191835.82	12497.19	3163.86	520.88
1.0/ 0.5	132030.44	191835.82	12497.19	3163.86	520.88
0.0/ 1.0	97219.44	191835.82	12497.19	3163.86	520.88
1.0/ 1.0	84737.54	191619.67	12381.91	3290.84	520.88

Assim, a configuração das dimensões óptimas (4/ 3/ 4) manteve-se inalterável em todas as instâncias testadas, verificando-se que a diminuição do valor robusto (*VLArob*) é linear na gama considerada (*Figura 5.20*), dependendo apenas do parâmetro de penalização das folgas de capacidade,  $\lambda_{flg}$  (ou *lam\_flg*): o valor esperado das folgas *Efolg* não varia na gama considerada, se exceptuarmos os valores correspondentes ao parâmetro nulo (neste caso, as folgas não são penalizadas). Para as restantes instâncias, a comparação de valores da *Tabela 5.8* pode explicar-se: a variação (cerca de 12500) entre pares de valores consecutivos do *VLArob*, resulta directamente da incorporação da penalização do *Edsvn*; todos os valores dos estimadores (*Ecsi*, *Edsvn*, *Eproc*, *Efolg*) permanecem constantes, exceptuando a última instância onde ocorre uma ligeira adaptação dos fluxos de produção (número e dimensão dos *batches*), mas sem provocar uma alteração nas dimensões para os diferentes processos.



**Figura 5.20** Variação do valor objectivo robusto, *VLArob*, para a gama de valores testados para  $\lambda_{flg}$  (EX2A)

Dado que interessa analisar os valores óptimos associados a diferentes configurações, de forma a tornar possível uma comparação entre configurações

alternativas, optou-se pela introdução de um valor não nulo para penalizar a procura não-satisfeita, bem como proceder à variação do referido parâmetro numa dada gama de valores. Tal variação tem um efeito de promoção das quantidades produzidas (Apêndice 5-D), pelo que se prevê alteração das configurações no decorrer dessa gama, sendo possível estabelecer o interesse relativo dessas configurações.

Nessa análise, para cada configuração alternativa, são avaliados os parâmetros esperados usuais (*VLA<sub>rob</sub>*, *Ecsi*, *Edsvn*, *%Eproc*, *%Efolg*), bem como interessa também comparar outros estimadores, ou critérios de índole económica para além do VLA não-robusto (*Ecsi*), como referem Abecassis e Cabral [1991] ou Barros [1991]:

- a Taxa Interna de Rendibilidade, TIR, corresponde ao valor da taxa de actualização que anula o VLA, onde se igualam os fluxos monetários de retorno (exploração) e de investimento; corresponde à taxa que se obtém sobre os capitais que permanecem investidos, sendo o investimento inicial recuperado sucessivamente;
- o tempo necessário para o retorno do investimento, *payback* (também Período de Recuperação, PR, ou *Return Of Investment*, ROI), é usualmente dirigido a investimentos em ambiente de risco, ou com diminuto horizonte temporal; este critério é especialmente ponderado no âmbito de equipamentos com períodos de obsolescência curtos, ou de indústrias com forte inovação tecnológica;
- a razão (em percentagem) entre benefícios e custos, *%Benef*, relacionando os fluxos actualizados e líquidos dos retornos com os custos de investimento; comparando duas configurações alternativas e seguindo este critério, é preferível a de maior razão, sendo o respectivo valor igual ou superior a 100.

Então, procedeu-se a uma variação conjunta do trio de parâmetros de penalização para o EX2A, encontrando-se uma listagem representativa de resultados no Apêndice 5-E, mas apresentam-se os que se entendem como mais significativos (isto é, associados a alterações) na *Tabela 5.9* e na *Tabela 5.10*. De realçar aqui dois aspectos: o tempo de resolução médio, fruto do aumento dos diversos parâmetros de penalização, aumenta para perto do triplo (172 segundos) daquele referenciado anteriormente (60 segundos), com  $\lambda_{flg} = 0$ .; considerou-se um valor médio para o retorno unitário (0.08) sensivelmente metade do valor médio anterior (0.15), no sentido de que a penalização por procuras não-satisfeitas apresente um efeito propagador sobre a produção; assim, os valores esperados para os estimadores associados ver-se-ão consideravelmente diminuídos.

Para a gama de valores de  $\lambda_{dsv}$  tratada, de 0.0 a 2.0, não se verificou qualquer variação sensível dos valores esperados, ocorrendo sempre a mesma configuração das dimensões (com iguais custos), à qual se associam os valores máximos possíveis para as produções (iguais retornos). Como já referido, tal aspecto deriva tanto do limitado número de cenários discretos, como da estabilidade da solução binária preconizada, a qual se modificará apenas perante instâncias de tal forma díspares que possam influenciar o valor da função objectivo, obtendo-se uma solução binária alternativa.

**Tabela 5.9** Valores significativos dos estimadores para EX2A, com  $\lambda_{dsv} = 1.$ ,  $\lambda_{flg} = 0.1$ , e variando  $\lambda_{qns}$

$\lambda_{dsv}/\lambda_{flg}/\lambda_{qns}$	<i>VLArob</i>	<i>Ecsi</i>	<i>Edsvn</i>	<i>%Eproc</i>	<i>%Efolg</i>
1./ 0.1/ 0.	13486,67	30240,09	5442,18	7.6	7,7
1./ 0.1/ 2.	6607,68	28879,19	6638,54	2,1	6,2
1./ 0.1/ 6.	-1277,33	24255,41	7351,39	0,4	8,6
1./ 0.1/ 12.	-4668,02	21612,82	7605,91	0,1	9,2
1./ 0.1/ 100.	-22349.74	21612.82	7605.91	0.1	9.2

Considerando a variação de  $\lambda_{flg}$  analisada, de 0.0 a 0.3, observa-se uma diminuição marcada do valor objectivo robusto (*VLArob*), o qual até pode apresentar valores negativos, embora o efeito sobre o VLA esperado (*Ecsi*) seja pouco significativo. Anota-se que ocorre um amortecimento do efeito da penalização da procura não-satisfeita, quantificado através de  $\lambda_{qns}$ , sendo necessário que este parâmetro de penalização apresente um valor mais elevado para se obter uma mesma configuração óptima; isto é, caso a penalização das folgas de capacidade aumente, verifica-se a necessidade de um aumento ao “prémio” de produção (ao penalizar-se a procura não-satisfeita, aumenta-se o factor atractivo da produção, como se mostra em Apêndice 5-D) para que seja seleccionada uma mesma configuração. Realce-se a significância acrescida para o valor esperado das folgas, dado o factor multiplicativo inerente ao número de índices que são considerados (números de estágios  $i$ , de produtos  $j$ , de períodos de tempo  $t$ , de cenários  $r$ ).

**Tabela 5.10** Estimadores económicos referentes a EX2A, para  $\lambda_{dsv} = 1.$ ,  $\lambda_{flg} = 0.1$ , e variando  $\lambda_{qns}$

$\lambda_{qns}$	<i>Ord(s)</i>	<i>Custo</i>	<i>Ecsi</i>	<i>%Benef</i>	<i>payback</i>	<i>TIR</i>
0.	4/ 3/ 3	145840.2	30240.09	120.7	4.0	16.5
2.	4/ 3/ 4	157249.9	28879.19	118.4	4.0	15.5
6.	5/ 4/ 4	164863.4	24255.41	114.7	4.2	14.5
12.	5/ 5/ 4	167983.2	21612.82	112.9	4.3	13.5
100.	5/ 5/ 4	167983.17	21612.82	112.9	4.3	13.5

Em face do aumento de  $\lambda_{qns}$  na gama de valores analisada, especialmente de 0. a 20. (embora para valores superiores a 12. a configuração permaneça invariável), verifica-se que:

- ocorre um gradual aumento da configuração óptima, com o aumento de custos de investimento associado ao aumento sucessivo das dimensões;

- embora a percentagem da procura não-satisfeita (valor esperado),  $\%Eproc$ , possa ser inicialmente significativa (mais de 5%), diminui rapidamente para menos de 1%, o que representa a quase totalidade da procura incerta;
- em sentido inverso, a percentagem da folga de capacidades,  $\%Efolg$ , tende a aumentar; sabendo que o volume total sobre o qual é calculado este estimador também tende a aumentar, bem como o efeito de significância (número de índices, referido anteriormente) associado, então este estimador deve ser criteriosamente analisado;
- o VLA não-robusto,  $Ecsi$ , tende a diminuir gradualmente; deve atender-se aqui à limitação do número de períodos temporais (cinco), pelo que o custo do investimento suportado no período inicial poderá originar um retorno mais elevado, caso se prolongue o limite temporal;
- a percentagem referente à razão entre os retornos e os custos,  $\%Benef$ , tende a diminuir sucessivamente, embora essa diminuição seja parcelarmente pouco relevante; tal diminuição é derivada da proeminência do aumento dos custos totais envolvidos, relativamente à evolução dos retornos correspondentes;
- o tempo de retorno do investimento (*payback*) apresenta um ligeiro aumento, pouco significativo, dada a evolução da importância relativa entre o aumento de custos e de retornos; é curial o facto do tempo de retorno (cerca de quatro períodos) ser enquadrável num horizonte temporal (cinco períodos) algo limitado para este tipo de situação;
- a taxa interna de rentabilidade, TIR, tende a diminuir sucessivamente, dado o superior impacto do aumento de custos relativamente ao aumento dos retornos; realça-se o facto, já referido no âmbito do VLA ( $Ecsi$ ), de o horizonte temporal ser limitado, sendo possível obter valores mais vantajosos se esse horizonte for alargado; ainda, constata-se que os valores estimados para a TIR suplantarem substancialmente a taxa de actualização (9%).

Por conseguinte, a variação dos parâmetros de penalização por procura não-satisfeita,  $\lambda qns$ , permite obter configurações alternativas para as dimensões dos processos, além de que também se adapta a necessidade dessas dimensões com a presença de satisfação das procuras incertas. Esta satisfação das procuras pode ser visualizada como o resultado de uma evolução favorável dos retornos sobre cada produto considerado.

Esta metodologia foi também efectuada com o exemplo EX2B, o qual difere do exemplo EX2A apenas no número de períodos temporais,  $NT$ : dez, em vez de cinco. A evolução dos diversos estimadores, técnicos e económicos, foi similar à descrita para o EX2A, registando-se uma afectação de dimensões, ligeiramente superior mas pouco significativa. Ou seja, a evolução do aumento de dimensões com o aumento da penalização por procura não-satisfeita,  $\lambda qns$ , foi similar mas mais rápida, não sendo necessário favorecer tanto a produção para obter a mesma capacidade instalada. Tal deriva do pressuposto de que as condicionantes incertas do problema (procuras, retornos) se mantêm por mais cinco períodos, pelo que se torna vantajoso um aumento dos investimentos, o qual ocorre apenas no período inicial.

Em termos de estimadores de índole económica, verifica-se um amplo aumento do VLA esperado (*Ecsi*), da TIR associada, ou da percentagem da razão entre benefícios e custos, *%Benef*: estes resultados devem-se ao facto de se considerarem cinco períodos complementares, com os respectivos montantes de retorno, mas sem qualquer custo adicional. Também é de referir que o conceito de VLA considera que todos os montantes recebidos são reinvestidos a uma taxa igual à taxa de actualização aplicada, o que poderá não se verificar no âmbito de um horizonte alargado. Quanto ao *payback*, este estimador apresenta sensivelmente os mesmos valores, da ordem dos quatro períodos temporais, pois as instâncias numéricas são geradas com iguais parâmetros das distribuições de probabilidades.

Da análise comparativa entre as instâncias de EX2A e EX2B constata-se: uma semelhança da evolução dos diversos estimadores perante as gamas dos parâmetros de penalização; a análise dos estimadores de índole económica (*Ecsi*, *%Benef*, *payback*, TIR) é coincidente no interesse do investimento, pois todos aqueles estimadores são francamente positivos, mesmo para um horizonte de cinco períodos temporais; caso assim não fosse, seria conveniente analisar criteriosamente um horizonte mais alargado (por exemplo, com dez períodos), onde os estimadores económicos já poderiam apresentar valores mais vantajosos, mas atendendo ao risco inerente e à manutenção dos pressupostos nesse período.

De seguida, procedeu-se ao estudo da optimalidade das instâncias de EX2C (7/5/6/3/5/5), no qual se considera uma metodologia similar à seguida anteriormente: inicialmente, trata-se o problema anulando o parâmetro de penalização por procura não-satisfeita,  $\lambda_{qns} = 0$ ; então, estuda-se o efeito da variação deste parâmetro de penalização sobre a configuração seleccionada, analisando-se os respeitantes estimadores económicos.

Nestas instâncias de EX2C, especificou-se um valor médio para os retornos unitários de  $ret_{j\pi} = 0.15$ , pelo que os respectivos valores praticamente duplicam os utilizados nos exemplos anteriores (EX2A e EX2B); ainda, o número de produtos (sete) também aumenta quase para o dobro, logo os valores esperados para os montantes globais são também consideravelmente superiores.

Realizou-se uma abordagem inicial com  $\lambda_{qns} = 0$ , estimando uma gama de variação para os restantes dois parâmetros de penalização, dos desvios por variabilidade,  $\lambda_{dsv}$ , e das folgas de capacidade,  $\lambda_{flg}$ , cujo efeito conjunto provocasse uma variação de cerca de 20% do valor objectivo robusto, *VLArob*. Os resultados obtidos constam da *Tabela 5.11*, correspondendo a uma configuração  $Ord(s) = (6/66/6/6/6)$  invariável em todas as instâncias testadas. Note-se nesta configuração que, devido à forma exponencial considerada para os custos dos processos, cujo expoente ronda 0.6, a formulação efectuada tende a privilegiar o menor número de processos e com uma maior dimensão. O facto de no segundo estágio serem seleccionados dois processos com dimensão máxima (6) decorre da dimensão dos *batches* ter um valor elevado, para aproveitamento das dimensões implementadas nos outros estágios, não podendo nesse segundo estágio ser reduzida para uma dimensão menor.

O terceiro valor de *Efolg* (em *itálico* na *Tabela 5.11*) justifica-se atendendo ao facto de que este valor esperado se refere à folga por *batch* produzido, ou seja: cada elemento avaliado através deste estimador resulta da razão entre a totalidade da folga e o número de *batches* (conforme Apêndice 5-D); a variável penalizada na função objectivo robusta é, precisamente, a totalidade da folga e que efectivamente diminui; no entanto, ocorre alguma modificação no sequenciamento da produção,

com diminuição do número de *batches*, provocando um efeito conjugado de aumento, embora ligeiro, do valor esperado das folgas, *Eproc*.

**Tabela 5.11** Valores esperados para o problema *spbatch\_ex2*, considerando instâncias de EX2C com  $\lambda_{qns} = 0$ .

$\lambda_{dsv}/\lambda_{flg}$	<i>VLArób</i>	<i>Ecsi</i>	<i>Edsvn</i>	<i>Eproc</i>	<i>Efolg</i>
0.0/ 0.0	259348.17	259348.17	21833.76	17066.04	1859.34
0.5/ 0.05	238525.30	259348.17	21833.76	17066.04	1859.34
1.0/ 0.1	217727.51	259301.90	21806.00	17324.78	1879.03

Repare-se que o aumento, também ligeiro, do valor esperado da procura não-satisfeita, *Eproc*, nessa mesma terceira instância e relativamente à segunda, é directamente justificado pelo aumento da penalização das folgas,  $\lambda_{flg}$ , conquanto a penalidade pela procura não-satisfeita,  $\lambda_{qns}$ , se mantenha nula.

De registo é o que respeita aos tempos de resolução: cerca de 17500 segundos para a instância com todos os parâmetros de penalização nulos, sendo mais de 30000 segundos para as restantes instâncias, onde esses parâmetros já não são nulos, excepto quando  $\lambda_{qns} = 0$ .

Por conseguinte, para uma variação de  $\lambda_{dsv}$  até 1.0, conjugada com a de  $\lambda_{flg}$  até 0.1, resulta uma variação do valor robusto *VLArób* em cerca de 17%, mas a solução permanece estabilizada, com uma solução binária invariável. Então, pretende-se avaliar o efeito da penalização por procura não-satisfeita,  $\lambda_{qns}$ , sobre alterações da configuração óptima, comparando-se os estimadores económicos respeitantes a cada configuração.

Assim, apresentam-se na *Tabela 5.12* os estimadores referentes à melhor solução inteira obtida em cada instância, posto que foram obtidas sem que o processamento numérico garantisse a respectiva optimalidade (excepto para  $\lambda_{qns} = 0$ , onde se reporta a solução óptima). O tempo de resolução, em todas as instâncias com soluções inteiras subóptimas, ultrapassava já os 50000 segundos, quando a resolução cessava devido ao número máximo de iterações do próprio sistema. Tal interrupção decorria de parametrização interna do *solver* OSL, não passível de manipulação.

Embora não se garanta a solução inteira óptima, para cada instância tratada, a forma de evolução dos diversos estimadores é similar à observada em momentos anteriores, com o aumento sucessivo das dimensões seleccionadas (ver *Tabela 5.13*): o decréscimo inicialmente acentuado do valor robusto, *VLArób*, vai atenuar-se com a diminuição da procura não-satisfeita; o valor não-robusto, *Ecsi*, também tende a diminuir, pois embora o nível global de produção aumente, o crescimento do montante associado não repõe o crescimento de custos; o valor esperado dos desvios de variabilidade, *Edsvn*, permanece estabilizado; a percentagem do valor esperado da procura não-satisfeita, *Eproc*, diminui fortemente com o aumento de dimensão, sendo depois necessário valores bem mais elevados de  $\lambda_{qns}$  para que este estimador se anule; a percentagem do valor esperado das folgas, *Efolg*, mantém-se estável numa gama de 6 a 8%.

**Tabela 5.12** Valores dos estimadores para a melhor solução inteira de EX2C, com  $\lambda_{dsv} = 1.$ ,  $\lambda_{flg} = 0.1$ , e variando  $\lambda_{qns}$

$\lambda_{dsv}/\lambda_{flg}/\lambda_{qns}$	<i>VLArob</i>	<i>Ecsi</i>	<i>Edsvn</i>	<i>%Eproc</i>	<i>%Efolg</i>
1./ 0.1/ 0.	217727.51	259301.90	21806.00	11.4	7.0
1./ 0.1/ 7.	98543.73	259085.06	21934.99	11.2	6.9
1./ 0.1/ 14.	103766.40	177575.18	21430.64	1.6	6.0
1./ 0.1/ 21.	93458.55	148858.35	21403.17	0.1	7.6
1./ 0.1/ 28.	92085.91	148858.35	21403.17	0.1	7.6
1./ 0.1/ 35.	90713.27	148858.35	21403.17	0.1	7.6
1./ 0.1/ 70.	98040.86	141819.05	21135.75	0.0	6.2

Na *Tabela 5.13* registam-se os estimadores de índole económica, também referentes à melhor solução inteira registada no cálculo das instâncias de EX2C. Ao crescimento sucessivo das dimensões seleccionadas, corresponde o aumento dos custos de investimento. Então, atendendo a que o montante global dos retornos já pouco pode crescer, pois a procura já se encontra quase totalmente satisfeita, esses aumentos de dimensões e custos tendem a deteriorar os restantes estimadores económicos (*%Benef*, *payback*, TIR).

No entanto, repare-se que estes estimadores económicos são calculados tendo por base o valor de retorno unitário especificado (0.15), mas em termos virtuais, ocorre uma sobrevalorização da produção correspondente a  $\lambda_{qns}/(NC.NT)$ . Então, para  $\lambda_{qns} = 7$ , com uma tal sobrevalorização virtual unitária de 0.2, o que mais que duplica o valor de retorno unitário especificado, regista-se a mesma configuração de dimensões que para  $\lambda_{qns} = 0$ .

Desta forma, constata-se a robustez da solução apresentada, no contexto dos parâmetros de penalização aplicados, atendendo ao possível aumento dos valores unitários dos retornos (incertos).

Ainda, para a solução óptima obtida para  $\lambda_{qns} = 0$ , recorre-se a uma metodologia enumerativa de todas as configurações previsíveis: por fixação das variáveis binárias (descrita em Sec. 3.2), percorrem-se todas as configurações possíveis, desde apenas um processo por estágio, com dimensão máxima,  $Ord(s) = (6/6/6/6/6)$ , até à configuração com dois processos por estágio, todos com dimensão máxima,  $Ord(s) = (66/66/66/66/66)$ .

O conjunto de soluções ( $7^5 = 16807$ ) assim tratadas permitiu a confirmação da solução óptima, apresentada na primeira linha da *Tabela 5.12*, bem como na da *Tabela 5.13*, embora tal não fosse o propósito.

Os estimadores para as melhores cinco configurações, seleccionadas por ordem decrescente do VLA não-robusto (*Ecsi*), descrevem-se na *Tabela 5.14.*, o que nos permite visionar a evolução das configurações segundo tal critério.

**Tabela 5.13** Estimadores económicos para melhor solução inteira de EX2C, com  $\lambda_{dsv} = 1.$ ,  $\lambda_{flg} = 0.1$ , e com variação de  $\lambda_{qns}$

$\lambda_{qns}$	<i>ord(s)</i>	<i>Custo</i>	<i>Ecsi</i>	<i>%Benef</i>	<i>payback</i>	<i>TIR</i>
0.	6/ 66/ 6/ 6/ 6	312954.93	259301.90	182.9	2.5	37.5
7.	6/ 66/ 6/ 6/ 6	312954.93	259085.06	182.8	2.5	37.5
14.	44/ 44/ 33/ 33/ 44	445147.79	177575.18	139.9	3.4	23.5
21.	44/ 66/ 55/ 55/ 44	483446.83	148858.35	130.8	3.6	20.5
28.	44/ 66/ 55/ 55/ 44	483446.83	148858.35	130.8	3.6	20.5
35.	44/ 66/ 55/ 55/ 44	483446.83	148858.35	130.8	3.6	20.5
70.	55/ 55/ 33/ 55/ 44	491154.66	141819.05	128.9	3.7	19.5

Note-se que, embora seleccionando as configurações pelo valor não-robusto (*Ecsi*), em vez do valor robusto (*VLArob*), a hierarquização realizada conduz à mesma solução óptima, mas tal pode não ser necessariamente verdadeiro. Além disso é providenciada uma série de informação sobre as melhores configurações alternativas, que pode ser aproveitada para a construção de procedimentos heurísticos.

**Tabela 5.14** Enumeração das melhores soluções de EX2C, para:  $\lambda_{dsv} = 1.$ ;  $\lambda_{flg} = 0.1$ ;  $\lambda_{qns} = 0.$

<i>Ord(s)</i>	<i>Ecsi</i>	<i>Custo</i>	<i>%Benef</i>	<i>payback</i>	<i>TIR</i>
6/ 6,6/ 6/ 6/ 6	259301.90	312954.93	182.9	2.5	37.5
6/ 5,5/ 6/ 6/ 6	250556.72	309896.72	180.9	2.5	36.5
6/ 4,4/ 6/ 6/ 6	192758.20	302602.29	163.7	2.8	31.5
6/ 6,6/ 6/ 6/ 2,2	144114.30	326965.58	144.1	3.2	24.5
6/ 5,5/ 6/ 6/ 2,2	141304.49	323907.37	143.6	3.3	24.5

Este segundo exemplo *spbatch\_ex2* complementa o primeiro, *spbatch\_ex1*, em termos da análise multiperíodo associada à definição do VLA. Na sequência de *spbatch\_ex1*, também em *spbatch\_ex2* não se verificou qualquer impacto resultante

do número de cenários,  $NR$ , considerados na modelação. Quanto ao número de períodos temporais,  $NT$ , embora se possam adoptar diferentes políticas de tratamento do risco associado a horizontes alargados, a consideração de um número limitado de períodos tem aplicação directa perante ambiente altamente incertos (obsolescência, inovação tecnológica, ciclo de vida dos produtos). Perante situações problemáticas que requeiram um estudo mais pormenorizado, será considerado um maior número de períodos ou um horizonte mais alargado, tendo o cuidado de garantir a manutenção dos diversos pressupostos ao longo desse horizonte. No caso de se analisar configurações alternativas, ou até para tentar garantir a optimalidade de uma dada configuração, pode recorrer-se a uma metodologia enumerativa caso as instâncias sejam limitadas.

## Sinopse

Neste Capítulo dedicado ao estudo numérico da optimalidade, apresenta-se na primeira secção um modelo para planeamento de processos generalizado, *spplan\_milp\_flex*, o qual conjuga aspectos de produção flexível, seja em termos de matérias-primas ou de produtos finais, com um enquadramento estocástico e robusto, possibilitando incorporar na solução a política face ao risco do decisor. O desenvolvimento do presente modelo resulta tanto do trabalho de implementação efectuado no Capítulo 3, que permitiu uma maior sensibilidade, como do estudo teórico do Capítulo 4, já que se combina: a flexibilidade dos processos; utilização de esquemas de produção; e as características de robustez, sem se abdicar da linearidade no modelo.

Analizou-se a robustez associada à formulação em *spplan\_milp\_flex*, constatando-se que os valores esperados dos desvios, da variabilidade e das folgas de capacidade, se mostram adaptáveis perante os valores atribuídos aos parâmetros de penalização: o modelo e as soluções são robustos. Em particular, verifica-se que o efeito de penalização da variabilidade se encontra amortecido, mas tal deve-se ao limitado número de cenários utilizados nos diversos exemplos numéricos.

Descrevem-se, sucintamente, os exemplos numéricos para diferentes situações problemáticas, onde se desenvolve um estudo sobre vias alternativas de produção, a partir das quais se podem estimar os parâmetros dos processos que as tornam economicamente vantajosas:

- exemplo *spplan\_ex1* — situação simplificada, com apenas dois processos flexíveis onde são processados cinco componentes, e onde se analisa a evolução da variabilidade e das folgas de capacidade com os parâmetros de penalização; assim, obtém-se conhecimento quanto à especificação desses parâmetros, de forma a promover a robustez, tanto do modelo como da solução;
- exemplo *spplan\_ex2* — encara-se a situação onde se necessita de um reagente, de valor económico relativamente elevado e ao qual se associa uma diminuta disponibilidade; então, comparam-se diferentes vias para a produção do reagente, ou mesmo a sua substituição por um outro e utilizando um processo alternativo;

- exemplo *splan\_ex3* — situação onde um produto de interesse económico reduzido (e/ou que necessite de ser transformado) se encontra associado a um produto economicamente vantajoso; são analisadas diferentes alternativas de processamento desse produto não desejado, numa estrutura composta por cinco processos e que considera oito componentes.

Na segunda secção é apresentado um modelo para dimensionamento e sequenciamento de processos descontínuos, *spbatch\_milp\_fmsz*, o qual generaliza o modelo determinístico de *batch\_milp\_fmsz* (*Flowshop/ Multiple machine/ SPC/ Zero Wait*) a um enquadramento estocástico em duas fases. O modelo visa a promoção da robustez, o que permite o tratamento do risco associado a investimentos de médio e longo prazo. O desenvolvimento deste modelo robusto baseia-se no estudo probabilístico realizado no Capítulo 4, que permitiu seleccionar o modelo mais adequado entre os que foram estudados; também se descrevem no Capítulo 3 alguns aspectos computacionais da generalização estocástica aqui apresentada.

No referente aos exemplos numéricos relativos à generalização proposta para o modelo determinístico de dimensionamento e sequenciamento de processos descontínuos, consideram-se dois aspectos: *i*) a generalização do enquadramento determinístico para o estocástico com robustez, ainda na perspectiva de minimização de custos de investimento; *ii*) generalização do número de períodos de tempo, concretizando-se a optimização multiperíodo do VLA, o qual integra os custos de investimento. Descrevem-se as principais características desses exemplos:

- exemplo *spbatch\_ex1* — face ao enquadramento em ambiente de incerteza que generaliza o modelo determinístico e uniperíodo (estático) acima referido, observa-se uma diminuição significativa em termos de custos (de 8 a 20%) perante uma ligeira relaxação das imposições sobre as procuras incertas (menos de 1% no valor esperado da procura não-satisfeita), mesmo perante um número reduzido de cenários; também, realce-se a similitude de comportamento das instâncias tratadas com vários números de cenários (desde 1 a 100), dada a grande dimensão do modelo;
- exemplo *spbatch\_ex2* — além do número de cenários, a dimensão do modelo depende também linearmente do número de períodos considerados; a abordagem multiperíodo (dinâmica) privilegia, como fase inicial e sem prejuízo de se virem a adoptar-se horizontes mais alargados, um número de períodos limitado; tal abordagem foi analisada, verificando-se a sua utilidade prática, mesmo perante instâncias de grande dimensão, nas quais se recorre à obtenção de uma gama de soluções de boa qualidade, mas sem garantia do óptimo.

Comprovou-se a robustez patenteada pelo modelo *spbatch\_milp\_fmsz*, verificando-se que os estimadores respondem adequadamente à variação da penalização da procura não-satisfeita. Em termos de solução, o conceito de robustez também é aqui aplicável, mas deve registar-se que a invariabilidade das configurações resulta principalmente da discretização dos volumes: apenas quando o benefício potencial de um aumento de dimensão for relativamente elevado, suplantando os custos correspondentes, é que esse aumento de dimensão será realizado.

---

Finalmente e comparando com os correspondentes modelos da literatura, descritos no Capítulo 2 e analisados no Capítulo 4, os modelos robustos aqui apresentados afiguram-se de qualidade superior, pois são mais elaborados e aproximados da realidade incerta. Ao integrarem os anteriores modelos e ao não seccionarem o respectivo espaço de procura, qualquer das soluções ou configurações prevista nos modelos da literatura continua a ser válida no âmbito do modelo generalizado: a solução óptima deste modelo generalizado não será de qualidade inferior à do modelo mais específico. Também se verifica a validade desta abordagem em termos quantitativos, já que se observa uma significativa redução nos custos de investimento, decorrente apenas de uma ligeira relaxação de restrições “macias” (*soft*), a qual configura um alargamento do espaço de procura.





### Conclusões e projecções

Neste estudo, apresenta-se uma proposta de tratamento através duma abordagem generalizadora de casos de optimização bem conhecidos na Engenharia Química, nomeadamente: o planeamento de processos; e o dimensionamento e sequenciamento de processos descontínuos.

Tais casos de optimização são formuláveis como modelos de MILP, os quais se apresentam usualmente como de superior complexidade, constituindo problemas matemáticos de grande dimensão, caso se assumam condições realistas. Assim acontece quando se propõe a generalização dos referidos modelos, complementando-os com duas vertentes: uma estocástica, que permite incorporar a incerteza inerente aos problemas; e uma de promoção da robustez, para tratamento do risco.

A proposta de tratamento desses casos de optimização recorre ao desenvolvimento de heurísticas, elaboradas de acordo com o problema modelado, e cuja qualidade é avaliada efectuando-se a necessária análise aos desvios, de índole probabilística, perante instâncias razoáveis.

Também se analisa a optimalidade de cada problema equacionado, tratando-se instâncias associadas a situações problemáticas diversas, onde se comprova a utilidade da abordagem de generalização aqui aplicada, em termos das vertentes citadas. Então, avalia-se a sensibilidade à incerteza e, assumindo uma atitude crítica perante o risco, promove-se a robustez da solução e do modelo. Assim, em detrimento da obtenção dum óptimo matemático, de cariz determinístico e porventura sensível a diminutas variações nos dados, pretende-se valorizar a determinação de gamas de variação para os principais indicadores. Tais gamas correspondem à vizinhança do óptimo robusto, para as instâncias consideradas.

Nas secções subsequentes, descreve-se sumariamente o trabalho desenvolvido e as principais conclusões a retirar quanto à generalização de modelos, no âmbito de problemas complexos da Engenharia Química, o que permite pesquisar espaços de procura de maiores dimensões e obter de soluções de superior qualidade. Apresentam-se também desenvolvimentos recentes e específicos, quer nas áreas de planeamento de processos, de dimensionamento e sequenciamento de processos

descontínuos, quer ainda nos métodos e esquemas de cálculo aí aplicáveis. Daí e projectando desenvolvimentos futuros, perspectivam-se avanços quanto ao trabalho desenvolvido e indicam-se aspectos passíveis de evolução.

## 6.1 Trabalho desenvolvido

Numa abordagem dirigida a cada tema e transversalmente aos vários Capítulos, revêem-se os aspectos mais relevantes do trabalho desenvolvido.

### 6.1.1 Planeamento de processos

No planeamento a longo prazo dos investimentos em sistemas de processos químicos (*process planning*), constata-se —da Sec. 2.1— que a abordagem à modelação apresenta duas tendências principais: *i*) desenvolvem-se sucessivamente formulações que se aproximam cada vez mais dos aspectos reais, quer por incorporação de características de resposta à variabilidade, quer por tratamento estocástico da incerteza associada a horizontes de longo prazo; e *ii*) introduzem-se, ainda na fase de modelação, características favorecedoras dos esquemas de cálculo a utilizar para aumentar a eficiência computacional, como sejam a delimitação do espaço de procura (incorporando planos de corte, por reformulação das subestruturas matemáticas do ELS), ou o recurso a procedimentos heurísticos preparados a partir das propriedades analíticas dos modelos.

Implementaram-se computacionalmente modelos de planeamento de processos —na Sec. 3.1— em linguagem Fortran e dirigindo a codificação para a utilização de rotinas de optimização da biblioteca NAG. Os códigos foram testados, por verificação de exemplos numéricos da literatura, ou por consistência dos resultados para instâncias específicas. Procedeu-se a uma progressão de cariz modular para as diversas implementações: iniciou-se com um modelo MILP determinístico; prosseguiu-se com uma formulação estocástica com recurso (2SSP), incorporando-se a técnica de amostragem estratificada; implementou-se a decomposição de Benders, para o problema generalizado e de grande dimensão (esparso), com robustez e considerando processos flexíveis, revelando-se extremas dificuldades no tratamento computacional. Como se verificou ser bastante morosa a codificação em Fortran dos diferentes modelos de Programação Matemática, sistematizou-se a definição dos coeficientes dos modelos (referentes a função objectivo, matriz das restrições e termos independentes), de forma similar à utilizada em ambientes de modelação, permitindo-se o acesso mais directo à biblioteca NAG. Daqui também ressalta a necessidade de meios especificamente construídos para a implementação computacional de modelos matemáticos, como o ambiente GAMS.

Na Sec. 4.1 procedeu-se a uma abordagem teórica aos problemas de planeamento de processos, comprovando-se a dificuldade de resolução dos modelos, pois procedeu-se à classificação da respectiva complexidade computacional. Dado que se tratam genericamente de problemas NP-hard, resulta a necessidade do desenvolvimento de procedimentos heurísticos para tratar instâncias de grande dimensão, mas sendo resolúveis problemas de menor dimensão. Esta abordagem teórica possibilitou um conhecimento detalhado das vantagens e limitações dos diversos modelos de planeamento de processos, constatando-se a pertinência de um estudo da optimalidade, para garante da robustez da solução referente à configuração

do sistema de processos químicos, atendendo à incerteza no planeamento de médio e longo prazo.

Consequentemente, propõe-se —na Sec. 5.1— um modelo estocástico, *spplan\_milp\_flex*, que generaliza aspectos de flexibilidade (na utilização de matérias-primas ou de produtos intermédios, na definição dos fluxos dos produtos) com o tratamento da incerteza associada a um horizonte de planeamento alargado. Aplica-se uma via estocástica e robusta, incorporando-se a atitude perante o risco do decisor na configuração a seleccionar. Analisando-se a robustez, constatou-se que os diversos estimadores (valores esperados dos indicadores económicos e técnicos) são manipuláveis, na gama de variação testada para os parâmetros de penalização. Ilustra-se a aplicabilidade do modelo com três exemplos numéricos, os quais consideram variantes em que se estudam vias alternativas de produção, que podem servir como referência para estimação dos parâmetros técnicos que tornam vantajosos os processos: no exemplo *spplan\_ex1*, trata-se uma situação simplificada, que permite a especificação dos parâmetros de penalização, promovendo-se a robustez da solução e do modelo; no exemplo *spplan\_ex2*, visa-se a necessidade de implementar diferentes vias para produção de um dado componente (reagente), a sua substituição por um outro componente mediante um processo alternativo, na situação onde esse componente apresente um valor elevado ou uma diminuta disponibilidade; e no exemplo *spplan\_ex3*, analisa-se a situação onde, a um produto de interesse económico, se encontra associado um outro produto, mas não desejado e que é necessário transformar, comparando-se as alternativas de processamento desse produto de interesse reduzido.

Logo, no contexto do problema de planeamento de processos, aliado à abordagem teórica efectuada sobre os modelos da literatura, desenvolveu-se um modelo generalizado, o qual congrega aspectos de flexibilidade na utilização de materiais e de tratamento da incerteza. Foram implementados diversos exemplos ilustrativos, no âmbito deste modelo, os quais provam a utilidade do tratamento.

### 6.1.2 Dimensionamento e sequenciamento de processos descontínuos

Outra aplicação do tema desta tese é a generalização de modelos que consideram o sequenciamento do ciclo de produção no âmbito do dimensionamento de processos descontínuos (*design and scheduling of batch processes*), em ambiente multiproduto (*flowshop*). Portanto, não se enquadra aqui o sequenciamento multiobjectivo (*jobshop*).

Iniciou-se a Sec. 2.2 apresentando os aspectos que complementam a questão integrada do sequenciamento no dimensionamento dos processos descontínuos. As modelações sucessivas vão alargando a gama de opções (estágios simples ou com vários processos em paralelo, campanhas de produto único ou com produtos múltiplos, pressupondo alguma política de armazenagem ou preferindo espera nula), observando-se que o espaço de procura dos problemas se vai também estendendo. Também se registam alguns procedimentos de natureza numérica para favorecer a optimização do sequenciamento da produção, sendo observadas: a aplicação de restrições de carácter lógico; a convexificação de modelos não lineares; a aplicação de subestruturas matemáticas do problema ELS; a utilização de estruturas SOS.

Na Sec. 3.2, para implementação dos modelos de interesse em ambiente GAMS e numa metodologia modular, procedeu-se sequencialmente a: implementação de modelos determinísticos, disponíveis na literatura; realização de análise probabilística, conjugando-se o ambiente GAMS com a codificação em

Fortran de diversos procedimentos heurísticos; implementação do modelo generalizado, apresentando vertentes estocástica e com robustez, em face da incerteza associada. As implementações dos modelos foram testadas, bem como foram aperfeiçoadas técnicas para inicialização e para delimitação de pesquisa que possam tornar mais eficiente a resolução pelos *solvers*. Também, comparando com a codificação em Fortran: confirmou-se que a concretização através do ambiente GAMS é bem mais eficiente, além da facilidade de correcção ou de eventual ajuste; porém, a utilização de ciclos de atribuição conjugados com aspectos condicionais, necessários à implementação de heurísticas, constitui uma codificação com um grau de dificuldade e de ineficiência elevado em ambiente GAMS. Algumas aplicações de suporte, quer em Fortran quer em GAMS, são descritas no Apêndice 3.

Os modelos dirigidos ao dimensionamento e sequenciamento de processos descontínuos são estudados e comparados na Sec. 4.2, comprovando-se no Apêndice 4-A a ocorrência de relações de equivalência entre instâncias específicas de quatro modelos alternativos —correspondentes à combinatória de {processo único, processos múltiplos} com {campanhas de produto único, campanhas de processos múltiplos}— o que indica similar dificuldade de resolução computacional. Para cada modelo estudado, descrevem-se os aspectos mais relevantes do trabalho desenvolvido: *i)* para o modelo *batch\_milp\_fssz*, foram desenvolvidos diversos procedimentos de aproximação e realizou-se a análise probabilística aos desvios, obtendo-se uma fracção significativa de soluções óptimas, com desvios de cerca de 1% para as soluções subóptimas; *ii)* para *batch\_milp\_fmsz*, também foram desenvolvidos procedimentos de aproximação, com a análise probabilística respectiva, sendo obtidos resultados de qualidade similar aos do modelo anterior; *iii)* para *batch\_milp\_fsmz* (considerando MPC), observou-se que a resolução era eficiente mas condicionada pela não exigência de integralidade do número de *batches*, e que se permite prever uma diminuição dos custos de investimento de cerca de 5%, relativamente ao modelo *batch\_milp\_fssz* (SPC); e *iv)* o modelo *batch\_milp\_fmsz*, supondo processos múltiplos por estágio, permite ultrapassar as dificuldades na obtenção de soluções admissíveis perante instâncias realistas (aumento do número de produtos ou das quantidades procuradas) do modelo anterior (a processo único), mas a existência de contradições teóricas conduz a incoerência numérica.

Na Sec. 5.2, propõe-se a generalização do modelo determinístico de *batch\_milp\_fmsz* (*Flowshop/ Multiple machine/ SPC/ Zero wait*) a um enquadramento estocástico em duas fases e com promoção da robustez, através do modelo *spbatch\_milp\_fmsz*. Tal enquadramento em ambiente de incerteza permite uma diminuição significativa (de 8% a 20%) em termos de custos perante uma ligeira relaxação das imposições sobre as procuras incertas (respectivamente, de 1% a 6%), mesmo perante um número reduzido de cenários. Nas instâncias testadas, verificou-se uma semelhança do comportamento dos diversos estimadores perante a variação do número de cenários, além de se sustentar a aplicabilidade e suficiência na utilização de um número limitado de períodos de tempo. Também se comprovou a robustez do modelo *spbatch\_milp\_fmsz*, verificando-se a adequação dos estimadores à variação da penalidade por procura não-satisfeita, bem como a robustez da solução (resultando a invariabilidade das configurações principalmente da discretização dos volumes).

Assim, decorrendo do estudo das modelações existentes —no âmbito do problema de dimensionamento e sequenciamento *flowshop* de processos descontínuos—, desenvolveram-se procedimentos heurísticos para modelos de

interesse, tendo-se proposto generalizar o modelo determinístico que se revelou de aplicação mais realista, por adaptação a um enquadramento estocástico e robusto, além de multiperíodo. Apresentam-se diversos exemplos numéricos, que permitem concluir do interesse desta abordagem, dadas as perspectivas de diminuição significativa dos custos de investimento e de obtenção de robustez.

## 6.2 Desenvolvimentos recentes e afins

Dado o interesse que os assuntos focados —tanto os problemas em estudo, como os métodos de resolução— desde sempre têm suscitado, descrevem-se ainda alguns desenvolvimentos recentes e afins ocorridos em simultâneo com o trabalho desta Tese. Desta forma, visa-se enquadrar o trabalho aqui desenvolvido para se constatar da sua actualidade, embora se fundamente em modelos algo datados.

No contexto do planeamento de processos, Ahmed e Sahinidis [2003] apresentam um modelo estocástico para expansão de capacidades no enquadramento de *Multi-Stage Stochastic Integer Programming* (MSSIP), com um modelo de planeamento de processos a ser considerado em cada fase. Então, como os custos de expansão de capacidades são formulados como custos com encargos fixos, recorrendo a variáveis binárias em cada fase, obtém-se um problema NP-hard. Supondo a relaxação das restrições de integralidade e de não-antecipação, procede-se à decomposição do problema de MSSIP numa sequência de subproblemas determinísticos de LP, aos quais se aplica um procedimento de aproximação baseado na agregação de capacidades nos períodos iniciais, similar ao realizado anteriormente pelos mesmos autores [Ahmed e Sahinidis, 2000]. Para evitar que essa agregação de capacidades em períodos iniciais viole as restrições de não-antecipação, aplica-se um procedimento de correcção onde se igualam as capacidades de um processo, em todos os cenários de uma dada fase, ao valor máximo em todos esses cenários. Então, realiza-se uma análise probabilística, tendo-se obtido sempre soluções admissíveis, cujos desvios diminuem assintoticamente para zero com o aumento da dimensão do problema. A inovação deste trabalho decorre da existência de variáveis binárias em múltiplas fases para além da primeira, o que se traduz em não-convexidades, sendo mesmo assim possível realizar um tratamento computacional eficiente quando não existe qualquer método genérico para abordar este problema de MSSIP. Porém, tal procedimento de aproximação (que agrega as expansões em períodos iniciais) apenas será válido perante instâncias que considerem que o encargo fixo associado às expansões é elevado, pelo que apresenta um risco significativo para o investimento caso não se confirmem as previsões de procura.

Ultimamente, têm sido direccionados esforços para a optimização dos factores financeiros, como sejam o tratamento do risco financeiro [Lavaja *et al.*, 2006], a modelação dos contratos de compra e venda no âmbito da cadeia de fornecimentos [Park *et al.*, 2006], além da consideração dos inventários para atenuar as flutuações das procuras [Ryu, 2006].

No contexto do dimensionamento e sequenciamento de processos descontínuos, têm surgido com alguma frequência comunicações registando vias estocásticas ou robustas, em resposta a variações quer externas (preços e procuras dos produtos), quer internas (dos parâmetros técnicos, tempos de processamento), bem como se recorre a métodos de resolução aproximativos.

Dietz *et al.* [2006] tratam o dimensionamento de unidades multiproducto descontínuas, associado às respeitantes condições operacionais para produção de quatro proteínas em oito estágios sequenciais, através duma optimização multicritério: minimizam-se os custos de investimento, incluindo-se o número de equipamentos em dimensões discretas, bem como o impacto ambiental, considerando ainda um índice de flexibilidade para o funcionamento. Devido à dificuldade inerente aos métodos de optimização combinatória, os autores optaram pela conjugação dum algoritmo genético estocástico com um simulador discreto (*discrete-event simulator*, DES), procedendo a uma avaliação Pareto das soluções obtidas para o dimensionamento dos equipamentos. Daqui ressalta a importância de considerar a optimização dos processos e do impacto ambiental logo na fase inicial de desenvolvimento do sistema de produção descontínua.

Moreno *et al.* [2006] apresentam um modelo multiperíodo para optimização das decisões de dimensionamento e planeamento em unidades multiproducto descontínuas, considerando: valores de retorno e custos operacionais e de investimento; afectação de dimensões discretas às unidades, descontínuas ou semicontínuas; implementação de armazenagem intermédia; e variações das procuras e preços dos componentes, devido tanto a efeitos sazonais como a estruturais. Deste modo, um caso realista, envolvendo a produção de cinco oleorresinas em sete estágios, é tratado até à optimalidade despendendo um esforço computacional razoável. De realçar a abordagem em múltiplos períodos para este tipo de problema, o que se revela decisivamente vantajoso, bem como a interacção existente entre os planos de decisão, quanto ao dimensionamento e ao planeamento, no âmbito de diferentes políticas: comercial, de produção, de vendas, de inventário.

Yi e Reklaitis [2006] apresentam uma solução analítica para o problema de dimensionamento óptimo de uma rede de armazenagem destinada a produtos finais, em série ou em paralelo, no contexto de variações aleatórias nos tempos de operação ou nas disponibilidades de matérias-primas. A função objectivo a minimizar consiste nos custos de pronto e de armazenagem, bem como nos de investimento. Os autores desenvolveram um modelo de análise da produção e inventário, que possibilita a obtenção de soluções analíticas através de um método gráfico para delimitar os fluxos aleatórios. Consequentemente, o esforço computacional vai diminuir, providenciando resultados úteis para as decisões de investimento a tomar em ambiente incerto.

Em termos de desenvolvimentos afins ao da generalização de modelos aqui tratada, refira-se Goyal e Ierapetritou [2003], que formalizam a expansão das fronteiras do planeamento através de análise dos dados, bem como Banerjee e Ierapetritou [2004], que encaram a propagação do modelo em ambiente de incerteza e a partir de um conjunto de execuções simuladas.

### 6.3 Perspectivas e desenvolvimentos futuros

É patente que os resultados obtidos favorecem a prossecução de trabalhos de investigação nesta área de optimização matemática. Parece pertinente o estudo de novas técnicas de resolução aplicáveis a problemas de MILP em ambiente de incerteza, como seja a aplicação de regras heurísticas obtidas a partir de análise fundamental, e de novas formas de modelação generalizada para os problemas desta área da engenharia de processos químicos. Sinteticamente, abordam-se vias para

possíveis trabalhos de investigação, bem como se apontam diversas limitações que conviria tratar em posteriores desenvolvimentos.

Atendendo a Ahmed e Sahinidis [2003], uma abordagem de MSSIP aparece como complementar aos modelos propostos no Cap. 5. No caso do planeamento de processos flexíveis com robustez, em *spplan\_milp\_flex* (Sec. 5.1), a aplicação de uma heurística similar à destes autores (agregação das expansões de capacidade nos períodos iniciais) encontra-se em oposição ao propósito de incorporar robustez no modelo através da penalização das folgas de capacidades. No entanto, interessa proceder à decomposição em múltiplos subproblemas, os quais conviria resolver eficientemente utilizando algum procedimento heurístico. Quanto ao tratamento do *spbatch\_milp\_fmsz* em regime de MSSIP, é de considerar o faseamento da implementação das capacidades, através da paralelização de processos em cada estágio, em vez de se implementar apenas no período inicial. Contudo, recorde-se a dificuldade extrema da resolução de instâncias de grande dimensão neste problema de MILP, devido ao elevado número de variáveis contínuas, bem como de restrições. Então, se possível, é de considerar a integração dos fluxos, financeiros com os materiais, e o tratamento do risco financeiro.

Relativamente ao trabalho de Dietz *et al.* [2006], saliente-se que é passível de implementação um algoritmo genético (ou outra heurística) para procurar outras soluções boa qualidade, quando não há garantia de optimalidade. Tal acontece no último exemplo numérico, *spbatch\_ex2*, para dimensionamento e sequenciamento multiperíodo e com robustez, onde apenas é possível definir uma gama de soluções alternativas, hierarquizadas segundo um estimador específico. Seria então possível implementar um algoritmo aproximativo, em ambiente GAMS e englobando um procedimento automático, recorrendo à técnica de fixação de variáveis binárias.

Outra generalização a equacionar é a de alargar a modelação efectuada para dimensionamento e sequenciamento a outras políticas de produção e armazenagem. Combinando aspectos do trabalho de Moreno *et al.* [2006], que consideram produção SPC num horizonte multiperíodo, com outros como o de Yi e Reklaitis [2006], que desenvolvem uma estimativa analítica para os custos de armazenagem, será oportuno comparar a consideração detalhada de políticas alternativas para produção e armazenagem. Também aí se prevê, ante a dificuldade da tarefa, o desenvolvimento de heurísticas, bem como a necessidade de novos avanços teóricos. Um caso é a consideração de MPC em estágios com processos múltiplos, cujo sequenciamento recorre a uma formulação por TSP, para o que se lobriga algum tipo de reformulação.

Contudo, a aplicação a casos realistas de eventuais progressos de carácter teórico, ou de adequação em termos de modelação (multicritério, detalhe dos aspectos operacionais), ou melhorias na implementação computacional e tratamento numérico, estará sempre dependente das ferramentas disponíveis.

Finalmente, deseja-se que a Engenharia Química vença os novos desafios no âmbito dos sistemas de processos, coadjuvada por um instrumento de função tão relevante como a Optimização, com procedimentos eficazes e cada vez mais eficientes.





## Bibliografia

---

- Abecassis, F., Cabral, N., *Análise económica e financeira de projectos* (Fundação Calouste Gulbenkian, Lisboa, 1991)
- Adams, J.C., Brainerd, W.S., Martin, J.T., Smith, B.T., Wagener, J.L., *Fortran 90 Handbook* (McGraw-Hill, New York, 1992)
- Ahmed, S., Sahinidis, N.V., An approximation scheme for stochastic integer programs arising in capacity expansion, *Operations Research* **51** 3 (2003) 461–471
- Ahmed, S., Sahinidis, N.V., Analytical investigations of the process planning problem, *Computers and Chemical Engineering* **23** (2000) 1605-1625
- Ahmed, S., Sahinidis, N.V., Pistikopoulos, E. N., An improved decomposition algorithm for optimization under uncertainty, *Computers and Chemical Engineering* **23** (2000) 1589-1604
- Ahmed, S., Sahinidis, N.V., Robust process planning under uncertainty, *Ind. Eng. Chem. Res.* **37** (1998) 1883-1892
- Athier, G., Floquet, P., Pibouleau, L., Domenech, S., Optimization of heat exchanger networks by coupled Simulated Annealing and NLP procedures, *Computers and Chemical Engineering* **20** Suppl.(1996) S13-S18
- Banerjee, I., Ierapetritou, M.G., Model Independent Parametric Decision Making, *Annals of Operations Research* **132** (2004) 135–155
- Barbaro, A., Bagajewicz, M.J., Managing financial risk in planning under uncertainty, *AIChE J.* **50** (2004a) 963–989
- Barbaro, A., Bagajewicz, M.J., Use of inventory and option contracts to hedge financial risk in planning under uncertainty, *AIChE J.* **50** (2004b) 990–998
- Barbosa-Póvoa, A.P., Macchietto, S., Detailed design of multipurpose batch plants, *Computers and Chemical Engineering* **18** (1994) 1013-1042
- Barbosa-Póvoa, A.P., Pantelides, C.C., Design of multipurpose plants using the resource-task network unified framework, *Computers and Chemical Engineering* **21** Suppl. (1997) S703-S708
- Barbosa-Póvoa, A.P., Pantelides, C.C., Design of multipurpose production facilities: a RTN decomposition-based algorithm, *Computers and Chemical Engineering* **23** Suppl. (1999) S7-S10
- Barros, C., *Decisões de investimento e financiamento de projectos* (Ed. Sílabo, Lisboa, 1991)
- Bassett, M.H., Gardner, L.L., Steele, K., Dow AgroSciences uses simulation-based optimization to schedule the new-product development process, *Interfaces* **34** 6 (2004) 426-437

- 
- Bernal-Haro, L., Azzaro-Pantel, C., Domenech, S., Pibouleau, L., Design of multipurpose batch chemical plants using a Genetic Algorithm, *Computers and Chemical Engineering* **22** Suppl. (1998) S777-S780
- Biegler, L. T., Grossmann, I. E., Retrospective on optimisation, *Computers and Chemical Engineering*, **28** (2004) 1169-1192
- Biegler, L.T., Grossmann, I.E., Westerberg, A.W., *Systematic Methods of Chemical Process Design* (Prentice-Hall, New Jersey, 1997).
- Birewar, D.B., Grossmann, I.E., Efficient optimization algorithms for Zero-Wait scheduling of multiproduct batch plants, *Ind. Eng. Chem. Res.* **28** (1989a) 1333-1345
- Birewar, D.B., Grossmann, I.E., Incorporating scheduling in the optimal design of multiproduct batch plants, *Computers and Chemical Engineering* **13** (1989b) 141-161
- Blau, G., Mehta, B., Bose, S., Pekny, J., Sinclair, G., Keunker, K., Bunch, P., Risk management in the development of new products in highly regulated industries, *Computers and Chemical Engineering* **24** (2000) 659-664
- Bok, J., Grossmann, I.E., Park, S., Supply chain optimization in continuous flexible process networks, *Ind. Eng. Chem. Res.* **39** (2000) 1279-1290
- Bok, J.K., Lee, H., Park, S., Robust investment model for long-range capacity expansion of chemical processing networks under uncertain demand forecast scenarios, *Computers and Chemical Engineering* **22** (1998) 1037-1049
- Bose, S., Pekny, J.F., A model predictive framework for planning and scheduling problems: a case study of consumer goods supply chain, *Computers and Chemical Engineering* **24** (2000) 329-335
- Boykin, R.F., Optimizing chemical production at Monsanto, *Interfaces* **15** 1 (1985) 88-95
- Brooke, A., Kendrick, D., Meeraus, A., *GAMS: A User's Guide* (The Scientific Press, USA, 1992)
- Camm, J.D., Chorman, T.E., Dill, F.A., Evans, J.R., Sweeney, D.J., Wegryn, G.W., Blending OR/MS, judgment, and GIS: restructuring P&G's supply chain, *Interfaces* **27** (1997) 128-142
- Caroe, C.C., Schultz, R., Dual decomposition in stochastic integer programming, *Operations Research Letters* **24** (1999) 37-45
- Cavin, L., Fischer, U., Gloverb, F., Hungerbühler, K., Multi-objective process design in multi-purpose batch plants using a Tabu Search optimization algorithm, *Computers and Chemical Engineering* **28** (2004) 459-478
- Clay, R.L., Grossmann, I.E., A disaggregation algorithm for the optimization of stochastic planning models, *Computers and Chemical Engineering* **21** (1997) 751-774

- 
- Coulson, J.M., Richardson, J.F., *Tecnologia Química*, Vol. VI (Fundação Calouste Gulbenkian, Lisboa, 1989)
- D'Alessandro, A.J., Baveja, A., Divide and conquer: Rohm and Haas' response to a changing specialty chemicals market, *Interfaces* **30** (2000) 1-16
- Daichendt, M.M., Grossmann, I.E., Integration of hierarchical decomposition and mathematical programming for the synthesis of process flowsheets, *Computers and Chemical Engineering* **22** (1997) 147-175
- Dietz, A., Azzaro-Pantel, C., Pibouleau, L., Domenech, S., Multiobjective optimization for multiproduct batch plant design under economic and environmental considerations, *Computers and Chemical Engineering* **30** (2006) 599-613
- Diwekar, U.M., Kalagnanam, J.R., Efficient sampling technique for optimization under uncertainty, *AIChE J.* **43** (1997) 440-447
- Douglas, J.M., *Conceptual Design of Chemical Processes* (McGraw Hill, New York, 1988)
- Edgar, T.F., Himmelblau, D.M., *Optimization of Chemical Processes* (McGraw-Hill, Singapore, 1989).
- Eppen, G.D., Gould, F.J., Schmidt, C.P., *Introductory Management Science* (Prentice-Hall, Fourth Ed., 1993)
- Eppen, G.D., Martin, R.K., Solving multi-item capacitated lot-sizing problems using variable redefinition, *Operations Research* **35** 6 (1987) 832-848
- Erlenkotter, D., A dual based procedure for uncapacitated facility location, *Operations Research* **26** 6 (1978) 992-1009
- Erlenkotter, D., Manne, A.S., Capacity expansion for India's nitrogenous fertilizer industry, *Management Science* **14** 10 (1967) B553-B572
- Floudas, C.A., Research Challenges, Opportunities and Synergism in Systems Engineering and Computational Biology, *AIChE J.* **51** (2005) 1872-1884
- Floudas, C.A., Lin, X., Continuous-time versus discrete-time approaches for scheduling of chemical processes: a review, *Computers and Chemical Engineering* **28** (2004) 2109-2129
- Furman, K.C., Sahinidis, N.V., Computational Complexity of Heat Exchanger Network Synthesis, *Computers and Chemical Engineering* **25** (2001) 1371-1390
- Garey, M. R., Johnson, D. S., *Computers and Intractability: a guide to the theory of NP-completeness* (W. H. Freeman & Co., New York, 1979)

- 
- Gnoni, M.G., Iavagnilio, R., Mossa, G., Mummolo, G., Di Leva, A., Production planning of a multi-site manufacturing system by hybrid modelling: A case study from the automotive industry, *Int. J. Production Economics* **85** (2003) 251–262
- González, J.R., Palahy, M., Pukkala, T., Integrating fire risk considerations in forest management planning in Spain – a landscape level perspective, *Landscape Ecology* **20** (2005) 957–970
- Goyal, V., Ierapetritou, M.G., Integration of data analysis and design optimization for the systematic generation of equipment portfolio, *Ind. Eng. Chem. Res.* **42** (2003) 5204-5214
- Grossmann, I. E., Biegler, L. T., Future perspective on optimisation, *Computers and Chemical Engineering* **28** (2004) 1193–1218
- Grossmann, I.E., Challenges in the new millennium: product discovery and design, enterprise and supply chain optimization, global life cycle assessment, *Computers and Chemical Engineering* **29** (2004) 29–39
- Grossmann, I.E., Enterprise-wide optimization: a new frontier in Process Systems Engineering, *AIChE J.* **51** (2005) 1846-1857
- Grossmann, I.E., Westerberg, A.W., Research challenges in process systems engineering, *AIChE J.* **46** (2000) 1700-1703
- Grunow, M., Gunther, H.-O., Yang, G., Plant co-ordination in pharmaceuticals supply networks, *OR Spectrum* **25** (2003) 109–141
- Gupta, A., Maranas, C.D., Managing demand uncertainty in supply chain planning, *Computers and Chemical Engineering* **27** (2003) 1219-1227
- Hahn, G.J., Shapiro, S.S., *Statistical Models in Engineering* (John Wiley & Sons, New York, 1967)
- Himmelblau, D.M., Bickel, T.C., Optimal expansion of a hydrodesulfurization process, *Computers and Chemical Engineering* **4** (1980) 101-112
- Hugo, A., Pistikopoulos, E.N., Environmentally conscious long-range planning and design of supply chain networks, *Journal of Cleaner Production* **13** (2005) 1471-1491
- Ierapetritou, M. G., Pistikopoulos, E. N., Design of multiproduct batch plants with uncertain demands, *Computers and Chemical Engineering* **19** Suppl. (1995) S627-S632
- Ierapetritou, M.G., Pistikopoulos, E. N., Novel optimization approach of stochastic planning models, *Ind. Eng. Chem. Res.* **33** (1994) 1930-1942
- Ierapetritou, M.G., Pistikopoulos, E.N., Batch plant design and operations under uncertainty, *Ind. Eng. Chem. Res.* **35** (1996) 772-787

- 
- Jackson, J.R., Grossmann, I.E., Temporal decomposition scheme for nonlinear multisite production planning and distribution models, *Ind. Eng. Chem. Res.* **42** (2003) 3045-3055
- Jayaraman, V.K., Kulkarni, B.D., Karale, S., Shelokar, P., Ant colony framework for optimal design and scheduling of batch plants, *Computers and Chemical Engineering* **24** (2000) 1901–1912
- Jimenez, A.G., Rudd, D.F., Use of a recursive mixed-integer programming model to detect an optimal integration sequence for the mexican petrochemical industry, *Computers and Chemical Engineering* **11** (1987) 291-301
- Julka, N., Srinivasan, R., Karimi, I., Agent-based supply chain management. 1: A refinery application, *Computers and Chemical Engineering* **26** (2002) 1771-1781
- Kallrath, J., Combined strategic and operational planning: An MILP success story in chemical industry, *OR Spectrum* **24** (2002b) 315–341
- Kallrath, J., Planning and scheduling in the process industry, *OR Spectrum* **24** (2002a) 219–250
- Kelly, J.D., Mann, J.L., Flowsheet decomposition heuristic for scheduling: a relax-and-fix method, *Computers and Chemical Engineering* **28** (2004) 2193–2200
- Klingman, D., Phillips, N., Steiger, D., Wirth, Young, W., The challenges and success factors in implementing an integrated products planning system for Citgo, *Interfaces* **16** 3 (1986) 1-19
- Klingman, D., Phillips, N., Steiger, D., Young, W., The successful deployment of Management Science throughout Citgo Petroleum Corporation, *Interfaces* **17** 1 (1987) 4-25
- Kocis, G.R., Grossmann, I.E., Global optimization of nonconvex Mixed-Integer Nonlinear Programming (MINLP) problems in process synthesis, *Ind. Eng. Chem. Res.* **27** (1988) 1407-1421
- Kondili, E., Pantelides, C.C., Sargent, R.W.H., A general algorithm for short-term scheduling of batch operations—I. MILP formulation, *Computers and Chemical Engineering* **17** (1993) 211-227
- Krarup, J. and Bilde, O., Plant location, set covering and economic lot sizes: An O(mn) algorithm for structured problems, in : Collatz, L. (ed.), *Optimeirung bei Graphentheoretischen und Ganzzahligen Probleme* (Birkhauser Verlag, Basel, 1977) 155-180
- Kuik, R., Solomon, M., Van Wassenhove, L.N., Batching decisions: structure and models, *European Journal of Operational Research* **75** (1994) 243-263
- Laguna, M., Applying robust optimization to capacity expansion of one location in telecommunications with demand uncertainty, *Management Science* **44** (1998) S101-S110

- 
- Lasdon, L.S., Terjung, R.C., An efficient algorithm for multi-item scheduling, *Operations Research* **19** (1971) 946-969
- Lasschuit, W., Thijssen, N., Supporting supply chain planning and scheduling decisions in the oil and chemical industry, *Computers and Chemical Engineering* **28** (2004) 863-870
- Lavaja, J., Adler, A., Jones, J., Pham, T., Smart, K., Splinter, D., Steele, M., Bagajewicz, M.J., Financial risk management for investment planning of new commodities considering plant location and budgeting, *Ind. Eng. Chem. Res.* **45** (2006) 7582-7591
- Lee, Y.M., Chen, E.J., BASF uses a framework for developing Web-based production-planning-optimization tools, *Interfaces* **32** 6 (2002) 15-24
- Linnhoff, B., Townsend, D.W., Boland, D., Hewitt, G.F., Thomas, B.E.A., Guy, A.R., Marsland, R.H., *A User Guide On Process Integration For The Efficient Use Of Energy* (The Institution of Chemical Engineers, Rugby , 1982)
- Liu, M.L., Sahinidis, N.V., Bridging the gap between heuristics and optimization: capacity expansion case, *AIChE J.* **43** (1997) 2289-2299
- Liu, M.L., Sahinidis, N.V., Computational trends and effects of approximations in an MILP model for process planning, *Ind. Eng. Chem. Res.* **34** (1995) 1662-1673
- Liu, M.L., Sahinidis, N.V., Optimization in process planning under uncertainty, *Ind. Eng. Chem. Res.* **35** (1996) 4154-4165
- Lohl, T., Schulz, C., Engell, S., Sequencing of batch operations for a highly coupled production process: Genetic Algorithms Versus Mathematical Programming, *Computers and Chemical. Engineering* **22** Suppl. (1998) S579-S585
- Malcolm, S.A., Zenios, S.A., Robust optimization for power systems capacity expansion under uncertainty, *Journal Operations Research Society* **45** (1994) 1040-1049
- Manne, A.S., Markowitz, H. (eds), *Studies in Process Analysis* (John Wiley & Sons, New York and London, 1963)
- Martin, R.K., Generating alternative mixed-integer programming models using variable redefinition, *Operations Research* **35** 6 (1987) 820-831
- Modi, A.K., Karimi, I.A., Design of multiproduct batch processes with finite intermediate storage, *Computers and Chemical Engineering* **13** (1989) 127-139
- Moreno, M.S., Montagna, J.M., Iribarren, O.A., Multiperiod optimization for the design and planning of multiproduct batch plants, *Computers and Chemical Engineering* xxx (2006) xxx-xxx (IN PRESS)

- 
- Moro, L. F. L., Zanin, A. C., Pinto, J. M., A planning model for refinery diesel production, *Computers and Chemical Engineering* **22** Supp. (1998) S1039-S1042
- Moro, L. F. L., Pinto, J. M., Mixed-Integer Programming approach for short-term crude oil scheduling, *Ind. Eng. Chem. Res.* **43** (2004) 85-94
- Neiro, S. M. S., Pinto, J. M., A general modeling framework for the operational planning of petroleum supply chains, *Computers and Chemical Engineering* **28** (2004) 871–896
- Norton, L.C., Grossmann, I.E., Strategic planning for complete process flexibility, *Ind. Eng. Chem. Res.* **33** (1994), 69-76
- Oh, H., Karimi, I.A., .Regulatory factors and capacity-expansion planning in global chemical supply chains, *Ind. Eng. Chem. Res.* **43** (2004) 3364-3380
- Pacheco, J.R.S., Teixeira, M.R., Projecto Cleópatra II, em *IO2006 - 12.º Congresso da APDIO*, ISEG, Lisboa, 2006
- Papadimitriou, C. H., Steiglitz, K., *Combinatorial Optimization: algorithms and complexity* (Prentice-Hall, Englewood Cliffs, 1982)
- Papageorgiou, L.G., Rotstein, G.E., Shah, N., Strategic supply chain optimization for the pharmaceutical industries, *Ind. Eng. Chem. Res.* **40** (2001) 275-286
- Park, M., Park, S., Mele, F.D., Grossmann, I.E., Modeling of purchase and sales contracts in supply chain optimization, *Ind. Eng. Chem. Res.* **45** (2006) 5013-5026
- Pekny, J.F., Miller, D.L., Exact solution of the no-wait flowshop scheduling problem with a comparison to heuristic methods, *Computers and Chemical Engineering* **15** (1991) 741-748
- Penttinen, M.J., Impact of stochastic price and growth processes on optimal rotation age, *Eur. J. Forest Res.* **125** (2006) 335–343
- Petkov, S.B., Maranas, C.D., Design of single-product campaign batch plants under demand uncertainty, *AIChE J.* **44** (1998) 896-911
- Pinto, J. M., Joly, M., Moro, L. F. L. Planning and scheduling models for refinery operations, *Computers and Chemical Engineering* **24** (2000) 2259–2276
- Realf, M.J., Ammons, J.C., Newton, D., Strategic design of reverse production systems, *Computers and Chemical Engineering* **24** (2000) 991-996
- Ribeiro, F.R., Nunes, C.P., *As Indústrias Químicas em Portugal. Perspectivas para o Século XXI* (Escolar Editora, Lisboa, 2001)
- Rodera, H., Bagajewicz, M.J., Trafalis, T.B., Mixed-integer multiobjective process planning under uncertainty, *Ind. Eng. Chem. Res.* **41** (2002) 4075-4084

- 
- Romero, J., Badell, M., Bagajewicz, M., Puigjaner, L., Integrating budgeting models into scheduling and planning models for the chemical batch industry, *Ind. Eng. Chem. Res.* **42** (2003) 6125-6134
- Rudd, D.F., Watson, C.C., *Strategy of Process Engineering* (John Wiley & Sons, New York, 1968).
- Ryu, J., Multiperiod planning strategies with simultaneous consideration of demand fluctuations and capacity expansion, *Ind. Eng. Chem. Res.* **45** (2006) 6622-6625
- Sahinidis, N.V., Grossmann, I.E., Fornari, R.E., Chathrathi, M., Optimization model for long range planning in the chemical industry, *Computers and Chemical Engineering* **13** (1989) 1049-1063
- Sahinidis, N.V., Grossmann, I.E., Multiperiod Investment Model for Processing Networks with Dedicated and Flexible Plants, *Ind. Eng. Chem. Res.* **30** (1991b) 1165-1171
- Sahinidis, N.V., Grossmann, I.E., Reformulation of multiperiod MILP models for planning and scheduling of chemical processes, *Computers and Chemical Engineering* **15** (1991a) 255-272
- Sahinidis, N.V., Grossmann, I.E., Reformulation of the multiperiod MILP model for capacity expansion of chemical processes, *Operations Research* **40** (1992) S127-S144
- Sargent, R., Process systems engineering: A retrospective view with questions for the future, *Computers and Chemical Engineering* **29** (2005) 1237-1241
- Schrage, L., *Linear Programming Models with LINDO* (Lindo Systems, USA, 1986)
- Shah, N., Pharmaceutical supply chains: key issues and strategies for optimisation, *Computers and Chemical Engineering* **28** (2004) 929-941
- Shah, N., Process industry supply chains: Advances and challenges, *Computers and Chemical Engineering* **29** (2005) 1225-1235
- Sparrow, R.E., Forder, G.J., Rippin, D., The choice of equipment sizes for multiproduct batch plants. Heuristics vs. Branch and Bound, *Ind. Eng. Chem. Proc. Des. Dev.* **14** (1975) 197-203
- Subrahmanyam, S., Kudva, G.K., Basset, M.H., Pekny, J.F., Application of distributed computing to batch plant design and scheduling, *AIChE J.* **42** (1996b) 1648-1661
- Subrahmanyam, S., Pekny, J.F., Reklaitis, G., Decomposition approaches to batch plant design and planning, *Ind. Eng. Chem. Res.* **35** (1996a) 1866-1876
- Subrahmanyam, S., Pekny, J.F., Reklaitis, G., Design of batch chemical plants under markets uncertainty, *Ind. Eng. Chem. Res.* **33** (1994) 2688-2701

- Suh, M., Lee, T., Robust optimization method for the economic term in chemical process design and planning, *Ind. Eng. Chem. Res.* **40** (2001) 5950-5959
- Tan, S., Mah, R.S.H. Evolutionary design of noncontinuous plants, *Computers and Chemical Engineering* **22** (1998) 69-85
- Van Roy, T.J., Multi-level production and distribution planning with transportation fleet optimization, *Management Science* **35** (1989) 1443-1453
- Voudouris, V.T., Grossmann, I.E., MILP model for scheduling and design of a special class of multipurpose batch Plants, *Computers and Chemical Engineering* **20** (1996) 1335-1360
- Voudouris, V.T., Grossmann, I.E., Mixed-Integer Linear Programming reformulations for batch processes design with discrete equipment sizes, *Ind. Eng. Chem. Res.* **31** (1992) 1315- 1325
- Voudouris, V.T., Grossmann, I.E., Optimal synthesis of multiproduct batch plants with cyclic scheduling and inventory considerations, *Ind. Eng. Chem. Res.* **32** (1993) 1962-1980
- Wagner, H.M., Whitin, T.M., Dynamic version of the economic lot size model, *Management Science* **5** (1958) 89-96.
- Wellons, H.S., Reklaitis, G.V., The design of multiproduct batch plants under uncertainty with staged expansion, *Computers and Chemical Engineering* **13** (1989) 115-126
- Williams, H.P., *Model Building in Mathematical Programming* (John Wiley & Sons, New York, 2<sup>nd</sup> ed., 1985)
- Wolsey, L.A., Progress with single-item lot-sizing, *European Journal of Operational Research* **86** (1995) 395-401
- Wolsey, L.A., *Integer Programming* (John Wiley & Sons, Inc., New York, 1998)
- Xia, Q., Macchietto, S., Design and synthesis of batch plants - MINLP solution based on a stochastic method, *Computers and Chemical Engineering* **21** Suppl. (1997) S697-S702
- Yi, G., Reklaitis, G.V., Optimal design of batch-storage network with uncertainty and waste treatments, *AIChE J.* **52** (2006) 3473-3490
- Zhu, X.X., Majozi, T., Novel continuous time MILP formulation for multipurpose batch plants. 2. Integrated planning and scheduling, *Ind. Eng. Chem. Res.* **40** (2001) 5621-5634
-



### Ilustração de aplicações de suporte

Ilustram-se diversas aplicações de suporte, de relevância para o desenrolar do processo de investigação, nomeadamente no intuito de caracterizar devidamente e permitir a confirmação dos resultados apresentados. Assim, descreve-se sucintamente cada procedimento em pseudocódigo, de forma a possibilitar a sua correcta utilização, podendo comparar-se valores, quando de interesse.

O desenvolvimento do algoritmo consubstancia a lógica subjacente ao procedimento, pelo que dado o elevado custo associado à programação e manutenção de aplicações computacionais, é conveniente proceder a uma planificação prévia que permita a obtenção de um procedimento eficiente, dotado de uma lógica coerente e compreensível. O algoritmo desenvolvido, consistindo numa sequência lógica de operações (principalmente: manipulação, repetição, selecção ou decisão) e passos direccionados para o tratamento de uma tarefa bem específica, deve possuir as seguintes características:

- Cada passo é explicitado de forma determinística —permite-se a repetição dos processos de cálculo e a obtenção de idênticos resultados;
- O processo deve terminar após um número finito de passos — evitando-se assim que o algoritmo entre em ciclo interminável;
- O algoritmo deve tratar diversas contingências, possuindo alguma generalidade e flexibilidade.

Documentando o código implementado, possibilita-se para cada procedimento ilustrado, uma mais fácil compreensão e reutilização, uniformiza-se a sua manutenção por actualizações e melhorias, bem como se estabelece uma base para o desenvolvimento posterior de outros procedimentos. A documentação dos procedimentos (em programas, subrotinas, funções), comporta usualmente a seguinte informação:

- Designação e objectivo — apresentação do nome e da finalidade de utilização de tal procedimento;
- Modo de utilização — indicação do modo de referenciar o procedimento em questão, referindo a lista de argumentos;
- Argumentos — descrição dos argumentos, sendo normalmente ordenados segundo a sua ocorrência, ou seja, primeiro indicam-se os dados (de entrada, *input*) necessários para a inicialização e execução do procedimento;

habitualmente, indicam-se também os argumentos associados aos resultados (de saída, *output*, ou mistos, *inout*);

- Algoritmo e referências — descrição detalhada do algoritmo e de referências da literatura, onde se encontram aspectos pertinentes ao respectivo desenvolvimento; pode também apresentar-se notas diversas e utilidades de interesse para uma mais eficiente utilização do código, como seja a alocação de espaço de trabalho em memória, ou a formatação de saída de dados e resultados;
- Exemplo (numérico) ilustrativo — aplicação do procedimento, especificando valores, dimensões e tipologia dos dados de entrada; convenientemente, devem ser apresentados os resultados correspondentes.

Por conseguinte, descreve-se em seguida informação e aplicações de suporte, como a descrição das máquinas utilizadas, a determinação de instâncias amostrais e respectivas probabilidades, o cálculo do ciclo de sequenciamento de produções *batch* em campanhas de produtos múltiplos (MPC), ou outras de interesse.

### Descrição das máquinas

Para especificação completa do tratamento computacional, descrevem-se as máquinas utilizadas nos diversos exemplos numéricos. Caso necessário, é possível também discriminar os parâmetros numéricos (como a precisão utilizada, o tipo de arredondamento, a dimensão de memória para os diferentes formatos de números— inteiros curtos e longos, fraccionários em precisão simples ou dupla—, ou outros), para o que se deverá aplicar procedimentos adequados e disponíveis em bibliotecas matemáticas, como a NAG ou IMSL. Porém, aqui apenas interessa discriminar aspectos de ordem mais geral, tendo em vista não só a devida especificação da máquina onde tais resultados foram produzidos, mas também promover a compreensão de diferenças entre tempos de execução resultantes de serem utilizadas máquinas diferentes.

Assim, tomam-se as seguintes designações para as máquinas com as seguintes características:

- Máquina Alfa — servidor DEC Alpha EV5.6, em ambiente UNIX, com dois processadores “alpha EV5.6 (21164A)”, a 400 MHz e 256MB de RAM;
- Máquina Beta — computador pessoal OMIKRON, em ambiente WINDOWS, com processador Intel Pentium a 166MHz e 16MB de RAM;
- Máquina Gama — computador portátil ASUS-F3JC, em ambiente WINDOWS, com duplo processador Intel Core2 T5500 a 1.55GHz e 2GB de RAM.

Para completar a comparação do desempenho das máquinas utilizadas, implementou-se (em Fortran) o cálculo numérico pela “regra dos rectângulos” (melhora a dos “trapézios”) de  $\int_0^{2\pi} |\text{Sen}(x)| dx = 4$ , com um número de partições em progressão geométrica de razão 2, desde  $2^2$  a  $2^6$ . Os tempos obtidos (em segundos de CPU) foram, respectivamente: 50.53; 200.70; 11.59.

### Geração de números aleatórios

A geração de números aleatórios depende fundamentalmente da geração aleatória de números no intervalo  $[0,1]$ , através de geradores de base que utilizam algum método multiplicativo congruente, ou um método de alimentação por componentes da máquina, de forma a obter-se uma sequência cíclica, repetível e cíclica para os valores gerados. Cada subsequência de números aleatórios será especificável, a partir do valor inicial e do tipo de gerador utilizados na produção da sucessão de valores aleatórios.

Usualmente, é possível seleccionar: o tipo de gerador utilizado; a semente (*seed*), cuja especificação torna possível gerar uma mesma amostra; retomar e controlar simulações com diversos fluxos de números aleatórios.

Neste sentido, foi implementado um procedimento *GER\_ALEAT* que, utilizando as funções intrínsecas do Fortran, define duas amostras aleatórias, com distribuição uniforme: uma repetível,  $x_1$ , definida especificando a semente (*seed*); outra, não repetível,  $x_2$ , definida a partir de parâmetros de execução (por ex., o tempo de relógio) da máquina. O único dado necessário é a dimensão (número inteiro) da amostra, *namost*, sendo devolvidas as duas amostras descritas, com valores reais normalizados entre 0 e 1. Assim, dispõe-se de uma ferramenta que permite tanto a repetição como o procedimento inerentemente casuístico.

```
*****
PROCEDIMENTO GER_ALEAT (namost, x1, x2)
  !*** isize = dimensao do vector da semente:
      CALL RANDOM_SEED (SIZE = isize)
  ! amostra com especificacao da semente:
      nseed = 0
      CALL RANDOM_SEED (PUT=nseed(:isize))
      CALL RANDOM_NUMBER (HARVEST=x1)
  ! amostra sem especificacao da semente:
      CALL RANDOM_SEED ()
      CALL RANDOM_NUMBER (HARVEST=x2)
END PROCEDIMENTO GER_ALEAT
***
```

### Amostragem sequencial estratificada, de Hammersley

Além da geração de valores aleatoriamente, também é necessário proceder à amostragem estratificada de valores, os quais descrevam adequadamente o espaço de procura de um programas estocástico, mais concretamente, em termos das múltiplas dimensões envolvidas em cada amostra.

Para obter valores normalizados (entre 0 e 1) para os múltiplos parâmetros incertos, implementa-se o procedimento *HSS2* (*Hammersley Sampling Sequence*, versão 02), desenvolvida a partir do algoritmo descrito por Diwekar e Kalagnanam [1997].

O procedimento *HSS2* difere do procedimento original (implementado na versão *HSS1*), no propósito em que negligencia os valores para a primeira dimensão de cada amostra do espaço de procura, posto que esses valores correspondiam a valores fixos (por ex., para a primeira dimensão incerta da sétima amostra, o valor normalizado corresponderá sempre a  $1/7$ ). Complementa-se, considerando uma dimensão suplementar, a qual corresponderia no procedimento original à ordem “ $kdim+1$ ”, ou seja, procede-se à translação por diminuição de uma unidade na ordem da dimensão.

O procedimento *HSS2* necessita de ser alimentado com os números inteiros referentes ao número de amostras (*namost*) e de dimensões (*kdim*), devolvendo uma matriz de valores reais, normalizados (entre 0 e 1), os quais definem os parâmetros incertos para cada amostra e dimensão. Como procedimentos auxiliares, são utilizadas as funções:

- *iger\_prim* – define os números primos, a utilizar no procedimento posterior para reversão da base; ao procedimento genérico, acrescenta-se um conjunto de dados respeitante aos primeiros 30 números primos e que permite directamente o procedimento *HSS2* para amostras até este limite, sem ter de se repetir continuamente esta função; tem como dado a ordem *i* (valor inteiro) do número primo a determinar, devolvendo esse número primo;
- *rev\_bas* – procede à reversão da base, da qual decorre a determinação dos valores incertos normalizados, sendo aplicado um cuidado tratamento numérico, em termos da precisão envolvida nas funções intrínsecas utilizadas; além da ordem da amostra, número inteiro *n*, é necessário o valor da base *ibas* (um número primo, anteriormente determinado), devolvendo o valor normalizado *z* correspondente ao parâmetro incerto.

```
*****
PROCEDIMENTO HSS2 (kdim, namost, x)
  DO i=1, kdim; ir(i)= iger_prim(i); END DO
  DO n=1, namost
    DO i=1, kdim; z(i) = rev_bas(ir(i), n); END DO
  x(n, :) = 1 - z
  END DO
END PROCEDIMENTO HSS2

*****
PROCEDIMENTO IGER_PRIM(I)
! funcao: numeros primos, excepto "1"
  PARAMETER (np=30)
  INTEGER :: nprim(np) = &
    (/ 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, &
      31, 37, 41, 43, 47, 53, 59, 61, 67, 71, &
      73, 79, 83, 89, 97, 101, 103, 107, 109, 113 /)

  num_prim(:MIN(30,i)) = nprim(:MIN(30,i))

  IF (i > 30) THEN
    nsera_prim = nprim(30)
    DO j=31, i !**** SIEVE
      DO WHILE (nsera_prim .lt. 10000)
        nsera_prim = nsera_prim + 2; irest = 1
        DO k=1, j-1
          IF (num_prim(k) .gt. SQRT(FLOAT(nsera_prim)) ) EXIT
          irest = MOD(nsera_prim, num_prim(k) )
          IF (irest == 0) EXIT
        END DO
        IF (irest .gt. 0) THEN
          num_prim(j) = nsera_prim; EXIT
        END IF
      END DO
    END DO
  END IF
  iger_prim = num_prim(i)
END PROCEDIMENTO IGER_PRIM
```

```

*****
PROCEDIMENTO DOUBLE PRECISION FUNCTION REV_BAS (ibas, n)
! funcao: reversao na base "ibas"
  num_bas = 0; rev_bas = 0.; nx = n
  dou_n = FLOAT(nx); doub = FLOAT(ibas)
  m = NINT( dLOG(dou_n) / dLOG(doub))
  DO j=0, m
    idig = MOD(nx, ibas)
    num_bas = num_bas + idig * ibas**(j)
    rev_bas = rev_bas + idig * FLOAT(ibas)**(-j-1)
    nx = nx / ibas
  END DO
END PROCEDIMENTO REV_BAS

```

\*\*\*

Em seguida, apresentam-se os valores normalizados para uma amostragem estratificada com 15 instâncias amostrais em 5 dimensões incertas, resultantes do procedimento *HSS2*.

```

*****
Parametros da amostragem
#dimensoes, #amostras:          5  15
*** numeros primos:    2  3  5  7  11
Amostra  1
  0.5000  0.6667  0.8000  0.8571  0.9091
Amostra  2
  0.7500  0.3333  0.6000  0.7143  0.8182
Amostra  3
  0.2500  0.8889  0.4000  0.5714  0.7273
Amostra  4
  0.8750  0.5556  0.2000  0.4286  0.6364
Amostra  5
  0.3750  0.2222  0.9600  0.2857  0.5455
Amostra  6
  0.6250  0.7778  0.7600  0.1429  0.4545
Amostra  7
  0.1250  0.4444  0.5600  0.9796  0.3636
Amostra  8
  0.9375  0.1111  0.3600  0.8367  0.2727
Amostra  9
  0.4375  0.9630  0.1600  0.6939  0.1818
Amostra 10
  0.6875  0.6296  0.9200  0.5510  0.0909
Amostra 11
  0.1875  0.2963  0.7200  0.4082  0.9917
Amostra 12
  0.8125  0.8519  0.5200  0.2653  0.9008
Amostra 13
  0.3125  0.5185  0.3200  0.1224  0.8099
Amostra 14
  0.5625  0.1852  0.1200  0.9592  0.7190
Amostra 15
  0.0625  0.7407  0.8800  0.8163  0.6281

```

\*\*\*

### Definição do espaço de procura (cenários discretos e probabilidades)

A partir dos valores normalizados para cada uma das dimensões incertas das instâncias amostrais, pretende definir-se o espaço de procura do programa

estocástico, mais concretamente, deseja-se especificar cada cenário discreto e a probabilidade associada: implementa-se o procedimento *SPACE*.

Como referido, este procedimento utiliza os valores normalizados,  $x_{norm}$ , obtidos pela amostragem estratificada, implementada no procedimento *HSS2*, seguindo-se a translação para a gama prevista de valores,  $x_{trans}$ , por inversão da função cumulativa de densidade (*inverse cumulative density function, icdf*), executada no procedimento *TRANS\_ICDF*.

```
!*****
PROCEDIMENTO SPACE (kdim, namost, ktipo, xmu, sigma, x_trans)
  CALL hss2 (kdim, namost, x_norm)

  DO ja=1, namost
    CALL trans_icdf (kdim, x_norm(ja,:), ktipo, xmu, sigma,
      x_trans(ja,:))
  END DO
END PROCEDIMENTO SPACE
```

\*\*\*

Enquanto que, para proceder à determinação das instâncias normalizadas  $x_{norm}$  em *HSS2* apenas era necessário os números de amostras e de dimensões incertas, respectivamente,  $namost$  e  $kdim$ , para se realizar a translação para os valores previsíveis  $x_{trans}$  já é necessário conhecer: a tipologia para a distribuição de probabilidades aplicável,  $ktipo$ ; os parâmetros definidores da distribuição de interesse, usualmente, os valores esperados,  $xmu$ , o desvio padrão,  $sigma$ , ou outro valor de referência.

Então, aplica-se o procedimento *TRANS\_ICDF* para efectuar a devida translação às amostras de valores normalizados, em conformidade com a distribuição de probabilidades explicitada (tipologia,  $ktipo$ , valor esperado,  $xmu$ , variabilidade,  $sigma$ , outros parâmetros eventuais), obtendo-se o conjunto de instâncias amostrais,  $x_{trans}$ , que o modelo estocástico irá utilizar na construção dos cenários aleatórios.

```
!*****
PROCEDIMENTO TRANS_ICDF (kdim, x_norm, ktipo, xmu, sigma, x_trans)
  DO i=1, kdim
    translacao: SELECT CASE (ktipo(i))
      CASE(1) ! uniforme
        x_lower(i) = xmu(i) - SQRT(12.) * sigma(i) / 2
        x_upper(i) = xmu(i) + SQRT(12.) * sigma(i) / 2
      CASE(2) ! Gauss
        x_lower(i) = xmu(i) - 3 * sigma(i)
        x_upper(i) = xmu(i) + 3 * sigma(i)
    END SELECT translacao
  END DO
  x_trans = x_lower + x_norm * (x_upper - x_lower)
END PROCEDIMENTO TRANS_ICDF
```

\*\*\*

De forma semelhante e aplicando os mesmos parâmetros ( $ktipo$ ,  $xmu$ ,  $sigma$ ) para definir cada função de densidade de distribuição (*probability density function, pdf*), obtem-se o valor  $f$  da probabilidade associada a cada um dos cenários aleatórios  $x$  através do procedimento designado como *pdfvalue*.

```
*****
PROCEDIMENTO PDFVALUE (kdim, x, ktipo, xmu, sigma, f)
  DO i=1, kdim
    distribuicao: SELECT CASE (ktipo(i))
```

```

CASE(1) ! uniforme
  delta = sigma * SQRT (12.0)
  f = 1. / delta
CASE(2) ! Gauss
  f = EXP(-(x-xmu)*(x-xmu)/(2.*sigma*sigma)) / &
    (sigma * SQRT(2.*3.141592653589793))
END SELECT distribuicao
END DO
END PROCEDIMENTO PDFVALUE

```

\*\*\*

As funções de densidade de probabilidade encontram-se na literatura da especialidade, tendo sido consultado Hahn e Shapiro [1967], o qual discrimina diversas situações de aplicação para cada distribuição de probabilidade. A inversão da função cumulativa de densidade exige a aplicação de métodos numéricos elaborados, por exemplo, utilizando a biblioteca matemática NAG.

### Ordenação (*sorting*)

Em numerosos procedimentos, é necessário proceder à ordenação de um conjunto de valores, por exemplo, ordenando os equipamentos por ordem crescente do respectivo custo, ou mesmo seleccionando os estágios por ordem decrescente dos tempos de produção associados. Ora, os procedimentos para ordenação (*sorting*) são bem conhecidos da literatura aberta, sendo usualmente algoritmos polinomiais,  $O(n^2)$ , mas também se encontra algoritmos em  $O(n)$ , para alguns casos específicos.

Contudo, em vez da implementação de um ou vários dos algoritmos conhecidos, optou-se pela alternativa de utilizar iterativamente a função intrínseca que fornece o menor valor (ou localização) de um conjunto de dados. Subsequentemente, ocorrendo essa avaliação em ciclo iterativo, introduz-se como valor limite a considerar a última selecção efectuada (o menor valor, se ordem crescente; o maior, se decrescente), pelo que será seleccionado o valor seguinte na ordem de interesse. Verificou-se que, embora a coincidência de valores fosse pouco provável, quando tal ocorria, era seleccionada a localização de menor ordem.

Exemplificando, em Fortran 90, utiliza-se a função *minval* (ou *minloc*), que fornece o menor valor (localização, isto é, ordem do índice) dos parâmetros dos custos analisados,  $c\_aux(i,isp)$ , inicializando-se o processo com um valor arbitrário, por exemplo, o valor nulo; aplicando-se uma “máscara” (MASK) para definir esse valor limite na procura, *lbound*, bem como definindo a dimensão (DIM) em análise, o valor limite *lbound* vai assumindo em cada ciclo o valor mínimo anterior,  $c1(j)$ , pelo que se obtém uma sucessão de valores estritamente crescente neste parâmetro.

```

*****
!* Custos  $c\_aux(i,isp)$ : seleccionar por ordem crescente em cada estágio  $i$ 

```

```

  lbounb = 0.
  DO i=1,ni
    c1(i) = minval(c_aux(i,:), DIM=1, MASK=c_aux(i,:).GT.lbound)
    lbound = c1(i)
  END DO

```

\*\*\*

Em ambiente GAMS, como se trata de um ambiente dirigido à modelação, não se encontra optimizado para as situações usuais de programação. Donde, são necessários diversos ciclos (LOOPS) integrados, seleccionando-se o valor mínimo  $c1(i)$  de interesse; através do parâmetro auxiliar  $c\_aux2(i,isp)$ , eleva-se o

correspondente custo para um valor muito elevado (BIGM) e selecciona-se o custo mínimo seguinte, numa sucessão estritamente crescente.

```
*****
!* Custos  $c_{aux}(i,isp)$ : seleccionar por ordem crescente em cada estágio  $i$ 
  *iniciar
  BIGM = 999999;
  LOOP(i,
    LOOP(isp,
       $c_{aux2}(i,isp) = c_{aux}(i,isp)$ ;
    );
  );

  *ordenacao
  LOOP(i,
    LOOP(jsp,
       $c1(i) = \text{smin}(isp, c_{aux2}(i,isp))$  ;
      PUT  $c1(i)$ ; PUT /;
      LOOP(isp,
        IF( $c_{aux}(i,isp) EQ c1(i)$ ,
           $c_{aux2}(i,isp) = \text{BIGM}$  ;
        );
      );
    );
  );
  );

***
```

### Formação das campanhas de produção

A formação das campanhas de produção é uma componente indispensável na optimização do sequenciamento do ciclo de produção, em campanhas de produtos múltiplos (MPC) supondo política de não-espera (*zero wait*), pois só neste ponto se assegura a efectividade do cálculo efectuado, mais concretamente, a confirmação da não ocorrência de subciclos. Caso estes ocorram, devem ser incluídas as respeitantes restrições de eliminação (*subtour elimination restrictions*, no âmbito do TSP agregado), obtendo-se nova solução em termos das variáveis de transição de produções,  $Nch(i,k)$ .

Também, embora sejam enunciados os principais critérios a seguir para a formação das campanhas de produção em Birewar e Grossmann [1989a], tal supõe o tratamento manual e sob forma gráfica das soluções obtidas. Neste ponto, promove-se o tratamento automático do ciclo de sequenciamento, permitindo também a execução recorrente do cálculo optimizador em ambiente GAMS.

Descrevem-se as principais regras e passos sequenciais do procedimento, apresentando-se o código respeitante em seguida:

- Passo 1 — tratamento da diagonal da matriz das variáveis de transição, onde se observar  $Nch(i,i) > 0$ ., relativamente às produções consecutivas de um mesmo produto  $i$ ;
- Passo 2 — selecção dos ciclos de produção mais curtos e por ordem crescente, através dos menores  $naux$ ; verifica-se em  $cic\_list1$  se o ciclo de produção contém apenas dois produtos (lista com dois elementos), ou acrescentam-se mais produtos (elementos) em caso contrário; aqui, verifica-se se a adição de um novo produto cria algum subciclo, actualizando-se então a lista associada;

- Passo 3 — formação das campanhas, efectuando as listas respectivas em *cic\_list2*, bem como diminuindo sucessivamente em *cic\_list3* os valores das transições  $Nch(i,k)$  incluídas nas campanhas formadas.

```

*****
* cysched_camp.inc (MPC1)
* formacao de campanhas: verificar formacao de subciclos!

Nch0(i,k) = Nch.l(i,k); /* valores obtidos da optimização */

ALIAS (klist,klist1) ; ALIAS (klist,klist2) ;
ALIAS (i,ii); ALIAS (k,kk);

Parameters icyc(icamp, klist)
           icyc_glob(klist)
           ilist(icamp)
           ncy_val(icamp);

*****PASSO 1
           /* Diagonal da Matriz Nch */
icyc(icamp, klist) = 0;
ncamp = 0;
LOOP(i,
  n_aux = Nch0(i,i);
  IF(n_aux NE 0.,
    ncamp = ncamp + 1;
    ilist(icamp) $(ord(icamp) EQ ncamp) = n_aux;
    LOOP(klist $(ord(klist) LE n_aux),
      icyc(icamp,klist) $(ord(icamp) EQ ncamp) = ord(i);
    );
    Nch0(i,i)= 0.;
  );

); /*end LOOP*/

Nch1(i,k) = Nch0(i,k);

*****PASSO 2
*cic_camp1
ncamp1 = ncamp;
LOOP(icamp $(ord(icamp) GT ncamp),
IF( sum((i,k), Nch0(i,k)) GT 0.1,
  ncamp1 = ncamp1 + 1;
           /* inicializar lista da campanha */
  LOOP(i,
    LOOP(k,
      IF(Nch1(i,k) Eq 0.,
        Nch1(i,k) = 1000000.
      );
    );
  );

  n_aux = smin((i,k), Nch1(i,k) );
  LOOP(i,
    LOOP(k,
      IF(Nch0(i,k) EQ n_aux,
        iloc = ord(i);
        kloc = ord(k);

```

```

    );
  );
);
ncyc_val(icamp) = n_aux;

ilist(icamp) = 1; icyc(icamp,"1") = iloc;
ilist(icamp) = 2; icyc(icamp,"2") = kloc;

*****cic_list1
iflag_camp = 0;
LOOP(klist1 $(ord(klist1) GE 3),
IF(iflag_camp EQ 0,

iflag = 0;
LOOP(i,
LOOP(k,

/* lista com apenas 2 elementos? */
IF(iflag EQ 0,
IF(Nch0(i,k) $(ord(i) EQ kloc) AND (ord(k) EQ iloc)) GT 0.,
n_aux = Nch0(i,k) $(ord(i) EQ kloc) AND (ord(k) EQ iloc));
IF(n_aux LT ncyc_val(icamp),
ncyc_val(icamp) = n_aux;
);
iflag = 1;
)); /*end 2 if*/

/* acrescentar novo elemento na lista! */
IF(iflag EQ 0,
IF(ord(i) EQ kloc,
n_aux = smin((ii,kk)$ord(ii) EQ kloc), Nch1(ii,kk) );
IF(Nch0(i,k) $(ord(i) EQ kloc) EQ n_aux,
kloc2 = ord(k) ;
ilist(icamp) = ilist(icamp) + 1;
icyc(icamp,klist1) = kloc2;
IF(n_aux LT ncyc_val(icamp),
ncyc_val(icamp) = n_aux;
);
kloc = kloc2;
iflag = 2;
)); /*end 3 if*/

)); /*end 2 LOOP*/

/* novo elemento cria subciclo? >> IFLAG=3; IFLAGcamp=1*/
IF(iflag EQ 2,
LOOP(klist2 $(ord(klist2) LT ord(klist1)),

k_aux = ord(klist1)- ord(klist2);
LOOP(klist,
l_aux $(ord(klist) EQ k_aux) = icyc(icamp,klist);
);
IF(iflag EQ 2,
LOOP(i,
LOOP(k,
IF(NCh0(i,k)$((ord(i) EQ kloc) AND (ord(k) EQ l_aux)) GT 0.,
iflag = 3;
iflag_camp = 1;

```

```

        klist_ini = k_aux;
    );
    ); /*end 2 LOOP*/
    ); /*end IF*/
); /*end LOOP*/
); /*end IF*/
/*atualizacao da lista com subciclo >> IFLAG = 4*/
IF((iflag EQ 3) AND (klist_ini GT 1),
    k_aux = 0;
    LOOP(klist2 $(ord(klist2) GE klist_ini AND (ord(klist2) LE ord(klist1))),
        k_aux = k_aux + 1;
        icyc(icamp,klist) $(ord(klist) EQ k_aux) = icyc(icamp,klist2);
    );
    icyc(icamp,klist) $(ord(klist) GT k_aux) = 0;
    ilit(icamp) = k_aux;
    ncyc_val(icamp) = 1000000.
    LOOP(klist2 $(ord(klist2) LE k_aux),
        iloc = icyc(icamp,klist2);
        kloc = icyc(icamp,klist2+1);
        IF(ord(klist2) EQ k_aux,
            kloc = icyc(icamp,"1");
        );
        LOOP(i,
            LOOP(k,
                n_aux$(ord(i) EQ iloc AND (ord(k) EQ kloc)) = Nch0(i,k);
            )); /*end 2 LOOP*/
        IF(n_aux LT ncyc_val(icamp),
            ncyc_val(icamp) = n_aux;
        );
    ); /*end LOOP*/
    iflag = 4;
); /*end IF*/ /*END atualizacao da lista com subciclo*/
); /* end "big" IF IFLAGcamp */
); /* end "big" LOOP klist */
*****end cic_list1

***** PASSO 3
* listagem da campanha
    n_aux = ncyc_val(icamp) - 1.;
    k_aux = ilit(icamp);

*****cic_list2: formacao da campanha
    LOOP(klist1 $(ord(klist1) LE n_aux),
        LOOP(klist2 $(ord(klist2) LE k_aux),
            icyc(icamp,klist) $(ord(klist) EQ (k_aux*ord(klist1)+ord(klist2))) = icyc(icamp,klist2);
        ));
*****end cic_list2
* diminuir os NCh incluídos na campanha
*****cic_list3
    LOOP(klist1 $(ord(klist1) LE k_aux),
        iloc = icyc(icamp, klist1);
        kloc = icyc(icamp, klist1 + 1);
        IF(ord(klist1) EQ k_aux,
            kloc = icyc(icamp,"1");
        );
        LOOP(i,
            LOOP(k,
                Nch0(i,k) $(ord(i) EQ iloc AND (ord(k) EQ kloc)) = Nch0(i,k) - ncyc_val(icamp);
            );
        );
    );

```

```

Nch1(i,k) $((ord(i) EQ iloc) AND (ord(k) EQ kloc)) = Nch1(i,k) - ncyc_val(icamp);
));
);
*****end cic_list3
); /*end if*/
); /*end cic_camp1*/
display icyc;
***

```

A inclusão do código do procedimento em ambiente GAMS tem o propósito de ilustrar devidamente a dificuldade inerente à sua programação, pois este ambiente dirige-se à modelação matemática, não estando pois optimizado para uma programação que: utilize ciclos condicionais, sendo aqui necessário a constante codificação utilizando o símbolo \$; a saída de ciclos ou condições, que implica o recurso exaustivo à definição de *flags*; a definição directa dos parâmetros ou variáveis através dos respectivos índices, não se tendo de especificar constantemente a ordem do índice.

Apresenta-se um exemplo numérico da aplicação do procedimento descrito, descrevendo: primeiro, a matriz das transições  $Nch(i,k)$ , obtida por optimização do TSP agregado; depois, os parâmetros  $ICYC(icamp,klist)$ , indicam para cada ciclo da campanha de produção, a lista sequencial de produtos (por ordem do índice do produto). Isto é, haverá: um primeiro ciclo, com 2 batches do terceiro produto (C); um quarto ciclo, com o sequenciamento dos produtos de ordens 6,4,2 (ou seja, produz-se sequencialmente FDB); um derradeiro ciclo, o mais longo, com os produtos de ordens 6,1,3,4,2 (isto é, sucessivamente FACDB).

```

*****
---- 170 VARIABLE NCH.L (integer) variable
      A          B          C          D          E          F
A          1.000      1.000          1.000          2.000
B          3.000          4.000
C          2.000      1.000
D          5.000
E          1.000          3.000
F          5.000          1.000

---- 375 PARAMETER ICYC
      1          2          3          4          5          6
1          C          C
2          E          E          E
3          D          B          D          B          D          B
4          F          D          B
5          F          A          A
6          B          F          A
7          E          B          F          A
8          F          A          C          D          B
***

```

■

# Heurísticas: desenvolvimento, pseudocódigo

Detalha-se aqui a análise efectuada e descrevem-se os procedimentos heurísticos de aproximação que foram desenvolvidos no âmbito dos modelos para dimensionamento e sequenciamento de processos descontínuos. E complementa-se com o pseudocódigo respeitante. Também, relembram-se os modelos em estudo:

- *batch\_milp\_fssz* (*Flowshop/ Single machine/ SPC/ ZW*) — trata o dimensionamento óptimo (dimensões discretas) da unidade multiproducto (*flowshop*), de estágios com processo único (*single machine*), operando em campanha de produto único (*single product campaign*, SPC) e em política de espera nula (*zero wait*);
- *batch\_milp\_fmsz* (*Flowshop/ Multiple machine/ SPC/ ZW*) — similar ao modelo anterior, mas generaliza o número de processos em cada estágio (*multiple machine*);
- *batch\_milp\_fsmz* (*Flowshop/ Single machine/ MPC/ ZW*) — similar ao primeiro modelo referido, mas generaliza o modo de operação para campanhas com produtos múltiplos (*multiple product campaign*, MPC);
- *batch\_milp\_fmmz* (*Flowshop/ Multiple machine/ MPC/ ZW*) — similar ao modelo anterior (com MPC), generaliza também o número de processos em cada estágio (*multiple machine*).

### Modelo *batch\_milp\_fssz*

Embora não se proceda a um estudo formal da complexidade computacional deste modelo, verifica-se apenas empiricamente a ocorrência de combinatoriedade nas soluções binárias. Alguns aspectos a considerar:

- da restrição sobre o número (unitário) de dimensões seleccionadas em cada estágio, resulta que apenas se poderá seleccionar uma dimensão  $s'(j)$ , em qualquer dos estágios  $j$ ,

$$4A.1 \quad \begin{cases} y_{js} = 0, & \forall s \neq s'(j) \\ y_{j,s'(j)} = 1, & \exists! s'(j) \end{cases}$$

- observa-se que na definição do tempo de campanha  $t_i$  (para qualquer produto  $i$ ) devem satisfazer-se todos os estágios  $j$ ; assim, o valor de  $t_i$  corresponde ao máximo dos valores obtidos segundo o índice  $j$ , ou seja,

$$4A.2 \quad t_i = \max_j \left\{ S_{ij} Q_i CT_i \sum_{s=1}^{NS(j)} \frac{y_{js}}{dv_{js}} \right\} = \max_j \left\{ \frac{S_{ij} Q_i CT_i}{dv_{j,s'(j)}} \right\}, \quad \forall i$$

- verifica-se que a soma dos valores dos tempos (máximos) de campanha  $t_i$  deve observar o horizonte temporal,  $H$ ,

$$4A.3 \quad \sum_{i=1}^N \max_j \left\{ S_{ij} Q_i CT_i \sum_{s=1}^{NS(j)} \frac{y_{js}}{dv_{js}} \right\} \leq H$$

- da satisfação simultânea destes constrangimentos constrói-se um procedimento, auxiliar e recorrente, para verificação da solução binária proposta (VERIFICA\_SOLUCAO\_fssz);
- o objectivo do modelo é o de minimizar o custo de investimento total, pelo que deve ser escolhido o equipamento de menor dimensão (custo) que satisfaz as restrições, em cada estágio  $j$  e para cada produto  $i$ .

Sequencialmente, descrevem-se os procedimentos de aproximação implementados para estimativa da solução óptima de *batch\_milp\_fssz*.

#### PREVIO\_fssz: admissibilidade

Colocou-se um procedimento prévio, PREVIO\_fssz, para garantir a existência de solução admissível para o modelo. Para tal, supõe-se que devem ser satisfeitas as restrições temporais com os equipamentos de maiores dimensões (fazendo  $s' = NS$ ) em todos os estágios  $j$ . Se tal não se verificar, o problema considerando um só processo por estágio não apresenta qualquer solução admissível.

#### APROX0\_fssz: aproximação inicial

A aproximação baseia-se na suposição de que a soma dos tempos de campanha  $t_i$  também terá de observar o horizonte  $H$  em cada estágio  $j$ ; ou seja, não é possível que o tempo total necessário para a produção ultrapasse o horizonte  $H$  em algum dos estágios; assim, pretende-se estimar o volume mínimo necessário  $V_{h0}$  para cada estágio  $j$ , evitando a respectiva enumeração,

$$4A.4 \quad H = \sum_{i=1}^N \frac{S_{ij} Q_i CT_i}{V_{h0}(j)}, \quad \forall j \Rightarrow V_{h0}(j) = \sum_{i=1}^N \frac{S_{ij} Q_i CT_i}{H}, \quad \forall j$$

O volume assim estimado é adequado ao volume discreto imediatamente superior, especificando-se valor unitário para as respeitantes variáveis binárias,  $iy_{h0}(j,s'(j)) = 1$ ; subsequentemente, verifica-se a solução obtida desta forma, no procedimento VERIFICA\_SOLUCAO\_fssz.

#### APROX1\_fssz: construção de solução admissível

Utilizam-se diferentes procedimentos para a construção de soluções admissíveis:

- APROX1A\_fssz – afectar tempos de produção  $t_i$  de forma proporcional às quantidades procuradas  $Q_i$ ;
- APROX1B\_fssz – desde uma solução não admissível (por ex., de APROX0), reduzir tempos de produção associando-lhes os custos respeitantes, no âmbito duma procura local iterada (*iterated local search*).

#### Procedimento APROX1A\_fssz

Neste procedimento supõe-se que o tempo de campanha de cada um dos produtos  $t_i$  é proporcional à respectiva quantidade procurada  $Q_i$ , sendo o horizonte temporal  $H$  disponível distribuído pelos diferentes produtos  $i$ , segundo a proporcionalidade entre as quantidades procuradas ( $Q_i/Q_{total}$ ):

$$4A.5 \quad \begin{cases} Q_{total} = \sum_{i=1}^N Q_i \\ t_{total} = \sum_{i=1}^N t_i \leq H \end{cases}$$

Dado que apenas se minimiza o custo de investimento total, não é estabelecida qualquer preferência entre os diferentes produtos (a qual poderia advir caso se considerassem preços, lucros, disponibilidades, ou outras especificações relativas aos produtos, o que não acontece); então,

$$4A.6 \quad \frac{t_i}{t_{total}} = \frac{Q_i}{Q_{total}} \Rightarrow \frac{t_i}{Q_i} = \frac{t_{total}}{Q_{total}} \leq \frac{H}{Q_{total}}$$

uma solução assim obtida é admissível, pois,

$$4A.7 \quad t_i \leq Q_i \frac{H}{Q_{total}} \Rightarrow \sum_{i=1}^N t_i \leq \sum_{i=1}^N \left( Q_i \frac{H}{Q_{total}} \right) \leq H$$

O procedimento analisa sequencialmente todos os produtos  $i$ , para os quais vai seleccionando o menor volume discreto  $s'(j)$ , em cada estágio  $j$ , que verifique a suposição de proporcionalidade:

$$4A.8 \quad \frac{t_i}{Q_i} = \max_j \left\{ \frac{S_{ij} CT_i}{dv_{j,s'(j)}} \right\} \leq \frac{H}{Q_{total}}, \quad \forall i$$

Saliente-se que poderá ocorrer uma situação de impossibilidade neste procedimento de construção de solução, precisamente se existir uma tal desproporção entre os parâmetros do problema que se verifique, para algum produto  $i$  em algum estágio  $j$ , mesmo considerando aí a dimensão máxima  $NS(j)$ ,

$$4A.9 \quad \max_j \left\{ \frac{S_{ij} CT_i}{dv_{j,NS(j)}} \right\} > \frac{H}{Q_{total}}, \quad \exists(i, j)$$

#### Procedimento APROX1B\_fssz

A abordagem APROX1B\_fssz inicia-se com uma solução inadmissível (por ex., proveniente de APROX0\_fssz), pretendendo-se reduzir os tempos de produção  $t_i$  para satisfação do horizonte temporal  $H$ , mas conduzindo-se essa redução em conformidade com os custos associados,  $c(j,s)$ . Esta redução de tempos é originada

por uma alteração (aumento) da dimensão do equipamento num dado estágio, alteração essa seleccionada segundo a melhor razão (máxima) entre as variações provocadas nos tempos e nos custos respectivos.

Enquanto que o cálculo dessa variação nos custos é directa, na estimativa da variação dos tempos considera-se que: o valor do tempo de produção de um dado produto  $i$  é definido pelo máximo dos respectivos valores calculados nos diferentes estágios  $j$ , o qual se designa pelo “estágio crítico”,  $jcrit1(i)$ ; comprova-se que o valor máximo dos tempos se encontra associado com o menor valor para a dimensão dos lotes (*batches*) de produção,  $B_i$ ; aumentando a dimensão deste “estágio crítico”, altera-se também a especificação do “estágio crítico” para um outro estágio,  $jcrit2(i)$ , definido pelo valor seguinte na sucessão da diminuição dos tempos ou do aumento da dimensão dos *batches*. A variação total dos tempos devido ao aumento da dimensão num dado “estágio crítico” é então definida pela soma de todas as variações parcelares no tempo de campanha dos diferentes produtos  $i$ , sendo essas variações parcelares estimadas atendendo à amplitude da variação das dimensões dos *batches*, de  $Bcrit1(i)$  para  $Bcrit2(i)$ . Ora, é conhecido que,

$$4A.10 \quad t_i = \max_j \left\{ \frac{S_{ij} Q_i CT_i}{dv_{j,s'(j)}} \right\} = Q_i CT_i \max_j \left\{ \frac{S_{ij}}{dv_{j,s'(j)}} \right\} = n_i CT_i, \quad \forall i$$

e como,

$$4A.11 \quad n_i = Q_i \max_j \left\{ \frac{S_{ij}}{dv_{j,s'(j)}} \right\} = \frac{Q_i}{B_i}, \quad \forall i \Rightarrow B_i = \min_j \left\{ \frac{dv_{j,s'(j)}}{S_{ij}} \right\}, \quad \forall i$$

pode determinar-se a dimensão do *batch*  $B_i$ , associada ao tempo  $t_i$  ou número de *batches*  $n_i$ , sendo os seus valores definidos segundo o “estágio crítico”.

Quando se procede à alteração por aumento da dimensão num “estágio crítico”,  $jcrit1(i)$ , ao qual se encontram associadas as variáveis de tempos  $tcrit1(i)$  e dimensão dos *batches*  $Bcrit1(i)$ , vão obviamente ocorrer modificações nessas variáveis, as quais se alteram para  $tcrit2(i)$  e  $Bcrit2(i)$ , pois serão definidas por um outro estágio  $jcrit2(i)$ . É então possível relacionar a variação nos tempos através de:

$$4A.12 \quad \begin{aligned} \Delta t_i &= tcrit1_i - tcrit2_i \\ &= \frac{Q_i CT_i}{Bcrit1_i} - \frac{Q_i CT_i}{Bcrit2_i} = \frac{Q_i CT_i}{Bcrit1_i} - \frac{Q_i CT_i}{Bcrit1_i} \left( \frac{Bcrit1_i}{Bcrit2_i} \right) \\ &= \frac{Q_i CT_i}{Bcrit1_i} \left( 1 - \frac{Bcrit1_i}{Bcrit2_i} \right) = tcrit1_i \left( 1 - \frac{Bcrit1_i}{Bcrit2_i} \right), \quad \forall i \end{aligned}$$

Por conseguinte, sendo conhecida uma solução inicial inadmissível, à qual se propõe uma alteração por aumento de dimensão num dado estágio, determina-se a totalidade das variações nos tempos que ocorrem, compara-se com a variação nos custos de equipamento e selecciona-se a melhor razão entre essas variações, alterando-se as variáveis binárias respectivas (*swap*: no estágio seleccionado, a variável binária com valor unitário anula-se, tomando valor unitário a variável binária associada à dimensão imediatamente superior). A solução assim obtida é então verificada no procedimento VERIFICA\_SOLUCAO\_fssz, para se concluir acerca da sua admissibilidade (*feasibility*).

O procedimento APROX1B\_fssz pode ser executado iterativamente, até se obter solução admissível. Por conseguinte, este procedimento pode-se constituir como um método de procura local iterada (*iterated local search*, ILS), onde:

- a vizinhança de procura é definida através do aumento de dimensão em cada “estágio crítico”;
- a função de avaliação corresponde à razão entre as variações estimadas nos tempos e nos custos;
- a alteração processa-se segundo a solução vizinha com melhor (máximo) valor da referida razão;
- o procedimento termina quando se encontra uma solução admissível, ou se atinge um número limite de iterações.

### ALGORITMO heur\_fssz

#### PREVIO\_fssz: admissibilidade

!\* calcular tempos das campanhas para o volume *máximo(NS)* em cada estágio:

$$\text{tempos\_h}(i,j) = \text{SF}(i,j) * q(i) * \text{ct}(i) / \text{dv}(j, \text{NS})$$

!\* determinar tempos de campanha para cada produto (i):

$$t\_h(i) = \text{MAXVAL}(\text{tempos\_h}(i,j), \text{DIM}=2)$$

!\* calcular a soma dos tempos de campanha

$$\text{somatempos} = \text{SUM}(t\_h)$$

!\* Admissibilidade caso se observe o horizonte:

```
IF (somatempos > H) THEN
    IFEASfssz=1; WRITE (“ERRO: fssz inadmissível”); STOP
ELSE
    GOTO APROX0_fssz
END IF
```

#### END PREVIO\_fssz

\*\*\*\*\*

#### APROX0\_fssz: aproximação inicial

!\* iniciar valores binarios

$$iy\_h0 = 0$$

!\* estimativa do volume necessário em cada estágio

```
cic_estag: DO j=1,nj
    soma = 0.
    DO i=1,ni; soma = soma + SF(i,j)*q(i)*ct(i); END DO
    vol_h0(j) = soma / H
    is_loc = MINLOC (dv(j,:), MASK= dv(j,:) > vol_h0(j))
    iy_h0 (j,is_loc(1)) = 1
END DO cic_estag
```

!\* Admissibilidade da aproximação inicial?

```
CALL VERIFICA_SOLUCAO_fssz
```

```
IF (IERRO == 1) THEN IFLAG0 = 1; ELSE IFLAG0 = 0; END IF
```

```
END APROX0_fssz
```

```
*****
```

```
APROX1_fssz: construção de solução admissível
```

```
!* procedimento APROX1A_fssz: tempos proporcionais as procuras
```

```
CALL APROX1A_fssz
```

```
IF (IFLAG1A == 0) THEN
```

```
    iy_h1 = iy_h1A; t_h1 = t_h1A; cost_h1 = cost_h1A
```

```
END IF
```

```
!* procedimento APROX1B_fssz: redução de tempos (iterada) dirigida pelos custos
```

```
IFLAG1B = 1; icount = 1
```

```
iy_h1B0 = iy_h0; t_h1B0 = t_h0; iscrit_h1B0 = is_crit0
```

```
njs = nj * ns
```

```
DO WHILE (IFLAG1B == 1 .AND. icount <= njs)
```

```
    CALL APROX1B_fssz
```

```
    icount = icount + 1
```

```
    iy_h1B0 = iy_h1B; t_h1B0 = t_h1B; iscrit_h1B0 = iscrit_h1B
```

```
END DO
```

```
IF (IFLAG1B == 0 .AND. cost_h1B < cost_h1) THEN
```

```
    iy_h1 = iy_h1B; t_h1 = t_h1B; cost_h1 = cost_h1B
```

```
END IF
```

```
END APROX1_fssz
```

```
*****
```

```
APROX1A_fssz – construção de solução admissível afectando tempos de  
produção  $t_i$  de forma proporcional às quantidades procuradas  $Q_i$ 
```

```
!* 1) inicialização
```

```
    is_crit = is_crit0 !dimensões dadas pela APROX0
```

```
    razao_t = H / SUM(i, Q(i))
```

```
!* 2) ciclos para selecção da dimensão
```

```
cic_prod: DO i=1, ni; cic_estag: DO j=1,nj; cic_size: DO is=is_crit0(j),ns
```

```
    razao_i = ct(i) * SF(i,j) / dv(j,is)
```

```
    IF ( razao_i <= razao_t) THEN
```

```
        IF (is > is_crit(j)) THEN is_crit(j) = is; END IF; EXIT cic_size
```

```
    END IF
```

```
    IF (is == ns) THEN is_crit(j) = ns; END IF
```

```
END DO cic_size; END DO cic_estag; END DO cic_prod
```

```
!* 3) construir solução binária
```

```
    iy_h1A = 0; DO j=1,nj; iy_h1A(j,is_crit(j))=1; END DO
```

```
!* 4) verificar solução
```

```
CALL VERIFICA_SOLUCAO_fssz
```

```
IF (IERRO == 1) THEN IFLAG1A = 1; ELSE IFLAG1A = 0; END IF
```

```
END APROX1A_fssz
```

```
*****
```

### APROX1B\_fssz – construção de solução admissível reduzindo tempos de produção associando os custos respeitantes

```

!* 1) inicializacao
      IFLAG1B = 1; iy_h1B = iy_h0; iscrit_h1B = iscrit0
      DO j=1,nj; Vaux(:,j) = dv(j,iscrit0(j)); END DO  !Volume auxiliar
      Baux = Vaux / SF      ! dimensão do Batch (auxiliar)

!* 2) estagios criticos:
!*   jcrit1(i), o que define B(i) (ou seja, Bcrit1(i))
!*   jcrit2(i): o valor seguinte, quando a dimensão aumenta
      jcrit1 = MINLOC (Baux, DIM=2)
      DO i=1,ni; Bcrit1(i) = Vaux(i,jcrit1(i)) / SF(i,jcrit1(i)); END DO

!* determinar Bcrit2
      DO i=1,ni
        jcrit2(i) = MINLOC(Baux(i,:), DIM=1, MASK = Baux(i,:) > Bcrit1(i))
        Bcrit2(i) = Vaux(i,jcrit2(i)) / SF(i,jcrit2(i))
      END DO
      DO i=1,ni
        IF(iscrit0(jcrit1(i)) == ns) THEN
          Bcrit2(i) = Bcrit1(i)
        ELSE
          !* Bprov: Batch provisório, devido a aumento da dimensão
          Bprov(i) = dv(jcrit1(i), iscrit0(jcrit1(i)) + 1) / SF(i,jcrit1(i))
          IF(Bcrit2(i) > Bprov(i)) THEN Bcrit2(i) = Bprov(i); END IF
        END IF
      END DO

!* 3) razao_tc: razão entre a variação nos tempos e nos custos
      DO i=1,ni
        IF(iscrit0(jcrit1(i)) == ns) THEN
          delta_c(i) = 1.D10
        ELSE
          delta_c(i) = c(jcrit1(i),iscrit0(jcrit1(i)) + 1) - &
            c(jcrit1(i),iscrit0(jcrit1(i)))
        END IF
      END DO
      delta_t = t_h0 * (1. - Bcrit1 / Bcrit2)
      soma_t = 0.
      DO k=1,ni; DO l=1,ni
        IF(jcrit1(k) == jcrit1(l)) THEN soma_t(k) = soma_t(k)+delta_t(l); END IF
      END DO; END DO
      razao_tc = soma_t / delta_c

!* 4) alteração da solução (binária): aumento de dimensão
      ubound = 1.D10
      cic_prod: DO k=1,ni
        ibest_tc = MAXLOC (razao_tc, MASK = razao_tc < ubound)
        ubound = razao_tc(ibest_tc(1))
        jaux = jcrit1(ibest_tc(1))

```

```

    IF(iscrit0(jaux) < ns) THEN
        iy_h1B(jaux, iscrit0(jaux) + 1) = 1; iy_h1B(jaux, iscrit0(jaux) ) = 0
        iscrit_h1B(jaux) = iscrit0(jaux) + 1; EXIT cic_prod
    END IF
END DO cic_prod

!* 5) verificar solucao
CALL VERIFICA_SOLUCAO_fssz
IF (IERRO == 0) THEN IFLAG1B = 0; END IF

```

**END APROX1B\_fssz**

\*\*\*\*\*

**VERIFICA\_SOLUCAO\_fssz: os valores propostos  $iy_h(j,s'(j))$  são admissíveis?**

```

!* calculo dos tempos das campanhas
som_inv_vol = SUM(iy_h / dv, DIM=2)
DO i=1,ni; DO j=1,nj
    tempos_h(i,j) = SF(i,j)*q(i)*ct(i)*som_inv_vol(j)
END DO; END DO

! tempos de campanha para cada produto (i)
t_h = MAXVAL (tempos_h, DIM=2)

! verificar horizonte temporal
somatempos = SUM (t_h)
IF (somatempos > H) THEN IERRO = 1; ELSE IERRO = 0; END IF

!* custo total da solucao actual
cost_actual = SUM (c * iy_h)

```

**END VERIFICA\_SOLUCAO\_fssz**

\*\*\*\*\*

### Modelo *batch\_milp\_fmsz*

Vai agora comparar-se a modelação descrita no âmbito de *multiple machine* com a anterior, de *single machine* (em *batch\_milp\_fssz*), assumindo uma instância específica em que o número máximo de processos em cada estágio se reduza a um só processo.

Alguns aspectos a considerar quanto a este modelo, *batch\_milp\_fmsz*:

- da restrição relativa à selecção dos processos, apenas se poderá seleccionar uma dimensão  $s'(j)$  em número  $p'(j)$ , em qualquer dos estágios  $j$ ;

$$4A.13 \quad \begin{cases} y_{jsp} = 0, & \forall s \neq s'(j), p \neq p'(j) \\ y_{j,s'(j),p'(j)} = 1, & \exists^1(s'(j), p'(j)) \end{cases}$$

- observa-se que na definição do número de *batches*  $n_i$  (para qualquer produto  $i$ ) devem satisfazer-se todos os estágios  $j$ ; assim, o valor de  $n_i$  corresponderá ao máximo dos valores obtidos segundo o índice  $j$ ,

$$4A.14 \quad n_i = \max_j \left\{ S_{ij} Q_i \sum_{s=1}^{NS(j)} \sum_{p=1}^{NP(j)} \frac{y_{jsp}}{dv_{js}} \right\} = \max_j \left\{ \frac{S_{ij} Q_i}{dv_{j,s'(j)}} \right\}, \quad \forall i$$

- da restrição de carácter lógico sobre o número de *batches*, observa-se que todas as variáveis  $nc_{ijsp}$  serão nulas, excepto aquelas correspondentes à dimensão  $s'$  e número  $p'$  seleccionados, que igualarão o valor de  $n_i$  em todos os estágios  $j$  (aqui, supõe-se que o limite superior lógico da restrição seja trivialmente verificado, desde que este limite seja suficientemente amplo):

$$4A.15 \quad \begin{cases} nc_{ijsp} = 0, & \forall s \neq s'(j), p \neq p'(j) \\ nc_{ij,s'(j),p'(j)} = n_i, & \exists!(s'(j), p'(j)) \end{cases}$$

- atente-se que o tempo de campanha  $t_i$  para cada produto  $i$  deve satisfazer todo o estágio  $j$ , pelo que corresponderá ao máximo dos valores obtidos para o somatório; neste somatório, os números de ciclos  $nc_{ijsp}$  encontram-se restringidos a  $n_i$  em todo o estágio  $j$ , conforme 4A.15, pelo que

$$4A.16 \quad t_i = \max_j \left\{ \frac{\tau_{ij}}{p'(j)} n_i \right\} = \max_j \left\{ \frac{\tau_{ij}}{p'(j)} \max_j \left\{ \frac{S_{ij} Q_i}{dv_{j,s'(j)}} \right\} \right\} = \max_j \left\{ \frac{\tau_{ij} S_{ij} Q_i}{p'(j) dv_{j,s'(j)}} \right\}, \quad \forall i$$

- também, a soma dos valores dos tempos de campanha  $t_i$  deve observar o horizonte temporal  $H$ , pelo que,

$$4A.17 \quad \sum_{i=1}^N \max_j \left\{ \frac{\tau_{ij} S_{ij} Q_i}{p'(j) dv_{j,s'(j)}} \right\} \leq H$$

- o objectivo é o de minimizar o custo total, pelo que deve ser escolhido o grupo de equipamentos  $(s', p')$  ao menor custo que satisfaz as restrições, em cada estágio  $j$  e para cada produto  $i$ .

Seja a instância do problema em *batch\_milp\_fmssz*, onde se supõe que:

- o número de processos em cada estágio  $NP(j) = 1$ ; deste modo, o índice correspondente  $p$  é redundante e pode desprezar-se;
- os limites superiores aos números de batches não constituem restrições activas,  $nc_{ijsp}^{Upp} = bigbnd, \quad \forall i, j, s, p$ ;
- todos os restantes parâmetros e dados coincidem com os correspondentes do modelo *batch\_milp\_fssz*.

Observa-se que a função objectivo, desprezando-se o índice redundante  $p$ , coincide com a de *batch\_milp\_fssz*,

$$4A.18 \quad [\min] \quad z = \sum_{j=1}^M \sum_{s=1}^{NS(j)} \sum_{p=1}^{NP(j)} c_{jsp} y_{jsp} \Rightarrow \sum_{j=1}^M \sum_{s=1}^{NS(j)} c_{js} y_{js}$$

Pela mesma razão, são coincidentes as restrições quanto à selecção de um só processo por estágio,

$$4A.19 \quad \sum_{s=1}^{NS(j)} \sum_{p=1}^{NP(j)} y_{jsp} = 1 \Rightarrow \sum_{s=1}^{NS(j)} y_{js} = 1, \quad \forall j$$

Atendendo às relações de 4A.14 a 4A.16, observa-se que

$$4A.20 \quad t_i = \max_j \left\{ \frac{\tau_{ij}}{p'(j)} n_i \right\} = CT_i n_i, \quad p'(j) = 1, \quad \forall i$$

Donde, a definição dos tempos de campanha em 4A.16 coincide com a descrita em 4A.2 para o modelo em *single machine*. Finalmente, também coincide nos dois modelos a conhecida restrição quanto ao horizonte temporal, como se confere da comparação das relações em 4A.17 com as de 4A.3.

Verifica-se equivalência entre as funções objectivo e os espaços de procura, pelo que se pode afirmar que a instância específica do problema em *batch\_milp\_fmsz* é equivalente ao problema em *batch\_milp\_fssz*. Também é possível comprovar a correspondência numérica das variáveis correspondentes.

Em seguida, descrevem-se os procedimentos heurísticos implementados para aproximação da solução do presente problema, *batch\_milp\_fmsz*.

#### **PREVIO\_fmsz: admissibilidade**

Implementou-se um procedimento prévio, para garantir a existência de solução admissível. Para tal, supõe-se que devem ser satisfeitas as restrições temporais com os equipamentos de máxima dimensão,  $dv(j,NS)$ , e em número máximo,  $NP(j)$ , em cada estágio  $j$ . Se tal não se verificar, o problema considerado no enquadramento de processos múltiplos deve ser modificado, aumentando-se os valores máximos  $NP(j)$  e/ou  $NS(j)$ .

#### **APROX0\_fmsz: aproximação inicial**

O procedimento para estimativa inicial é similar ao desenvolvido para o problema de processo único, pois também se supõe que a soma dos tempos de campanha  $t_i$  deverá satisfazer o horizonte disponível  $H$  em todos os estágios. Por conseguinte, deseja-se prever o número de processos e os volumes mínimos necessários,  $p\_h0$  e  $V\_h0$ , em cada estágio  $j$ , de acordo com:

$$4A.21 \quad H = \sum_{i=1}^N \frac{S_{ij} Q_i \tau_{ij}}{p\_h0 * V\_h0} \Rightarrow p\_h0 * V\_h0 = \sum_{i=1}^N \frac{S_{ij} Q_i \tau_{ij}}{H}, \quad \forall j$$

Os valores previstos desta forma contínua são aproximados aos valores discretos superiores, número de processos  $p'(j)$  e dimensão  $s'(j)$ , que foram especificados nos dados do problema, considerando-se valor não-nulo apenas para as respectivas variáveis binárias,  $iy\_h0(j, s'(j), p'(j)) = 1$ . De seguida, verifica-se a admissibilidade desta solução utilizando-se o procedimento *VERIFICA\_SOLUCAO\_fmsz*.

#### **APROX1\_fmsz: construção de solução admissível**

Desenvolveram-se aproximações alternativas no intuito de construir diferentes soluções admissíveis:

- APROX1A\_fmsz – abordagem através do enquadramento de processo único por estágio, caso tal seja admissível;
- APROX1B\_fmsz – alteração de solução inadmissível, no enquadramento de procura local iterada (*iterated local search*); pretende-se aumentar a dimensão e/ou número dos processos, repercutindo a redução dos tempos de campanha com os custos associados.

#### *Procedimento APROX1A\_fmsz*

A implementação do procedimento APROX1A\_fmsz baseia-se no pressuposto de admissibilidade da solução em estágios simples, isto é, estágios com um só processo. Uma solução admissível construída desta maneira constituirá uma *upper bound* para o problema de processos múltiplos, a partir da qual se poderão aplicar procedimentos de melhoria (em APROX2\_fmsz, adiante). O procedimento é constituído pelas seguintes etapas: verificação de admissibilidade para o enquadramento de processo único; e, se admissível, aplicação dos procedimentos já implementados, nomeadamente, aproximação inicial (APROX0\_fssz) e construção de solução admissível (APROX1\_fssz).

#### *Procedimento APROX1B\_fmsz*

O procedimento APROX1B\_fmsz prevê a construção de uma solução admissível a partir de uma inadmissível (oriunda de APROX0\_fmsz ou de APROX1A\_fmsz), tendo como objectivo a redução dos tempos de produção  $t_i$  no intuito de se observar o horizonte  $H$ , dirigindo essa procura de acordo com os respectivos custos dos equipamentos,  $c_{j\text{sp}}$ . A redução de tempos preconizada provém de uma alteração (aumento) da dimensão ou número do equipamento num dado estágio, sendo essa alteração seleccionada conforme a melhor razão obtida entre as variações, nos tempos e nos custos, associadas à referida alteração. Para a adequada selecção da alteração, avaliam-se os seus efeitos: por aumento da dimensão,  $dv(j,s'(j))$ , nos números de *batches*,  $n_i$ ; por aumento do número de processos,  $p(j)$ , nos tempos de ciclo,  $CT_i$ .

A abordagem efectuada é similar àquela realizada para *single machine*, a qual tinha produzido resultados satisfatórios, sendo aqui também possível estimar directamente a variação nos custos originada pela alteração seleccionada, seja em dimensão, seja em número. Quanto à variação dos tempos, atende-se a que o valor do tempo de campanha de cada produto é definido pelo respectivo “estágio crítico”,  $j\_nb1(i)$  ou  $j\_ct1(i)$ , respectivamente, segundo a dimensão ou o número. Alterando a dimensão ou número dos processos num “estágio crítico”, pode obter-se um “segundo estágio crítico” para cada produto,  $j\_nb2(i)$  ou  $j\_ct2(i)$ , definido pelo valor seguinte dos tempos por ordem decrescente. Estima-se a variação total dos tempos, originada pela alteração da dimensão ou número num dado “estágio crítico”, e compara-se com a correspondente variação dos custos, seleccionando-se a melhor (de valor máximo da razão das variações) alteração analisada.

De facto, atendendo às igualdades em 4A.16,

$$4A.22 \quad t_i = \max_j \left\{ \frac{\tau_{ij}}{p'(j)} n_i \right\} = CT_i \cdot n_i, \quad \text{com} \quad \begin{cases} n_i = \max_j \left\{ \frac{S_{ij} Q_i}{dv_{j,s'(j)}} \right\} \\ CT_i = \max_j \left\{ \frac{\tau_{ij}}{p'(j)} \right\} \end{cases}, \quad \forall i$$

Então, em cada iteração da busca local, alterando a dimensão (ou número) num dado “estágio crítico”,  $j\_nb1(i)$  (ou  $j\_ct1(i)$ ), vão modificar-se os valores representados nas variáveis auxiliares associadas,  $nbcrit1(i)$  (ou  $ctcrit1(i)$ ), que serão guardados em  $nbcrit2(i)$  (ou  $ctcrit2(i)$ ), dado que estas são definidas por um “segundo estágio crítico”,  $j\_nb2(i)$  (ou  $j\_ct2(i)$ ). Relacionando a variação nos tempos através destas variáveis auxiliares, apresenta-se para a variação dos números de *batches*,  $n(i)$ , devido a aumento na dimensão,

$$4A.23 \quad \begin{aligned} \Delta t_i &= tcrit1_i - tcrit2_i \\ &= CT_i \cdot nbcrit1_i - CT_i \cdot nbcrit2_i = CT_i \cdot nbcrit1_i - CT_i \cdot nbcrit1_i \left( \frac{nbcrit2_i}{nbcrit1_i} \right) \\ &= CT_i \cdot nbcrit1_i \left( 1 - \frac{nbcrit2_i}{nbcrit1_i} \right) = tcrit1_i \left( 1 - \frac{nbcrit2_i}{nbcrit1_i} \right), \quad \forall i \end{aligned}$$

e de forma similar, para a variação dos tempos de ciclo,  $CT(i)$ , por aumento do número de processos,

$$4A.24 \quad \begin{aligned} \Delta t_i &= tcrit1_i - tcrit2_i \\ &= n_i \cdot CTcrit1_i - n_i \cdot CTcrit2_i = n_i \cdot CTcrit1_i - n_i \cdot CTcrit1_i \left( \frac{CTcrit2_i}{CTcrit1_i} \right) \\ &= n_i \cdot CTcrit1_i \left( 1 - \frac{CTcrit2_i}{CTcrit1_i} \right) = tcrit1_i \left( 1 - \frac{CTcrit2_i}{CTcrit1_i} \right), \quad \forall i \end{aligned}$$

Sintetizando: a partir de uma solução inadmissível, propõe-se uma alteração por aumento de dimensão (ou número) num dado estágio; determina-se a totalidade das variações provocadas nos tempos, compara-se com a variação nos custos e selecciona-se a melhor razão entre essas variações; então, alteram-se as respeitantes variáveis binárias. Esta solução é verificada no procedimento VERIFICA\_SOLUCAO\_fmshz, pretendendo-se analisar a sua admissibilidade.

Este procedimento APROX1B\_fmshz é iterativo, terminando a sua execução quando se obtiver uma solução admissível. Consequentemente, este procedimento constitui um método de procura local iterada (*iterated local search*), onde:

- a vizinhança de procura é definida através do aumento de dimensão ou número em cada “estágio crítico”;
- a função de avaliação corresponde à razão entre as variações estimadas nos tempos e nos custos;
- a alteração processa-se segundo a solução vizinha com melhor (máximo) valor da referida razão;
- o procedimento termina quando se encontra uma solução admissível, ou se atinge o número limite de iterações.

A hipótese de alteração simultânea de dimensão e número dos processos em cada estágio, de acordo com o custo associado, foi implementada e testada. Obtiveram-se, de facto, soluções admissíveis em todas as instâncias em que tal era expectável. Como os desvios eram significativos, optou-se então pela alteração parcelar, ou da dimensão ou do número dos processos, em cada iteração do processo de busca local.

Saliente-se que também poderia ser equacionada a abordagem de repartir o tempo disponível  $H$  pelos diferentes produtos  $i$ , de forma proporcional às respectivas quantidades procuradas,  $Q_i$ , tal como foi realizado no enquadramento de processo único (em APROX1A\_fsz). Todavia, dada a diminuta qualidade das aproximações aí observadas, não se perspectiva que se venham a obter melhores resultados no enquadramento mais complexo de processos múltiplos. Assim, não se implementou esta abordagem, mas pode comprovar-se a admissibilidade das soluções dela decorrentes (sempre que a instância em causa também seja admissível), com fundamentação analítica similar à efectuada em termos de processo único.

### APROX2\_fsz: melhoria de solução admissível

O procedimento APROX2\_fsz pretende melhorar uma solução admissível (proveniente de APROX1A\_fsz ou de APROX1B\_fsz), visando a redução dos custos dos equipamentos,  $c_{jsp}$ , para os estágios “não-críticos” (estágios cujo tempo de processamento, para cada produto, não corresponde ao valor máximo dos tempos do produto considerado, sendo portanto inferior ao tempo de campanha). Essa redução dos custos deriva de uma alteração (diminuição) da dimensão ou número do equipamento num dado estágio, donde os tempos de campanha poderão (ou não)  $t_i$  aumentar, mas de forma a que se continue a observar o horizonte  $H$ . Tal alteração é seleccionada segundo a melhor razão obtida entre as variações nos custos e nos tempos associados. Para a eficiente selecção da alteração a efectuar, de forma similar mas com efeito oposto ao de APROX1B\_fsz (aumento de dimensão ou número dos processos), avaliam-se os efeitos da alteração: por diminuição da dimensão,  $dv(j,s'(j))$ , no aumento dos números de *batches*,  $n_i$ ; por diminuição do número de processos,  $p(j)$ , nos tempos de ciclo,  $CT_i$ .

O tratamento efectuado assemelha-se ao de APROX1B\_fsz, mas com sentido oposto (diminuições em vez de aumentos), pelo que se relaciona a variação nos tempos através das variáveis auxiliares dos números de *batches*,  $n(i)$ , por diminuição na dimensão,

$$\begin{aligned}
 \Delta t_i &= tcrit2_i - tcrit1_i \\
 4A.25 \quad &= CT_i \cdot nbcrit2_i - CT_i \cdot nbcrit1_i = CT_i \cdot nbcrit1_i \left( \frac{nbcrit2_i}{nbcrit1_i} \right) - CT_i \cdot nbcrit1_i \\
 &= CT_i \cdot nbcrit1_i \left( \frac{nbcrit2_i}{nbcrit1_i} - 1 \right) = tcrit1_i \left( \frac{nbcrit2_i}{nbcrit1_i} - 1 \right), \quad \forall i
 \end{aligned}$$

Similarmente, por aumento do número de processos, estima-se a variação dos tempos de campanha em termos da variação dos tempos de ciclo,  $CT(i)$ .

Então, iniciando o procedimento com uma solução admissível, analisa-se uma alteração diminuindo da dimensão (ou número) num dado estágio; a variação nos custos é comparada com a totalidade das variações nos tempos, seleccionando-se a

melhor razão; alteram-se as variáveis binárias associadas a tal razão. A admissibilidade da solução assim obtida é testada em VERIFICA\_SOLUCAO\_fmsz.

O procedimento APROX2\_fmsz também é iterativo (*iterated local search*), apresentando como critério de paragem o esgotamento das alternativas (diminuições) admissíveis, bem como um limite máximo para as iterações.

### ALGORITMO heur\_fmsz

#### PREVIO\_fmsz: admissibilidade

!\*especificar número e volume máximos para cada estágio  $j$ :

iy\_h (:,:,)= 0; iy\_h (:, NS, NJ) = 1

!\*verificar admissibilidade desta solução:

CALL VERIFICA\_SOLUCAO\_fmsz

IF (IFEASfmsz == 1) THEN

WRITE (“ERRO: fmsz inadmissível”); STOP

ELSE

GOTO APROX0\_fmsz

END IF

#### END PREVIO\_fmsz

\*\*\*\*\*

#### APROX0\_fmsz: aproximação inicial

iy\_h0 = 0; IFLAG0 = 1

cic\_estag: DO j=1,nj !\* volume total

soma = 0.

DO i=1,ni; soma = soma + SF(i,j) \* q(i) \* tau(i,j); END DO

vtotal = soma / H

ip\_h0(j) = CEILING(vtotal / dv(j,ns)) !\* numero minimo de unidades

!\* especificacao de variaveis binarias

v\_h = vtotal / ip\_h0(j)

isiz = MINLOC (dv(j,:), MASK= dv(j,:) >= v\_h)

is\_h0(j) = isiz(1)

iy\_h0(j, is\_h0(j), ip\_h0(j)) = 1

END DO cic\_estag

!\* verificar solucao

CALL VERIFICA\_SOLUCAO\_fmsz

IF (IERRO == 0) THEN IFLAG0 = 0; END IF

#### END APROX0\_fmsz

\*\*\*\*\*

#### APROX1\_fmsz: construção de solução admissível

!\* inicializacao

iy\_h1 = 0; cost\_h1 = 1.D15; IFLAG1 = 1

!\* APROX1A\_fmsz: aproximacao com um unico processo por estagio

IFLAG1A = 1

icount = 0

```

DO j=1,nj
  iy_teste = iy_h0(j,,:)
  np0_loc = MAXLOC (iy_teste, DIM=2)
  IF (np0_loc(1) > 1.1) THEN
    icount = icount + 1; EXIT
  END IF
END DO

IF(icount == 0) THEN CALL APROX1A_fmsz; END IF

IF (IFLAG1A == 0) THEN
  iy_h1 = iy_h1A; t_h1 = t_h1A; cost_h1 = cost_h1A
  fnb_h1 = fnb_h1A; IFLAG1 = IFLAG1A
END IF

!* APROX1B_fmsz: reducao de tempos (iterada) dirigida pelos custos
!* ILS (Iterated Local Search)
ILS: IF (IFLAG1 == 1) THEN
  IFLAG1B = 1; icount = 1

  IF (IFLAGfssz == -1) THEN
    iy_h1B0 = iy_h0; t_h1B0 = t_h0; fnb_h1B0 = 0.
  ELSE
    iy_h1B0 = iy_h1A; t_h1B0 = t_h1A; fnb_h1B0 = fnb_h1A
  END IF

  njsp = (ns * np)**nj
  DO WHILE (IFLAG1B == 1 .AND. icount <= njsp)
    CALL APROX1B_fmsz
    icount = icount + 1
    iy_h1B0 = iy_h1B; t_h1B0 = t_h1B; fnb_h1B0 = fnb_h1B
  END DO

  IF (IFLAG1B == 0 .AND. cost_h1B < cost_h1) THEN
    iy_h1 = iy_h1B; t_h1 = t_h1B; cost_h1 = cost_h1B
    fnb_h1 = fnb_h1B; IFLAG1 = IFLAG1B
  END IF
END IF ILS

```

**END APROX1\_fmsz**

\*\*\*\*\*

**APROX1A\_fmsz: construção supondo processo único**

```

!* iniciar
  iy_h1A = 0; iy_hs0 = iy_h0(:,,1)
  c_s = c_m(:,,1); ct = MAXVAL (tau, DIM=2); IFLAG1A = 1

!* procedimentos para processo unico
  CALL PREVIO_fssz
  IF (IFEASfssz == 1) THEN RETURN; END IF

!* APROX0_fssz: aproximacao inicial
  CALL APROX0_fssz
  IF (IFLAGfssz0 == 0) THEN iy_h1A(:,,1) = iy_hs0; END IF

```

```

!* APROX1_fssz: construção de solucao admissivel (fssz)
  IF (IFLAGfssz0 == 1) THEN
    CALL APROX1_FSSZ
    IFLAGfssz = IFLAGfssz1
    IF (IFLAGfssz1 == 0) THEN iy_h1A(:,1) = iy_hs1; END IF
  END IF

!* verificar solucao
  CALL VERIFICA_SOLUCAO_fmsz
  IF (IERRO == 0) THEN IFLAG1A = 0; END IF

```

**END APROX1A\_fmsz**

\*\*\*\*\*

### **APROX1B\_fmsz: construção de solução admissível através de procura local iterada**

```

!* 1) inicializacao
  IFLAG1B = 1; iy_h1B = iy_h1B0
  DO j=1,nj ! agregacao ns*np
    DO is=1,ns; DO ip=1,np
      isp = is + ns * (ip - 1); iy_aux(j,isp) = iy_h1B0(j,is,ip)
    END DO; END DO
  END DO
  isp_c1 = MAXLOC (iy_aux, DIM=2, MASK=iy_aux > .1)

cic_estag1: DO j=1,nj !* size e numero (para o c1)
  DO is=1,ns; DO ip=1,np
    isp = is + ns * (ip - 1)
    IF (isp == isp_c1(j)) THEN
      is_c1(j) = is; ip_c1(j) = ip; cycle cic_estag1
    END IF
  END DO; END DO
END DO cic_estag1

!* 2) Alteracao nos volumes: varia nb(i)
!* 2A) estagio critico, j_nb1(i): define o numero de batches do produto (i)
  DO j=1,nj
    Vaux(:,j) = dv(j,is_c1(j))
    DO i=1,ni; fnb_aux(i,j) = q(i) * SF(i,j) / Vaux(i,j); END DO
  END DO
  j_nb1 = MAXLOC (fnb_aux, DIM=2)
  DO i=1,ni; fnbcrit1(i) = q(i) * SF(i,j_nb1(i)) / Vaux(i,j_nb1(i)); END DO

!* fnbcrit2: nb do "estágio critico seguinte"
!* fnbprov: nb da dimensao "seguinte" no mesmo "estagio critico"
  DO i=1,ni
    j_nb2(i)=MAXLOC(fnb_aux(i,:), DIM=1, MASK=fnb_aux(i,:) > fnbcrit1(i))

    fnbcrit2(i)=MAXVAL(fnb_aux(i,:),DIM=1,MASK=fnb_aux(i,:) < fnbcrit1(i))
  END DO
  DO i=1,ni

```

```

        IF (is_c1(j_nb1(i)) == ns) THEN
            fncrit2(i) = fncrit1(i)
        ELSE
            fnbprov(i) = q(i) * SF(i,j_nb1(i)) / &
                dv(j_nb1(i), is_c1(j_nb1(i)) + 1)
            IF(fncrit2(i) < fnbprov(i)) THEN
                fncrit2(i) = fnbprov(i)
            END IF
        END IF
    END DO
!* 2B) razao_tc: razao entre a variacao nos tempos e nos custos
    DO i=1,ni
        IF(is_c1(j_nb1(i)) == ns) THEN
            delta_cs(i) = 1.D10
        ELSE
            delta_cs(i) = c_m(j_nb1(i), is_c1(j_nb1(i)) + 1, ip_c1(j_nb1(i))) -&
                c_m(j_nb1(i), is_c1(j_nb1(i)), ip_c1(j_nb1(i)))
        END IF
    END DO
    delta_ts = t_h1B0 * (1. - fncrit2 / fncrit1)
    soma_ts = 0.
    DO k=1,ni; DO l=1,ni
        IF(j_nb1(k) == j_nb1(l)) THEN
            soma_ts(k) = soma_ts(k) + delta_ts(l)
        END IF
    END DO; END DO
    razao_tes = soma_ts / delta_cs
!* 3) Alteracao no numero de processos: varia CT(i)
!* 3A) estagio critico, j_ct1(i): define o CycleTime do produto (i)
    DO i=1,ni; DO j=1,nj; ct_aux(i,j) = tau(i,j) / ip_c1(j); END DO; END DO
    j_ct1 = MAXLOC (ct_aux, DIM=2)
    DO i=1,ni; ctcrit1(i) = tau(i,j_ct1(i)) / ip_c1(j_ct1(i)); END DO
!* ctcrit2: CT do "estágio critico seguinte"
!* ctprov: CT do "numero seguinte" no mesmo "estagio critico"
    DO i=1,ni
        j_ct2(i) = MAXLOC (ct_aux(i,:), DIM=1, MASK=ct_aux(i,:) < ctcrit1(i))
        ctcrit2(i) = MAXVAL (ct_aux(i,:), DIM=1, MASK=ct_aux(i,:) < ctcrit1(i))
    END DO
    DO i=1,ni
        IF (ip_c1(j_ct1(i)) == np) THEN
            ctcrit2(i) = ctcrit1(i)
        ELSE
            ctprov(i) = tau(i,j_ct1(i)) / (ip_c1(j_ct1(i)) + 1)
            IF(ctcrit2(i) < ctprov(i)) THEN
                ctcrit2(i) = ctprov(i)
            END IF
        END IF
    END DO

```

```

                END IF
            END IF
        END DO
!* 3B) razao_tcp: razao entre a variacao nos tempos e nos custos
        DO i=1,ni
            IF(ip_c1(j_ct1(i)) == np) THEN
                delta_cp(i) = 1.D10
            ELSE
                delta_cp(i) = c_m(j_ct1(i), is_c1(j_ct1(i)), ip_c1(j_ct1(i)) + 1) - &
                    c_m(j_ct1(i), is_c1(j_ct1(i)), ip_c1(j_ct1(i)))
            END IF
        END DO
        delta_tp = t_h1B0 * (1. - ctcrit2 / ctcrit1)
        soma_tp = 0.
        DO k=1,ni; DO l=1,ni
            IF(j_ct1(k) == j_ct1(l)) THEN
                soma_tp(k) = soma_tp(k) + delta_tp(l)
            END IF
        END DO; END DO
        razao_tcp = soma_tp / delta_cp
!* 4) alteracao da solucao (binaria): aumento de dimensão/numero
!* dimensao (s)
        ibest_tcs = MAXLOC (razao_tcs)
        jauxs = j_nb1(ibest_tcs(1))
        ubounds = razao_tcs(ibest_tcs(1))
        DO WHILE (is_c1(jauxs) == ns)
            ibest_tcs = MAXLOC (razao_tcs, MASK= razao_tcs < ubounds)
            IF(ibest_tcs(1) == 0) THEN
                ibest_tcs(1) = 1; jauxs = j_nb1(ibest_tcs(1)); EXIT
            END IF
            ubounds = razao_tcs(ibest_tcs(1))
            jauxs = j_nb1(ibest_tcs(1))
        END DO
!* numero processos (p)
        ibest_tcp = MAXLOC (razao_tcp)
        jauxp = j_ct1(ibest_tcp(1))
        uboundp = razao_tcp(ibest_tcp(1))
        DO WHILE(ip_c1(jauxp) == np)
            ibest_tcp = MAXLOC (razao_tcp, MASK= razao_tcp < uboundp)
            IF(ibest_tcp(1) == 0) THEN
                ibest_tcp(1) = 1; jauxp = j_ct1(ibest_tcp(1)); EXIT
            END IF
            uboundp = razao_tcp(ibest_tcp(1))
            jauxp = j_ct1(ibest_tcp(1))
        END DO
!* alteracao binaria

```

```

IF(razao_tcs(ibest_tcs(1)) < razao_tcp(ibest_tcp(1)) ) THEN
  iy_h1B(jauxs, is_c1(jauxs), ip_c1(jauxs)) = 0
  iy_h1B(jauxs, is_c1(jauxs) + 1, ip_c1(jauxs)) = 1
ELSE
  iy_h1B(jauxp, is_c1(jauxp), ip_c1(jauxp)) = 0
  iy_h1B(jauxp, is_c1(jauxp) + 1, ip_c1(jauxp) + 1) = 1
END IF

```

!\* 5) verificar solucao

```

CALL VERIFICA_SOLUCAO_fmsz
IF (IERRO == 0) THEN IFLAG1B = 0; END IF

```

**END APROX1B\_fmsz**

\*\*\*\*\*

### **APROX2\_fmsz: melhoria de solução admissível**

!\* 1) inicializacao

```

IFLAG2A = 0; iy_h2A = iy_h2A0; t_h2A = t_h2A
fnb_h2A = fnb_h2A0; cost_h2A = SUM(c_m * iy_h2A0)

DO j=1,nj ! agregacao ns*np
  DO is=1,ns; DO ip=1,np
    isp = is + ns * (ip - 1); iy_aux(j,isp) = iy_h2A0(j,is,ip)
  END DO; END DO
END DO
isp_c1 = MAXLOC (iy_aux, DIM=2, MASK=iy_aux > .1)

```

cic\_estag1: DO j=1,nj ! size e numero (para o c1)

```

  DO is=1,ns; DO ip=1,np
    isp = is + ns * (ip - 1)
    IF (isp == isp_c1(j)) THEN
      is_c1(j) = is; ip_c1(j) = ip; cycle cic_estag1
    END IF
  END DO; END DO

```

END DO cic\_estag1

!2) ESTAGIOS: "criticos" vs "nao-criticos"

!\* 2A) estagio critico, j\_nb1(i): define o numero de batches do produto (i)

```

  DO j=1,nj
    Vaux(:,j) = dv(j,is_c1(j))
    DO i=1,ni; fnb_aux(i,j) = q(i) * SF(i,j) / Vaux(i,j); END DO
  END DO
  j_nb1 = MAXLOC(fnb_aux, DIM=2)
  DO i=1,ni; fnbcrit1(i) = q(i) * SF(i,j_nb1(i)) / Vaux(i,j_nb1(i)); END DO

```

icounts = 0

cic\_estag2: DO j=1,nj

ifails = 0

cic\_prod1: DO i=1,ni

IF(j\_nb1(i) == j) THEN ifails = ifails + 1; END IF

END DO cic\_prod1

IF (ifails == 0) THEN

```

        icounts = icounts + 1; jnb_prov(icounts) = j
    END IF
END DO cic_estag2

!* 2B) estagio critico, j_ct1(i): define o CycleTime do produto (i)
    DO i=1,ni; DO j=1,nj; ct_aux(i,j) = tau(i,j) / ip_c1(j); END DO; END DO
    j_ct1 = MAXLOC(ct_aux, DIM=2)
    DO i=1,ni; ctcrit1(i) = tau(i,j_ct1(i)) / ip_c1(j_ct1(i)); END DO
    icountp = 0
cic_estag3: DO j=1,nj
    ifailp = 0
    cic_prod2: DO i=1,ni
        IF(j_ct1(i) == j) THEN ifailp = ifailp + 1; END IF
    END DO cic_prod2
    IF (ifailp .EQ. 0) THEN
        icountp = icountp + 1; jct_prov(icountp) = j
    END IF
END DO cic_estag3

!* teste: existem estagios NAO criticos?
    IFLAGnaocrit = icounts + icountp
    IF (IFLAGnaocrit == 0) THEN IFAIL = 1; RETURN; END IF

!* ALTERACAO: diminuir volume vs diminuir numero
    ubound = 1.D20; is_c2 = is_c1; ip_c2 = ip_c1; iterac = 0
cic_ils: DO WHILE (iterac < IFLAGnaocrit)
    iterac = iterac + 1

!* 3A) Alteracao nos volumes: varia nb(i)
    razao_cts = 0.; razmax_cts(1) = 0.; jbest_cts(1) = 1
ILS1: IF (icounts > 0) THEN
cic_icounts: DO k=1, icounts
    jauxs = jnb_prov(k); is_c2 = is_c1
    IF (is_c1(jauxs) > 1) THEN is_c2(jauxs) = is_c1(jauxs) - 1; END IF

!* fnbcrit2: nb da dimensao "anterior" no "estagio NAO critico"
    DO j=1,nj
        Vaux(:,j) = dv(j,is_c1(j))
        DO i=1,ni; fnb_aux(i,j) = q(i) * SF(i,j) / Vaux(i,j); END DO
    END DO
    Vaux(:,jauxs) = dv(jauxs,is_c2(jauxs))
    DO i=1,ni; fnb_aux(i,jauxs) = q(i) * SF(i,jauxs) / Vaux(i,jauxs); END DO
    j_nb2 = MAXLOC (fnb_aux, DIM=2)
    DO i=1,ni; fnbcrit2(i) = q(i) * SF(i,j_nb2(i)) / Vaux(i,j_nb2(i)); END DO

!* razao_ct: razao entre a variacao nos custos e nos tempos
    delta_cs = 0.
    IF(is_c1(jauxs) > 1) THEN
        delta_cs(jauxs) = c_m(jauxs, is_c1(jauxs), ip_c1(jauxs)) - &
            c_m(jauxs, is_c1(jauxs) - 1, ip_c1(jauxs))
    END IF

```

```

END IF
delta_ts = 0.
DO i=1,ni
  IF(j_nb2(i) == jauxs) THEN
    delta_ts(i) = t_h2A0(i) * (fnbcrit2(i) / fnbcrit1(i) - 1.)
  END IF
END DO

soma_ts = 0.
DO m=1,ni; DO l=1,ni
  IF(j_nb2(m) == j_nb2(l)) THEN
    soma_ts(m) = soma_ts(m) + delta_ts(l)
  END IF
END DO; END DO

DO i=1,ni
  soma_ts(i) = soma_ts(i) + 1.D-9
  IF(j_nb2(i) == jauxs) THEN
    razao_cts(jauxs) = delta_cs(jauxs) / soma_ts(i)
  END IF
END DO

IF (razao_cts(jauxs) == 0. .AND. delta_cs(jauxs) > 0.) THEN
  razao_cts(jauxs) = delta_cs(jauxs) * 1.D9
END IF

END DO cic_icounts

IF(ubound == 0.) THEN
  razmax_cts = 0.
  jbest_cts = 1
ELSE
  razmax_cts = MAXVAL(razao_cts, MASK= razao_cts < ubound)
  jbest_cts = MAXLOC (razao_cts, MASK= razao_cts < ubound)
END IF

END IF ILS1

!* 3B) Alteracao no numero de processos: varia CT(i)
  razao_ctp = 0.; razmax_ctp(1) = 0.; jbest_ctp(1) = 1

ILS2: IF (icountp > 0) THEN
cic_icountp: DO k=1, icountp
  jauxp = jct_prov(k)
  ip_c2 = ip_c1
  IF (ip_c1(jauxp) > 1) THEN
    ip_c2(jauxp) = ip_c1(jauxp) - 1
  END IF
END DO

!* ctcrit2: CT do numero "anterior" no "estagio NAO critico"
DO i=1,ni; DO j=1,nj; ct_aux(i,j) = tau(i,j) / ip_c1(j); END DO; END DO
DO i=1,ni; ct_aux(i,iauxp) = tau(i,iauxp) / ip_c2(jauxp); END DO
j_ct2 = MAXLOC(ct_aux, DIM=2)

```

```

    ctcrit2 = ctcrit1
    DO i=1,ni; ctcrit2(i) = tau(i,j_ct2(i)) / ip_c2(j_ct2(i)); END DO
!* razao_ctp: razao entre a variacao nos custos e nos tempos
    delta_cp = 0.
    IF(ip_c1(jauxp) > 1) THEN
        delta_cp(jauxp) = c_m(jauxp, is_c1(jauxp), ip_c1(jauxp)) - &
            c_m(jauxp, is_c1(jauxp), ip_c1(jauxp) - 1)
    END IF
    delta_tp = 0.
    DO i=1,ni
        IF(j_ct2(i) ==iauxp) THEN
            delta_tp(i) = t_h2A0(i) * (ctcrit2(i) / ctcrit1(i) - 1.)
        END IF
    END DO
    soma_tp = 0.
    DO m=1,ni; DO l=1,ni
        IF(j_ct2(m) == j_ct2(l)) THEN
            soma_tp(m) = soma_tp(m) + delta_tp(l)
        END IF
    END DO; END DO
    DO i=1,ni
        soma_tp(i) = soma_tp(i) + 1.D-9
        IF(j_ct2(i) ==iauxp) THEN
            razao_ctp(jauxp) = delta_cp(jauxp) / soma_tp(i)
        END IF
    END DO
    IF (razao_ctp(jauxp) == 0. .AND. delta_cp(jauxp) > 0.) THEN
        razao_ctp(jauxp) = delta_cp(jauxp) * 1.D9
    END IF
END DO cic_icountp
    IF(ubound == 0.) THEN
        razmax_ctp = 0.
        jbest_ctp = 1
    ELSE
        razmax_ctp = MAXVAL(razao_ctp, MASK= razao_ctp < ubound)
        jbest_ctp = MAXLOC(razao_ctp, MASK= razao_ctp < ubound)
    END IF
END IF ILS2
!* 4) alteracao da solucao (binaria): diminuir dimensao/numero
!* dimensao (s) vs numero processos (p)
   iauxs = jbest_cts(1);iauxp = jbest_ctp(1);ialter = 0
!* alteracao binaria
    IF(razmax_cts(1) >= razmax_ctp(1) ) THEN
        ubound = razmax_cts(1)
        iy_h2A(iauxs, is_c1(iauxs), ip_c1(iauxs)) = 0

```

```

        iy_h2A(jauxs, is_c2(jauxs), ip_c1(jauxs)) = 1
        ialter = is_c1(jauxs) - is_c2(jauxs)
    ELSE
        ubound = razmax_ctp(1)
        iy_h2A(jauxp, is_c1(jauxp), ip_c1(jauxp)) = 0
        iy_h2A(jauxp, is_c1(jauxp), ip_c2(jauxp)) = 1
        ialter = ip_c1(jauxp) - ip_c2(jauxp)
    END IF
!* 5) verificar solucao
    CALL VERIFICA_SOLUCAO_fmsz
    condic: IF (IERRO == 0 .AND. ialter > 0) THEN
        IFLAG2A = 0; EXIT cic_ils
    ELSE
        iy_h2A = iy_h2A0; t_h2A = t_h2A0
        fnb_h2A = fnb_h2A0; cost_h2A = SUM(c_m * iy_h2A0)
        IF (iterac == IFLAGGnaocrit) THEN IFAIL = 1; END IF
    END IF condic
END DO cic_ils
END APROX2_fmsz
*****

VERIFICA_SOLUCAO_fmsz
!*definir o custo total da presente solução:
    Cost_h = SUM((j,s,n), C * iy_h)

!*estimar numero de ciclos para o volume especificado,  $dv(j,s'(j))$ , em cada estágio
j:
    nb_h(i,j) = SF(i,j)*q(i)/dv(j,s'(j))

!*determinar numero de ciclos para cada produto i,
    nbmax_h(i) = MAXVAL (nb_h(i,j), DIM=2)

!*estimar tempos de produção em cada estágio j,
    tempos_h(i,j) = tau(i,j) / p'(j) * nbmax_h(i)

!*tempos de campanha para cada produto i:
    t_h(i) = MAXVAL (tempos_h(i,j), DIM=2)

!*calcular a soma dos tempos de campanha:
    somatempos = SUM(i, t_h(i))

!*infeasibility caso não se observe o horizonte H:
    IF (somatempos > H) THEN ERRO; END IF
END VERIFICA_SOLUCAO_fmsz
*****

```

### Modelo batch\_milp\_fmsz

Supondo *single machine*, pretende comparar-se a modelação descrita no âmbito de MPC com a correspondente modelação de SPC, em *batch\_milp\_fssz*, definindo-se uma instância em que o número de transições entre *batches* do mesmo

produto seja precisamente o número de *batches* desse produto menos uma unidade, configurando assim campanhas de produto único.

Então, considere-se a instância do problema em *batch\_milp\_fsmz*, onde:

- o número de transições entre *batches* do mesmo produtos observa a conhecida igualdade, impondo SPC,

$$Nch_{ii} = n_i - 1, \quad \forall i$$

- tal como em *batch\_milp\_fssz*, negligenciam-se “cabeça” (*head*) e “cauda” (*tail*) do sequenciamento, supondo-se nulas as previsíveis folgas entre as campanhas de produção dos diferentes produtos, bem como se negligenciam os tempos de limpeza,  $CL_{ikj} = 0$ ;
- todos os restantes parâmetros e dados coincidem com os correspondentes do modelo *batch\_milp\_fssz*.

Constata-se-se a coincidência da função objectivo com a de *batch\_milp\_fssz*. Também coincidem as restrições para selecção de um só processo por estágio.

Atendendo à imposição sobre o número de transições dos *batches*, acima descrita, observa-se para cada produto

$$4A.26 \quad \sum_{k=1}^N Nch_{ik} = n_i, \quad \forall i \quad \Leftrightarrow \quad (n_i - 1) + \sum_{\substack{k=1 \\ k \neq i}}^N Nch_{ik} = n_i$$

$$\Rightarrow \sum_{\substack{k=1 \\ k \neq i}}^N Nch_{ik} = 1, \quad \forall i \quad \Rightarrow \quad \begin{cases} Nch_{ik'} = 1, & \exists^1 k' \neq i \\ Nch_{ik} = 0, & \forall k \neq k', \quad \forall k \neq i \end{cases}$$

De forma similar, também

$$4A.27 \quad \sum_{i=1}^N Nch_{ik} = n_k, \quad \forall k \quad \Leftrightarrow \quad (n_k - 1) + \sum_{\substack{i=1 \\ i \neq k}}^N Nch_{ik} = n_k$$

$$\Rightarrow \sum_{\substack{i=1 \\ i \neq k}}^N Nch_{ik} = 1, \quad \forall k \quad \Rightarrow \quad \begin{cases} Nch_{i'k} = 1, & \exists^1 i' \neq k \\ Nch_{ik} = 0, & \forall i \neq i', \quad \forall i \neq k \end{cases}$$

Das relações 4A.26 e 4A.27, decorre que a restrição sobre o horizonte temporal se apresentará da forma

$$4A.28 \quad \sum_{i=1}^N \left[ n_i \tau_{ij} + \sum_{k=1}^N SL_{ikj} Nch_{ik} \right] \leq H, \quad \forall j \Rightarrow$$

$$\sum_{i=1}^N \left[ n_i \tau_{ij} + (n_i - 1) SL_{ijj} + 1 \cdot SL_{ik'j} \right] \leq H, \quad \forall j, \quad \exists^1 k' \neq i$$

Ocorre a seguinte relação entre as folgas e os tempos de ciclo, para SPC:

$$4A.29 \quad SL_{ijj} = CT_i - \tau_{ij}, \quad \forall i, j$$

De facto, a relação entre as folgas e os tempos de ciclo em 4A.29 foi aqui derivada do algoritmo de Biegler *et al.* [1997], oportunamente descrito na Sec. 2.2,

para a determinação das folgas por transição de um *batch*  $i$ , para um *batch* do mesmo produto  $i$  ( $k = i$ ), considerando-se tempos de limpeza nulos,  $CL_{ij} = 0$ :

$$\begin{aligned}
 & \text{1) Definição dos tempos de início, } Tini_j : \\
 & \quad \begin{cases} Tini_1 = \tau_{i1} \\ Tini_j = Tini_{j-1} + \tau_{k,j-1}, j = 2..M \end{cases} \\
 & \quad \text{ou, } Tini_j = \tau_{i1} + \sum_{l=1}^{j-1} \tau_{il}, j = 2..M \\
 & \text{2) Cálculo das folgas e da folga mínima:} \\
 & \quad \begin{aligned} \mathit{folg}_j &= Tini_j - \sum_{l=1}^j \tau_{il}, j = 1..M \\ &= \tau_{i1} + \sum_{l=1}^{j-1} \tau_{il} - \sum_{l=1}^j \tau_{il} = \tau_{i1} - \tau_{ij}, j = 1..M \end{aligned} \\
 & \quad \delta = \min_j \{ \mathit{folg}_j \} \\
 & \quad = \min_j \{ \tau_{i1} - \tau_{ij} \} = \tau_{i1} - \max_j \{ \tau_{ij} \} \\
 & \text{3) Actualização das folgas:} \\
 & \quad \begin{aligned} SL_{ij} &= \mathit{folg}_j - \delta \\ &= (\tau_{i1} - \tau_{ij}) - (\tau_{i1} - \max_j \{ \tau_{ij} \}) \\ &= CT_i - \tau_{ij} \end{aligned}
 \end{aligned}$$

Assim, substituindo 4A.29 em 4A.28, e negligenciando-se “cabeça” (*head*) e “cauda” (*tail*) do sequenciamento, obtém-se

$$\begin{aligned}
 \text{4A.30} \quad & \sum_{i=1}^N [n_i \tau_{ij} + (n_i - 1)(CT_i - \tau_{ij})] \leq H, \quad \forall j \Rightarrow \\
 & \sum_{i=1}^N [n_i CT_i + (-1)SL_{ij}] \leq H \Rightarrow \sum_{i=1}^N n_i CT_i \leq H \Rightarrow \sum_{i=1}^N t_i \leq H
 \end{aligned}$$

Por conseguinte, comprova-se equivalência entre as funções objectivo e os espaços de procura, pelo que se pode afirmar que a instância específica do problema em *batch\_milp\_fsmz* é equivalente ao problema em *batch\_milp\_fssz*. Também se constata numericamente a correspondência das variáveis, no contexto das condicionantes da instância.

### Modelo *batch\_milp\_fmmz*

Quanto ao último modelo analisado neste contexto, considerando *multiple machine*, compara-se com o modelo imediatamente anterior e onde se supôs *single machine*, *batch\_milp\_fsmz*, definindo-se uma instância onde o número máximo de processos em cada estágio seja apenas de um, bem como se assumem condicionantes iguais para ambos os modelos.

Defina-se a instância do problema em *batch\_milp\_fmmz*, de acordo com:

- o número máximo de processos em cada estágio é unitário,  $NP(j) = 1$ ; por conseguinte, o índice  $p$  é redundante e pode negligenciar-se;
- os outros parâmetros e dados são coincidentes em ambos os modelos em análise,  $batch\_milp\_fmmz$  e  $batch\_milp\_fsmz$ , nomeadamente, supõem-se nulos os valores para as folgas,  $SL_{ikj} = 0$ .

Para a função objectivo, negligenciando-se o índice redundante  $p$ , verifica-se a coincidência com a de  $batch\_milp\_fsmz$ , de forma similar à descrita em 4A.18.

No mesmo sentido, as restrições referentes à selecção de um só processo por estágio também coincidem, tal como descrito em 4A.19, bem como se equivalem as restrições definidoras dos números de *batches*,

$$4A.31 \quad S_{ij} Q_i \sum_{s=1}^{NS(j)} \sum_{p=1}^{NP(j)} \frac{y_{jsp}}{dv_{js}} - n_i \leq 0 \Rightarrow S_{ij} Q_i \sum_{s=1}^{NS(j)} \frac{y_{js}}{dv_{js}} - n_i \leq 0, \quad \forall i, j$$

As restrições relativas aos números de transições entre as produções *batch* dos diferentes produtos,  $Nch_{ikj}$ , são coincidentes nos dois modelos.

Finalmente, coincidem em ambos os modelos as restrições quanto ao horizonte temporal, pois para  $batch\_milp\_fmmz$ ,

$$4A.32 \quad \sum_{i=1}^N n_i \tau_{ij} - H \cdot \sum_{s=1}^{NS(j)} \sum_{p=1}^{NP(j)} p(j) y_{jsp} \leq 0, \quad \forall j \Rightarrow$$

$$\sum_{i=1}^N n_i \tau_{ij} \leq H, \quad \forall j, \quad \exists^1 (s'(j), p'(j) = 1)$$

e para  $batch\_milp\_fsmz$ ,

$$4A.33 \quad \sum_{i=1}^N \left[ n_i \tau_{ij} + \sum_{k=1}^N Nch_{ikj} SL_{ikj} \right] \leq H, \quad \forall j \Rightarrow$$

$$\sum_{i=1}^N n_i \tau_{ij} \leq H, \quad \forall j, \quad e \text{ para } SL_{ikj} = 0.$$

Assim, constata-se a equivalência entre as funções objectivo e os espaços de procura, para a instância definida do problema em  $batch\_milp\_fmmz$  e relativamente às condicionantes referentes ao problema em  $batch\_milp\_fsmz$ . Embora o modelo  $batch\_milp\_fmmz$  possa apresentar incoerência nos seus pressupostos, pode comprovar-se a correspondência numérica das variáveis respectivas, no âmbito das condicionantes.

■

# Ilustração dos exemplos numéricos para análise probabilística de heurísticas

Apresentam-se de forma compreensiva, por descrição das funções geradoras de dados aleatórios em ambiente GAMS, os:

1. Dados do exemplo EX1 (série A);
2. Dados do exemplo EX2 (série B);
3. Estimadores para análise de desvios.

Complementa-se, apresentando os dados referentes à primeira instância aleatória para cada um dos exemplos referidos.

### 1. Dados do exemplo EX1 (série A)

```
*
* APfssz_data.inc
* modulo para definicao probabilistica dos parametros
  OPTION SEED = 26102006;
  H = UNIFORM (6000, 8000);
  Q(i) = UNIFORM (100000, 200000);
  alpha(j)= UNIFORM (200, 600);
  beta(j) = UNIFORM (0.5, 0.7);
  LOOP (j,
    dv(j,"1") = UNIFORM(800,1200);
    dv(j,"2") = UNIFORM(1500,2000);
    dv(j,"3") = UNIFORM(2001,2500);
    dv(j,"4") = UNIFORM(2800,3200);
  );
  SF(i,j) = UNIFORM (1,5);
  tau(i,j) = UNIFORM (2,9);
*****
*
100 NUMERO DE AMOSTRAS
  2 3 4 ni, nj, ns
  1 amostra
  6396.92 H
  130336.41 194230.75 Q(i)
  45236.23 56813.05 67928.10 81854.57 c(j,s)
```

19838.73	23776.95	30066.48	32794.57	c(j,s)
26642.83	35923.66	44871.32	53203.96	c(j,s)
1138.03	1637.38	2177.92	2933.26	dv(j,s)
1073.16	1535.65	2443.41	2901.62	dv(j,s)
974.22	1618.44	2361.19	3153.28	dv(j,s)
2.95	3.68	4.17		SF(i,j)
2.80	4.48	4.47		SF(i,j)
5.66	5.90	3.23		tau(i,j)
2.84	6.00	2.28		tau(i,j)

\*\*\*\*\*

## 2. Dados do exemplo EX2 (série B)

\*

\* APfssz\_data\_ex2b.inc

\* modulo para definicao probabilistica dos parametros

```

IF (ord(namost) EQ 1, OPTION SEED = 1000; );
IF (ord(namost) EQ 2, OPTION SEED = 2000; );
IF (ord(namost) EQ 3, OPTION SEED = 3000; );
IF (ord(namost) EQ 4, OPTION SEED = 4000; );
IF (ord(namost) EQ 5, OPTION SEED = 5000; );
IF (ord(namost) EQ 6, OPTION SEED = 6000; );
IF (ord(namost) EQ 7, OPTION SEED = 7000; );
IF (ord(namost) EQ 8, OPTION SEED = 8000; );
IF (ord(namost) EQ 9, OPTION SEED = 9000; );
IF (ord(namost) EQ 10, OPTION SEED = 10000; );
IF (ord(namost) EQ 11, OPTION SEED = 11000; );
IF (ord(namost) EQ 12, OPTION SEED = 12000; );
IF (ord(namost) EQ 13, OPTION SEED = 13000; );
IF (ord(namost) EQ 14, OPTION SEED = 14000; );
IF (ord(namost) EQ 15, OPTION SEED = 15000; );
IF (ord(namost) EQ 16, OPTION SEED = 16000; );
IF (ord(namost) EQ 17, OPTION SEED = 17000; );
IF (ord(namost) EQ 18, OPTION SEED = 18000; );
IF (ord(namost) EQ 19, OPTION SEED = 19000; );
IF (ord(namost) EQ 20, OPTION SEED = 20000; );
IF (ord(namost) EQ 21, OPTION SEED = 21000; );
IF (ord(namost) EQ 22, OPTION SEED = 22000; );
IF (ord(namost) EQ 23, OPTION SEED = 23000; );
IF (ord(namost) EQ 24, OPTION SEED = 24000; );
IF (ord(namost) EQ 25, OPTION SEED = 25000; );
IF (ord(namost) EQ 26, OPTION SEED = 26000; );
IF (ord(namost) EQ 27, OPTION SEED = 27000; );
IF (ord(namost) EQ 28, OPTION SEED = 28000; );
IF (ord(namost) EQ 29, OPTION SEED = 29000; );
IF (ord(namost) EQ 30, OPTION SEED = 30000; );

```

H = UNIFORM (6000, 8000);

Q(i) = UNIFORM (100000, 300000);

alpha(j)= UNIFORM (300, 700);

```

beta(j) = UNIFORM (0.5, 0.7);
LOOP (j,
  dv(j,"1") = UNIFORM(800,1200);
  dv(j,"2") = UNIFORM(1500,2000);
  dv(j,"3") = UNIFORM(2001,2500);
  dv(j,"4") = UNIFORM(2800,3200);
  dv(j,"5") = UNIFORM(3500,4000);
);
SF(i,j) = UNIFORM (1,5);
tau(i,j) = UNIFORM (2,9);

*****
*
30 NUMERO DE AMOSTRAS
3 4 5 ni, nj, ns
1 amostra
7749.83 H
133468.72 114492.40 199909.17 Q(i)
34763.47 53131.51 58054.89 72514.84 79213.60 c(j,s)
23165.60 33881.17 40840.56 48614.92 56349.66 c(j,s)
43986.95 77159.90 90245.12 101185.33 120843.69 c(j,s)
23233.12 34365.38 35832.34 43347.57 46809.91 c(j,s)
1007.55 1921.83 2199.39 3085.57 3529.80 dv(j,s)
851.82 1588.24 2157.09 2869.98 3655.56 dv(j,s)
867.09 1939.30 2427.11 2859.35 3687.30 dv(j,s)
916.10 1949.83 2113.62 3051.99 3539.80 dv(j,s)
2.05 3.95 4.51 3.43 SF(i,j)
3.87 2.78 1.08 1.84 SF(i,j)
1.23 4.72 3.40 4.08 SF(i,j)
5.88 4.11 8.08 8.25 tau(i,j)
7.45 4.75 6.77 4.61 tau(i,j)
5.65 7.80 3.32 6.69 tau(i,j)
*****

```

### 3. Estimadores para análise de desvios

No contexto dos estimadores registados na *Tabela 4.4*, considerando-se apenas as soluções subóptimas definidas pelos procedimentos heurísticos, os desvios percentuais (%*dsv*) na função objectivo (*z*) a minimizar, entre a solução óptima (índice “OPT”) e para a solução heurística (índice “H”), são definidos de acordo com:

$$4B.1 \quad \%dsv\_z = \frac{|z_{OPT} - z_H|}{z_{OPT}} \cdot 100$$

Também para as soluções subóptimas, o número de desvios em cada solução binária (*Ndsv<sub>y</sub>*) é dado por:

$$4B.2 \quad Ndsv\_y = \sum_{j=1}^M \sum_{s=1}^{NS} \frac{|y_{OPT}(j,s) - y_H(j,s)|}{2}$$

No âmbito da *Tabela 4.8*, o desvio percentual referente às melhorias na função objectivo da solução em MPC (índice “M”), relativamente à respeitante solução (admissível e comparável) considerando SPC (índice “S”), é dado por

$$4B.3 \quad \%dsv\_z = \frac{|z_M - z_S|}{z_M} \cdot 100$$

Ainda neste âmbito, o número de desvios melhorando a solução binária, para as instâncias comparáveis entre MPCe SPC, decorre de

$$4B.4 \quad Ndsv\_y = \sum_{j=1}^M \sum_{s=1}^{NS} \frac{|y_M(j,s) - y_S(j,s)|}{2}$$

■

### Aspectos do modelo estocástico de planeamento de processos

Assume-se que a taxa de produção máxima  $R_{ist}$  (do produto principal  $j'$ ) do esquema de produção  $s$  é proporcional à capacidade da unidade de produção para um esquema de produção de referência,  $\rho_{i\bar{s}} = 1$ ,

$$5A.1 \quad R_{ist} = \rho_{is} q_{it}, \quad \forall i, s, t$$

Note-se que, supondo que cada esquema de produção  $s$  apenas possua um componente principal  $j'$ , a selecção do esquema  $s$  referenciará univocamente o componente principal  $j'$ , mas este componente  $j'$  pode ser componente principal de mais do que um esquema de produção.  $\rho_{is}$  é a taxa de produção máxima (do produto principal  $j'$ ) do esquema de produção  $s$  para processos contínuos flexíveis, referentes a um esquema de produção de referência  $\bar{s}$  definido pela capacidade do processo  $i$ . Este valor de  $\rho_{is}$  permite definir linearmente a taxa de produção de qualquer esquema  $s$ , tendo por base a capacidade de produção horária  $q$  (kg/hora) do processo  $i$ .

Definindo  $T_{istr}$  como a quantidade de tempo durante o período  $t$  que se gasta a produzir (o produto principal  $j'$ ) pelo esquema de produção  $s$ , em cada cenário de carácter aleatório  $r$ , pode calcular-se a quantidade (de produto principal  $j'$ ) produzida pelo esquema  $s$  durante o período de tempo  $t$ :

$$5A.2 \quad W_{istr} = \rho_{is} q_{it} T_{istr}, \quad \forall i, s, t, r$$

Atente-se que sendo aleatórias as procuras de produtos, bem como serão aleatórios os valores das variáveis de compra  $Pur$  e de venda  $Sal$ , também serão aleatórios os valores de produção  $W$  e os tempos alocados  $T$ .

A afectação do tempo total de produção aos diferentes esquemas de produção  $s$ , durante o período de tempo  $t$ , não pode exceder o máximo de tempo disponível  $h_{it}$  (horas) para o processo flexível  $i$ ,

$$5A.3 \quad \sum_{s \in PS(i)}^{NS} T_{istr} \leq h_{it}, \quad \forall i, t, r$$

Todos os fluxos químicos ( $Flux$ ) associados com o processo  $i$  e o esquema de produção  $s$ , considerando também o produto principal  $j'$ , são definidos pelos coeficientes de balanço de massa de uma forma linear,

$$5A.4 \quad Flux_{ijstr} = \mu_{ijs} W_{istr}, \quad \forall i, j, s, t, r$$

Os coeficientes  $\mu_{ijs}$ , referentes a cada componente  $j$  associado ao esquema  $s$  do processo  $i$ , apresentam valores positivos no caso de produção e negativos no de consumo. Para o produto principal  $j'$ , referente ao qual é definido o coeficiente de massa, este apresentará o valor unitário  $+1$ . A variável  $Flux_{ijstr}$  apresentará valores negativos no caso de componentes reagentes, pelo que convirá substituir esta variável no âmbito do modelo de PL/MILP, onde as variáveis são usualmente tratadas como positivas.

Dado que a equação 5A.2 é não linear em  $q$  e  $T$ , será transformada numa equação linear definindo uma nova variável  $\theta$ , da forma

$$5A.5 \quad \theta_{istr} = q_{it} T_{istr}, \quad \forall i, s, t, r$$

Consequentemente, as equações 5A.2 e 5A.3 podem ser reescritas:

$$5A.6 \quad W_{istr} = \rho_{is} \theta_{istr}, \quad \forall i, s, t, r$$

$$5A.7 \quad \sum_{s \in PS(i)} \theta_{istr} \leq h_{it} q_{it}, \quad \forall i, t, r$$

Atendendo à dimensionalidade da variável referente à capacidade dos processos  $Q$  ( $10^6$  kg-ano), o segundo membro deste grupo de desigualdades 5A.7 vai ser tratado de forma a definir a fracção de tempo disponível (por período anual)  $H$ , bem como afinar a dimensionalidade de  $Q$ :

$$5A.8 \quad H_{it} = \frac{h_{it}}{8760}$$

$$5A.9 \quad Q_{it} = 8,760 * 10^{-3} q_{it}$$

Obtém-se, então,

$$5A.10 \quad \sum_{s \in PS(i)} \theta_{istr} \leq H_{it} Q_{it}, \quad \forall i, t, r$$

Estamos interessados nos fluxos agregados,  $AFlux$ , associados com cada processo  $i$  durante o período  $t$ , para todos os esquemas de produção  $s$ :

$$5A.11 \quad AFlux_{ijtr} = \sum_{s \in PS(i)} \mu_{ijs} W_{istr}, \quad \forall i, j, t, r$$

A relação entre as variáveis  $W$  e  $\theta$  é puramente proporcional, referenciando a quantidade produzida (do produto principal  $j'$ ) relativamente à capacidade de produção  $Q$ , pelo que poderemos obter da equação 5A.6:

$$5A.12 \quad AFlux_{ijtr} = \sum_{s \in PS(i)} \mu_{ijs} \rho_{is} \theta_{istr}, \quad \forall i, j, t, r$$

Assim, o balanço global de massa, para cada componente  $j$  em cada período de tempo  $t$  e para cada cenário aleatório  $r$ , corresponde à igualdade entre: os fluxos de produção adicionados às quantidades compradas; e as quantidades vendidas. Isto é,

$$5A.13 \quad \sum_{l=1}^{NM} Pur_{jltr} + \sum_{i=1}^{NP} \sum_{s \in PS(i)} \mu_{ijs} \rho_{is} \theta_{istr} = \sum_{l=1}^{NM} Sal_{jltr} \quad , \quad \forall j, t, r$$

Em termos de VLA de cada cenário aleatório  $r$ , o custo de produção  $\delta_{ist}$  associado a cada esquema  $s$  do processo  $i$  é definido em termos da quantidade produzida  $W_{istr}$  do produto principal  $j'$  associado a esse esquema  $s$ , pelo que atendendo a 5A.6:

$$5A.14 \quad \xi_r = \sum_{j=1}^{NC} \sum_{l=1}^{NM} \sum_{t=1}^{NT} (\gamma_{jlt} Sal_{jltr} - \Gamma_{jlt} Pur_{jltr}) \\ - \sum_{i=1}^{NP} \sum_{s \in PS(i)} \sum_{t=1}^{NT} (\delta_{ist} \rho_{is} \theta_{istr}) - \sum_{i=1}^{NP} \sum_{t=1}^{NT} (\alpha_{it} QE_{it} + \beta_{it} y_{it}), \quad \forall r$$

Para composição das figuras, devem calcular-se diferentes valores esperados em função do factor de penalização,  $\lambda zp$ . Assim, pretende-se representar a variação do desvio esperado,  $E_{dsvn}$ , versus a variação de  $\lambda zp$ , sendo o desvio esperado obtido numericamente através dos resultados para cada instância aleatória  $r$ , da forma:

$$5A.15 \quad E_{dsvn} = \sum_{r=1}^{NR} prob_r dsvn_r$$

Também se pretende determinar a variação do valor esperado para a procura total não-satisfeita,  $E_{proc}$ , obtida da diferença entre os valores da procura,  $Proc$ , e os correspondentes valores das vendas,  $Sal$ ,

$$5A.16 \quad E_{proc} = \sum_{j=1}^{NC} \sum_{l=1}^{NM} \sum_{t=1}^{NT} \sum_{r=1}^{NR} prob_r (Proc_{jltr} - Sal_{jltr})$$

Ainda, deseja-se o valor esperado do excesso de capacidade total,  $E_{zp}$ , aferido através dos valores das variáveis de folga de capacidades,  $Zp$ ,

$$5A.17 \quad E_{zp} = \sum_{i=1}^{NP} \sum_{t=1}^{NT} \sum_{r=1}^{NR} prob_r Zp_{itr}$$

■



## Dados dos exemplos para o modelo estocástico de planejamento de processos

1. Dados do exemplo spplan\_ex1;
2. Dados do exemplo spplan\_ex2;
3. Dados do exemplo spplan\_ex3.

### 1. Dados do exemplo spplan\_ex1

```

SP4flex *** SP: with Flexibility

file=dados_sp4flex.dat
Ilustração do problema "SP4flex: SPplan51b"
SP com flexibilidade: 10 periodos de tempo, 2 processos e 5
produtos
Dados em: Exemplo_SP4flex (JLM)                Resultado: ****

Xest, Tolfes, Toliv:          0.000          1.0000E-06    1.0000E-06
Maxdpt:                        0
Msglvl, Itmax:                 0      2000000000
Maxnod, Intfst:               -1          -1
                                * * *

No. of Processes,              2
No. of Schemes,                2
No. of Time periods,           10
No. of Components,             5
No. of Markets,                1
No. of Scenarios,              4      Max. or min. (+1|-1),      +1
*** CI ***
200.0      200.0      200.0      200.0      200.0      200.0
200.0      200.0      200.0      200.0
*** QIO ***
          0.000      0.000
*** QE_upp ***
100.0      100.0      100.0      100.0      100.0      100.0
100.0      100.0      100.0      100.0
100.0      100.0      100.0      100.0      100.0      100.0
100.0      100.0      100.0      100.0
*** MU ***
          -1.100      0.000      1.000      0.000      0.000
          -1.250      0.000      0.000      1.000      0.000
          0.000      0.000      0.000      -1.120      1.000
          0.000      -1.200      0.000      0.000      1.000

```



```

*** D5_cenarios ***
0.200000000000000000      0.400000000000000000      0.300000000000000000
0.100000000000000000
70.00      74.00      79.00      85.00      92.00      100.0
109.0      119.0      130.0      142.0
70.00      71.00      72.00      74.00      76.00      79.00
82.00      86.00      90.00      94.00
70.00      69.00      68.00      66.00      64.00      61.00
58.00      54.00      50.00      46.00
70.00      67.00      63.00      58.00      52.00      45.00
37.00      26.00      16.00      5.000
*** rho ***
      1.000      1.000
      1.000      1.000
*** H ***
0.9000      0.9000      0.9000      0.9000      0.9000      0.9000
0.9000      0.9000      0.9000      0.9000
0.9000      0.9000      0.9000      0.9000      0.9000      0.9000
0.9000      0.9000      0.9000      0.9000
***

```

## 2. Dados do exemplo spplan\_ex2

```

file=dados_sp7flex2.dat
Ilustracao do problema "SP7flex2: SPplan51b"
SP com flexibilidade: 5 periodos de tempo, 5 processos e 8 produtos
Dados em: dados_sp7flex2.xls (JLM)      Resultado: ****
*
      0. 1.-6 1.-6 xest, tolfes, toliv
      0 0 +0 -1 -1 maxdpt, itmax, msglvl, maxnod, intfst
      5 2 5 8 1 4 +1 np, ns, nt, nc, nm, nr, max/min
      200 200 200 200 200      CI(t)
      0 0 50 0 0      QIO
      100 100 100 100 100      QE(it)upp
      100 100 100 100 100
      100 100 100 100 100
      100 100 100 100 100
      0. -1.1 1. 0. 0. 0. 0. 0. MU(isk) proc:esquema/componente
      -1.35 0. 1. 0. 0. .3 0. 0.
      0. -1.1 1. 0. 0. 0. 0. 0.
      0. 0. 0. 0. 0. 0. 0. 0.
      0. 0. -.8 -0.4 0. 0. 1. 0.
      0. 0. 0. 0. 0. 0. 0. 0.
      0. 0. -.8 -0.4 0. 0. 1. 0.
      0. 0. 0. 0. -1.4 0. 1. .2
      0. 0. 0. 0. -1.4 0. 1. .2
      0. 0. 0. 0. 0. 0. 0. 0.
      0 0 0 0 0      d(jt)upp
      0 0 0 0 0
      0 0 0 0 0
      0 0 0 0 0
      100 120 135 145 152
      150 165 174 182 187
      70 74 79 85 92
      200 200 200 200 200      a(jt)upp
      200 200 200 200 200
      30 30 30 30 30
      200 200 200 200 200

```

---

```

200 200 200 200 200
  0  0  0  0  0
  0  0  0  0  0
  0  0  0  0  0
0.120 0.114 0.109 0.104 0.099 alfa(it)
0.150 0.143 0.136 0.130 0.123
0.100 0.095 0.091 0.086 0.082
0.100 0.095 0.091 0.086 0.082
0.080 0.076 0.073 0.069 0.066
30.000 28.571 27.211 25.915 24.681 beta(it)
15.000 14.286 13.605 12.958 12.341
20.000 19.048 18.141 17.277 16.454
23.000 21.905 20.862 19.868 18.922
25.000 23.810 22.676 21.596 20.568
0.100 0.095 0.091 0.086 0.082 delta(ist)
0.100 0.095 0.091 0.086 0.082
0.150 0.143 0.136 0.130 0.123
1.+6 1.+6 1.+6 1.+6 1.+6
0.200 0.190 0.181 0.173 0.165
1.+6 1.+6 1.+6 1.+6 1.+6
0.180 0.171 0.163 0.155 0.148
0.180 0.171 0.163 0.155 0.148
0.170 0.162 0.154 0.147 0.140
1.+6 1.+6 1.+6 1.+6 1.+6
  0  0  0  0  0 gama(jt) gasal
  0  0  0  0  0
  0  0  0  0  0
  0  0  0  0  0
  0  0  0  0  0
1.61 1.71 1.79 1.86 1.91
2.00 2.10 2.18 2.25 2.30
1.30 1.24 1.18 1.14 1.10
0.51 0.57 0.61 0.64 0.66 GAMA(jt) gapur
0.70 0.64 0.60 0.57 0.55
1.00 1.09 1.17 1.24 1.29
0.42 0.46 0.50 0.53 0.55
0.55 0.59 0.65 0.68 0.70
1.+6 1.+6 1.+6 1.+6 1.+6 (big M e nao 0)
1.+6 1.+6 1.+6 1.+6 1.+6
1.+6 1.+6 1.+6 1.+6 1.+6
0.3 0.4 0.2 0.1 probr(r)
170 174 179 185 192 dcen(7tr)
170 171 172 174 176
170 169 168 166 164 46
170 167 163 158 152
  1.0 1.0 rho(is)
  1.0 1.0
  1.0 1.0
  1.0 1.0
  1.0 1.0
0.9 0.9 0.9 0.9 0.9 H(it)
0.9 0.9 0.9 0.9 0.9
0.9 0.9 0.9 0.9 0.9
0.9 0.9 0.9 0.9 0.9
0.9 0.9 0.9 0.9 0.9

```

\*\*\*

### 3. Dados do exemplo spplan\_ex3

```

file=dados_sp7flex3.dat
Ilustracao do problema "SP4flex: SPplan51b"
SP com flexibilidade: 5 periodos de tempo, 5 processos e 8 produtos
Dados em: dados_ex3_sp5.xls (JLM)                Resultado: ****
*
    0. 1.-6 1.-6 xest, tolfes, toliv
      0 0 +0 -1 -1 maxdpt, itmax, msglvl, maxnod, intfst
5 2 5 8 1 4 +1 np, ns, nt, nc, nm, nr, max/min
200 200 200 200 200 CI(t)
      0 0 50 0 0 QIO
100 100 100 100 100 QE(it)upp
100 100 100 100 100
100 100 100 100 100
100 100 100 100 100
100 100 100 100 100
-1.1 0. 0. 1. 0. 0. 0. 0. MU(isk) proc:esquema/componente
 0. 0. 0. 0. 0. 0. 0. 0.
-1.1 0. 0. 1. 0. 0. 0. 0.
 0. -1.4 0. 0. 1. .3 0. 0.
 0. -1.4 0. 0. 1. .3 0. 0.
 0. 0. 0. 0. 0. 0. 0. 0.
 0. 0. 0. 0. 0. -1.2 1. 0.
 0. 0. 0. 0. 0. 0. 0. 0.
 0. 0. 0. 0. 0. -1.2 1. 0.
 0. 0. -.4 0. 0. -.8 0. 1.
 0 0 0 0 0 d(jt)upp
 0 0 0 0 0
 0 0 0 0 0
100 120 135 145 152
200 210 219 227 236
 10 10 10 10 10
150 165 174 182 187
 70 74 79 85 92
200 200 200 200 200 a(jt)upp
200 200 200 200 200
200 200 200 200 200
 0 0 0 0 0
 0 0 0 0 0
 0 0 0 0 0
 0 0 0 0 0
0.120 0.114 0.109 0.104 0.099 alfa(it)
0.150 0.143 0.136 0.130 0.123
0.100 0.095 0.091 0.086 0.082
0.080 0.076 0.073 0.069 0.066
0.100 0.095 0.091 0.086 0.082
30.000 28.571 27.211 25.915 24.681 beta(it)
15.000 14.286 13.605 12.958 12.341
20.000 19.048 18.141 17.277 16.454
23.000 21.905 20.862 19.868 18.922
25.000 23.810 22.676 21.596 20.568
0.100 0.095 0.091 0.086 0.082 delta(ist)
1.+6 1.+6 1.+6 1.+6 1.+6
0.150 0.143 0.136 0.130 0.123
0.150 0.143 0.136 0.130 0.123
0.200 0.190 0.181 0.173 0.165
1.+6 1.+6 1.+6 1.+6 1.+6
0.180 0.171 0.163 0.155 0.148
1.+6 1.+6 1.+6 1.+6 1.+6
0.170 0.162 0.154 0.147 0.140
0.170 0.162 0.154 0.147 0.140

```

---

```

0      0      0      0      0      gama(jt) gasal
0      0      0      0      0
0      0      0      0      0
1.61  1.71  1.79  1.86  1.91
2.00  2.10  2.18  2.25  2.30
0.55  0.59  0.65  0.68  0.70
1.01  1.09  1.17  1.24  1.29
1.30  1.24  1.18  1.14  1.10
0.51  0.57  0.61  0.64  0.66  GAMA(jt) gapur
0.70  0.64  0.60  0.57  0.55
0.42  0.46  0.50  0.53  0.55
1.+6  1.+6  1.+6  1.+6  1.+6  (big M e nao 0)
1.+6  1.+6  1.+6  1.+6  1.+6
1.+6  1.+6  1.+6  1.+6  1.+6
1.+6  1.+6  1.+6  1.+6  1.+6
1.+6  1.+6  1.+6  1.+6  1.+6
0.3   0.4   0.2   0.1  probr(r)
200   204   209   215   222  dcen(5tr)
200   201   202   204   206
200   199   198   196   194
200   197   193   188   182
1.0   1.0   rho(is)
1.0   1.0
1.0   1.0
1.0   1.0
1.0   1.0
0.9   0.9   0.9   0.9   0.9  H(it)
0.9   0.9   0.9   0.9   0.9
0.9   0.9   0.9   0.9   0.9
0.9   0.9   0.9   0.9   0.9
0.9   0.9   0.9   0.9   0.9

```

\*\*\*



## Decomposição de Benders para o modelo estocástico de planeamento de processos

Para minorar as dificuldades computacionais, aplica-se um esquema de decomposição de Benders, especificamente direccionado para o problema do planeamento de processos com robustez. Em cada iteração  $k$ , estimam-se as variáveis associadas às expansões de capacidades ( $y$ ,  $QE$ ,  $Q$ ) através da resolução de um problema mestre, variáveis essas que representam as variáveis de projecto do 2SSP. As variáveis de controlo ( $Sal$ ,  $Pur$ ,  $\theta$ ) correspondentes ao plano operacional são consideradas no subproblema. O subproblema gera o limite inferior ( $LB$ ) para o VLA, enquanto que o problema mestre gera o limite superior ( $UB$ ), continuando-se o processo iterativo até que a diferença entre estes dois limites seja inferior a uma dada tolerância.

No enquadramento referido, a incorporação das restrições de corte relativas à “produção de capacidade” torna-se crucial para a delimitação do espaço de procura em primeira fase. De facto, o problema mestre em primeira fase constitui um problema MILP, o qual apresenta melhores características de resolução com a introdução de constrangimentos para as variáveis de expansão de capacidades,  $QE$ , associadas às variáveis binárias,  $y$ .

A desigualdade referente ao “corte de Benders”, que quantifica a  $UB$  para o valor-objectivo do programa estocástico, inclui uma parcela lagrangeana respeitante aos processos com capacidades subutilizadas:

$$\begin{aligned}
 \eta \leq & \sum_{r=1}^{NR} prob_r \xi_r - \lambda ds v \sum_{r=1}^{NR} prob_r . ds v n_r \\
 \text{5C.1} \quad & - \lambda z p \sum_{i=1}^{NP} \sum_{t=1}^{NT} \sum_{r=1}^{NR} Z p_{itr}^k + \sum_{i=1}^{NP} \sum_{t=1}^{NT} \sum_{r=1}^{NR} \lambda_{rit}^k \left( \sum_{s \in PS(i)} \theta_{isrt}^k - H_{ij} Q_{it} \right)
 \end{aligned}$$

Dada a focalização do problema mestre na determinação das expansões de capacidades e das variáveis binárias correspondentes, é aqui directamente aplicável o conjunto de restrições que desagregam a “produção de capacidade” e permitem reduzir (no algoritmo *branch-and-bound*) a extensão do espaço de procura inerente à relaxação das restrições de integralidade, ou seja, incorpora planos de corte associados à reformulação da subestrutura ELS:

**Modelo *plan\_milp\_spflexmest*:**

$$5C.2-a \quad [max] \quad \eta$$

sujeito a

$$5C.2-b \quad \eta - \sum_{r=1}^{NR} prob_r \xi_r + \lambda dsv \sum_{r=1}^{NR} prob_r . dsv_r - \\ \sum_{i=1}^{NP} \sum_{t=1}^{NT} \sum_{r=1}^{NR} \lambda_{itr}^k \left( \sum_{s \in PS(i)} \theta_{istr}^k - H_{ij} Q_{it} \right) \leq -\lambda zp \sum_{i=1}^{NP} \sum_{t=1}^{NT} \sum_{r=1}^{NR} Zp_{itr}^k$$

$$5C.2-c \quad dsv_r - \sum_{r'=1}^{NR} (prob_{r'} \xi_{r'}) + \xi_r \geq 0, \quad \forall r$$

$$5C.2-d \quad \xi_r + \sum_{i=1}^{NP} \sum_{t=1}^{NT} (\alpha_{it} QE_{it} + \beta_{it} y_{it}) = \sum_{j=1}^{NC} \sum_{l=1}^{NM} \sum_{t=1}^{NT} (\gamma_{jlt} Sal_{jlt}^k - \Gamma_{jlt} Pur_{jlt}^k) \\ - \sum_{i=1}^{NP} \sum_{s \in PS(i)} \sum_{t=1}^{NT} (\delta_{ist} \rho_{is} \theta_{istr}^k), \quad \forall r$$

$$5C.2-e \quad Q_{it-1} + QE_{it} - Q_{it} = 0, \quad \forall i, t$$

$$5C.2-f \quad QE_{it} - QE_{it}^{Upp} y_{it} \leq 0, \quad \forall i, t$$

$$5C.2-g \quad \sum_{i=1}^{NP} (\alpha_{it} QE_{it} + \beta_{it} y_{it}) \leq CI(t), \quad \forall t$$

$$5C.2-h \quad \varphi_{it\tau} - QE_{it} \leq 0, \quad \forall i, t, \tau \geq t$$

$$5C.2-i \quad \varphi_{it\tau} - QE_{it\tau}^{acum} y_{it} \leq 0, \quad \forall i, t, \tau \geq t$$

$$5C.2-j \quad \sum_{\tau=1}^t \varphi_{it\tau} - \left( \sum_{s \in PS(i)} \theta_{istr}^k - H_{ij} Q_{i0} \right) \geq 0, \quad \forall i, t, \tau \leq t$$

$$5C.2-k, l \quad \eta, \xi_r, dsv_r, QE_{it}, Q_{it}, \varphi_{it\tau} \geq 0; \quad y_{it} \in \{0,1\}, \quad \forall i, t$$

O modelo *plan\_milp\_spflexmest* apresenta:  $(1+2.NR+2.NP.NT+NP.NT.NT.(NT+1)/2)$  variáveis contínuas, correspondentes à variável de “corte” de Benders, ao VLA e aos desvios, às capacidades dos processos e respectivas expansões de capacidade, às variáveis desagregadas de “produção de capacidade”;  $(NP.NT)$  variáveis discretas, respeitantes às decisões binárias quanto às expansões de cada processo em cada período de tempo;  $(1+2.NR+2.NP.NT+NT+3.NP.NT.NT.(NT+1)/2)$  restrições, respectivamente, associadas ao “corte de Benders”, à definição do VLA e desvios, aos balanços de capacidades e às restrições lógicas sobre as quantidades a expandir, aos limites ao investimento e às restrições de corte de desagregação de variáveis de “produção de capacidade”.

Encontrando-se assim definidas as decisões quanto à expansão de capacidades ( $y$ ,  $QE$ ,  $Q$ ) e configurada a rede de processos químicos, considera-se então —em cada iteração  $k$  da segunda fase— o subproblema de planeamento operacional com as variáveis de controlo ( $Sal$ ,  $Pur$ ,  $\theta$ ):

**Modelo *plan\_lp\_spflexsub*:**

$$5C.3-a \quad [max] \Phi = \sum_{r=1}^{NR} prob_r \xi_r - \lambda dsv \sum_{r=1}^{NR} prob_r \cdot dsvn_r - \lambda zp \sum_{i=1}^{NP} \sum_{t=1}^{NT} \sum_{r=1}^{NR} Zp_{itr}$$

sujeito a

$$5C.3-b \quad dsvn_r - \sum_{r'=1}^{NR} (prob_{r'} \xi_{r'}) + \xi_r \geq 0, \quad \forall r$$

$$5C.3-c \quad \xi_r - \sum_{j=1}^{NC} \sum_{l=1}^{NM} \sum_{t=1}^{NT} (\gamma_{jlt} Sal_{jltr} - \Gamma_{jlt} Pur_{jltr}) + \sum_{i=1}^{NP} \sum_{s \in PS(i)} \sum_{t=1}^{NT} (\delta_{ist} \rho_{is} \theta_{istr}) = - \sum_{i=1}^{NP} \sum_{t=1}^{NT} (\alpha_{it} QE_{it}^k + \beta_{it} y_{it}^k), \quad \forall r$$

$$5C.3-d \quad \sum_{s \in PS(i)} \theta_{istr} + Zp_{itr} = H_{it} Q_{it}^k, \quad \forall i, t, r$$

$$5C.3-g \quad \sum_{l=1}^{NM} Pur_{jltr} + \sum_{i=1}^{NP} \sum_{s \in PS(i)} \mu_{ijs} \rho_{is} \theta_{istr} - \sum_{l=1}^{NM} Sal_{jltr} = 0, \quad \forall j, l, t, r$$

$$5C.3-h \quad Pur_{jltr} \leq a_{jltr}^{Upp}, \quad \forall j, l, t, r$$

$$5C.3-i \quad Sal_{jltr} \leq d_{jltr}^{Upp}, \quad \forall j, l, t, r$$

$$5C.3-j \quad \xi_r, dsvn_r, Sal_{jltr}, Pur_{jltr}, \theta_{istr}, Zp_{itr} \geq 0, \quad \forall i, s, j, l, t, r$$

Este modelo *plan\_lp\_spflexsub* apresenta:

- $(2.NR+2.NC.NM.NT.NR+ 2.NP.NS.NT.NR+NP.NT.NR)$  variáveis contínuas, correspondentes ao VLA e desvios, quantidades vendidas e compradas, aos níveis de fluxo dos processos, às folgas das capacidades aceitáveis pela formulação (note-se que as variáveis binárias do problema original, respeitantes às decisões de expansão, e as capacidades dos processos são tratadas como parâmetros neste subproblema);
- $(2.NR+NP.NT.NR+ NC.NT.NR+2.NC.NM.NT.NR)$  restrições, associadas às restrições para definição do VLA e dos desvios, da definição das folgas das capacidades, dos balanços globais aos componentes, bem como os constrangimentos às quantidades de compra e de venda.

Segue-se o algoritmo de decomposição de Benders, adaptado de Liu *et al.* [1996]:

- |    |   |
|----|---|
| 1) | $k = 1$<br><i>seleccionar</i> : $y_{it}^k, Q_{it}^k, QE_{it}^k$<br>$UB = +\infty; LB = -\infty$   |
| 2) | <i>Resolver subproblema</i> <b>plan_lp_sflexsub(k)</b><br><i>solução</i> : $\theta_{rist}^k, Sal_{rjlt}^k, Pur_{rjlt}^k, Zp_{rit}^k$<br>$LB = \max(LB, \Phi^k)$ |
| 3) | <i>Resolver problema</i> <b>plan_milp_sflexmest(k)</b><br><i>obter</i> : $y_{it}^k, Q_{it}^k, QE_{it}^k$<br>$UB = \eta^k$                                       |
| 4) | <i>Se</i><br>$(UB - LB) \leq \text{tolerância}$ então <b>PARAR!</b><br><i>caso contrário</i><br>$k = k + 1$ ; <i>voltar a</i> 2)                                |

■

# Aspectos da modelação para dimensionamento e sequenciamento de processos descontínuos

A generalização pretendida envolve a consideração de duas novas vertentes: uma estocástica, com o dimensionamento dos processos descontínuos associada à satisfação óptima de procuras incertas, num só horizonte temporal (“estática”) onde se minimizam os custos de investimento; outra, designada “dinâmica”, considerando simultaneamente a incerteza e um ambiente multiperíodo, onde se pretende a maximização do valor actualizado.

A incerteza na procura  $Q$ , cenário-dependente, reflecte-se na exigência da sua satisfação integral: enquanto que no modelo determinístico, a necessidade de satisfazer obrigatoriamente a procura vai impor a necessidade da capacidade dos equipamentos, num modelo estocástico tal imposição poderia levar à especificação de dimensões elevadas, mas dirigidas por cenários pouco prováveis; a modelação da quantidade produzida  $W$  utilizando uma restrição *soft*, a qual ocorre em simultâneo com a definição da procura não-satisfeita  $Qns$ , definida como a folga da restrição respeitante e a ser penalizada na função objectivo robusta:

$$5D.1 \quad W_{jtr} \leq Q_{jtr} \Rightarrow W_{jtr} + Qns_{jtr} = Q_{jtr}, \quad \forall j, t, r$$

A utilização da variável  $W_{jtr}$ , relativa às quantidades produzidas, em alternativa ao parâmetro do problema  $Q_{jtr}$ , respeitante às quantidades procuradas, no âmbito das restrições usuais para determinação do número de batches  $n_{jtr}$ , origina não-linearidades:

$$5D.2 \quad S_{ij} W_{jtr} \sum_{s=1}^{NS(i)} \sum_{p=1}^{NP(i)} \frac{y_{isp}}{dv_{is}} = n_{jtr}, \quad \forall i, j, t, r$$

Avaliando a introdução de uma variável auxiliar,  $Waux$ , agregando o produto de variáveis,

$$5D.3 \quad W_{jtr} \cdot y_{isp} = Waux_{ijsptr}, \quad \forall i, j, s, p, t, r$$

tal envolveria a consideração simultânea dos seguintes grupos de restrições:

$$5D.4 \quad Waux_{ijsptr} \leq Waux_{ijsp}^{Upp} y_{isp}, \quad \forall i, j, s, p, t, r$$

$$5D.5 \quad \sum_{s=1}^{NS(i)} \sum_{p=1}^{NP(i)} y_{isp} = 1, \quad \forall i$$

$$5D.6 \quad \sum_{s=1}^{NS(i)} \sum_{p=1}^{NP(i)} Waux_{ijsptr} = W_{jtr}, \quad \forall i, j, t, r$$

Desta maneira, o número, tanto de variáveis (em número de  $M.N.NS.NP.NT.NR$ ) como de restrições (em  $M.N.NS.NP.NT.NR+M.N.NT.NR$ ), viria significativamente aumentado, além de que ocorreria uma coincidência de âmbito de definição com a seguinte variável agregadora que é usualmente utilizada (conforme secção 2.2),

$$5D.7 \quad n_{jtr} \cdot y_{isp} = nc_{ijsptr}, \quad \forall i, j, s, p, t, r$$

Ora, recorrendo à formulação original [Kocis e Grossmann, 1988], onde se estabelece a restrição da dimensão dos *batches*,  $B_{jtr}$ , em função dos volumes  $V_i$  implementados, associando-lhe a discretização dos volumes e relacionando com as quantidades produzidas,  $W_{jtr}$ , obtém-se

$$5D.8 \quad \begin{aligned} S_{ij} B_{jtr} &\leq V_i, \quad \forall i, j, t, r \Rightarrow S_{ij} B_{jtr} \leq \sum_{s=1}^{NS(i)} \sum_{p=1}^{NP(i)} dv_{is} y_{isp} \Rightarrow \\ S_{ij} \frac{W_{jtr}}{n_{jtr}} &\leq \sum_{s=1}^{NS(i)} \sum_{p=1}^{NP(i)} dv_{is} y_{isp} \Rightarrow S_{ij} W_{jtr} \leq \sum_{s=1}^{NS(i)} \sum_{p=1}^{NP(i)} dv_{is} y_{isp} n_{jtr} \Rightarrow \\ S_{ij} W_{jtr} &\leq \sum_{s=1}^{NS(i)} \sum_{p=1}^{NP(i)} dv_{is} nc_{ijsptr}, \quad \forall i, j, t, r \end{aligned}$$

Assim, são utilizadas as variáveis semicontínuas,  $nc_{ijsptr}$ , evitando as variáveis auxiliares,  $Waux_{ijsptr}$ , bem como as restrições inerentes.

No modelo robusto interessa também penalizar as folgas de capacidade. Ora as folgas das restrições,  $folg_{ijtr}$ , ocorrendo em cada estágio e relativamente à quantidade (global) produzida,  $W_{jtr}$ , em cada período e cenário, que se definem através de

$$5D.9 \quad S_{ij} W_{jtr} + folg_{ijtr} = \sum_{s=1}^{NS(i)} \sum_{p=1}^{NP(i)} dv_{is} nc_{ijsptr}, \quad \forall i, j, t, r$$

correspondem à agregação das folgas ocorridas no âmbito de todos os ciclos de produção, pelo que será conveniente relacionar tais variáveis,  $folg_{ijtr}$ , com a folga efectiva,  $Zp_{ijtr}$ , a qual ocorre na produção de cada *batch*. Então,

$$5D.10 \quad \begin{aligned} S_{ij} \frac{W_{jtr}}{n_{jtr}} + \frac{folg_{ijtr}}{n_{jtr}} &= \sum_{s=1}^{NS(i)} \sum_{p=1}^{NP(i)} dv_{is} y_{isp} \Rightarrow \\ S_{ij} B_{jtr} + Zp_{ijtr} &= \sum_{s=1}^{NS(i)} \sum_{p=1}^{NP(i)} dv_{is} y_{isp} \end{aligned}$$

sendo a folga efectivamente verificada nas capacidades implementadas,  $Zp_{ijtr}$ , obténivel por desagregação da folga global,  $folg_{ijtr}$ , relativamente ao número de *batches* efectivamente produzidos,  $n_{jtr}$ :

$$5D.11 \quad Zp_{ijtr} = \frac{folg_{ijtr}}{n_{jtr}}, \quad \forall i, j, t, r$$

Esta relação entre duas variáveis impossibilita a consideração do parâmetro efectivo da folga de capacidades,  $Zp_{ijtr}$ , no âmbito da respectiva penalização na função objectivo robusta, mas possibilita a sua utilização quanto à avaliação efectiva da folga de capacidades que se verifica na configuração implementada.

Em seguida, procede-se a uma análise ao modelo *sbatch\_milp\_fsmz*, considerando os seguintes aspectos:

- supondo  $NT=NR=1$ , e com  $prob(1)=1$ , os índices  $t$  e  $r$  poderão ser negligenciados;
- dado que  $NR=1$ , os desvios serão nulos,

$$5D.12 \quad dsv_n_r = \sum_{r'=1}^1 (prob_{r'} \xi_{r'}) - \xi_r = 0$$

- fazendo  $\lambda qns = bigM$ , através do processo de optimização virá que,

$$5D.13 \quad Qns_i = 0. \Rightarrow W_i = Q_i, \quad \forall i$$

- especificando  $\lambda flg = 0$ . (não penalizando as folgas de capacidade) e  $ret_{jtr}=0$ . (não valorizando os retornos), então a função objectivo assumirá a seguinte forma de minimização dos custos de investimento,

$$5D.14 \quad \max \Phi = \max \xi_r = \max \left\{ - \sum_{j=1}^M \sum_{s=1}^{NS(i)} \sum_{p=1}^{NP(i)} c_{isp} y_{isp} \right\} = \min \left\{ \sum_{j=1}^M \sum_{s=1}^{NS(i)} \sum_{p=1}^{NP(i)} c_{isp} y_{isp} \right\}$$

- os restantes grupos de restrições equivalem-se às restrições em *batch\_milp\_fsmz*, como se pode observar directamente; logo, este problema em *sbatch\_milp\_fsmz* também é NP-Hard;
- assim, a análise conduzida para o problema *batch\_milp\_fsmz*, determinístico e estático (com um só período de tempo), manter-se-á válida para cada um dos cenários e em cada período de tempo do problema estocástico e multiperíodo *sbatch\_fsmz*;
- da restrição *sb.13-g*, apenas se poderá seleccionar uma dimensão  $s'(i)$  em número  $p'(i)$ , em qualquer dos estágios  $i$ ;

$$5D.15 \quad \begin{cases} y_{isp} = 0, & \forall s \neq s'(i), p \neq p'(i) \\ y_{i,s'(i),p'(i)} = 1, & \exists^1(s'(i), p'(i)) \end{cases}$$

- da conjunção das restrições *sb.13-f* e *sb.13-h*, observa-se que todas as variáveis  $nc_{ijsp}$  serão nulas, excepto aquelas correspondentes à dimensão  $s'$  e número  $p'$  seleccionados, igualando assim o valor de  $n_j$  ao longo de todos os estágios  $i$ ; ainda, supõe-se que o limite superior lógico da restrição *sb.13-f* seja trivialmente verificado ( $nc^{Upp} = bigM$ ), originando

$$5D.16 \quad \begin{cases} nc_{ijsp} = 0, \quad \forall s \neq s'(i), \quad p \neq p'(i) \\ nc_{ij,s'(i),p'(i)} = n_j, \quad \exists^1(s'(i), p'(i)) \end{cases}$$

- da restrição *sb.13-e*, observa-se que na definição dos números de *batches*  $n_j$  (para qualquer produto  $j$ ) deve satisfazer-se todos os estágios  $i$ ; assim, o valor de  $n_j$  será correspondente ao máximo dos valores obtidos segundo o índice  $i$ ,

$$5D.17 \quad S_{ij}W_j \leq p'(i)dv_{i,s'(i)}n_j, \quad \forall i, j \Rightarrow n_j = \max_i \left\{ \frac{S_{ij}W_j}{p'(i)dv_{i,s'(i)}} \right\}, \quad \forall i$$

- na restrição em *sb.13-i*, o tempo de campanha  $tcamp_j$  para cada produto  $j$  deve observar todo o estágio  $i$ , pelo que corresponderá ao máximo dos valores obtidos para o somatório; neste somatório, os números de ciclos  $nc_{ijsp}$  encontram-se restringidos a  $n_j$  em todo o estágio  $i$ , conforme as restrições *sb.13-f* e *sb.13-h*,

$$5D.18 \quad tcamp_j = \max_i \left\{ \frac{\tau_{ij}}{p'(i)} n_j \right\} = \max_i \left\{ \frac{\tau_{ij}S_{ij}W_j}{p'(i)dv_{i,s'(i)}} \right\}, \quad \forall j$$

- na restrição em *sb.13-j*, a soma dos valores dos tempos de campanha  $tcamp_j$  deve observar o horizonte temporal  $H$ , pelo que,

$$5D.19 \quad \sum_{j=1}^{NC} \max_i \left\{ \frac{\tau_{ij}S_{ij}W_j}{p'(i)dv_{i,s'(i)}} \right\} \leq H$$

Ora, o objectivo é o de maximizar com robustez o valor líquido actual, pelo que deve ser escolhido o grupo de equipamentos ( $s'$ ,  $p'$ ) que: satisfaz as restrições em cada estágio  $i$  e para cada produto  $j$ , relativas a todos os cenários ao longo de todos os períodos de tempo; permite a maior produção possível,  $W_{jtr}$ , com vertente de valorização do retorno, bem como modela a satisfação da procura (aleatória); pretende diminuir a variabilidade, penalizando os desvios negativos do VLA, bem como as folgas da capacidade instalada.

### Exemplo numérico *spbatch\_ex1*

Seja a situação problemática onde se pretende proceder ao dimensionamento dos processos descontínuos, considerando o respectivo sequenciamento com campanhas de produto único (SPC) e política de não-espera (*zero wait*), mas de forma a satisfazer a procura incerta dos componentes a produzir, num só período ou horizonte temporal (problema “estático”). Esta procura é modelizada através de um conjunto discreto de cenários, aos quais é associada uma probabilidade de ocorrência, sendo penalizada a procura não-satisfeita.

A finalidade aqui não é a de maximizar o VLA, dado que se considera apenas um período temporal, sendo oVLA usualmente considerado no âmbito multiperíodo (problema “dinâmico”), dados os elevados custos de investimento envolvidos e que não possibilitarão o *payback* num só período temporal. Então, não se considerando os valores de retorno, formulando-se  $ret_{jtr}=0.$ , obtem-se um mesmo valor para qualquer variável  $\xi$  e que representa apenas os custos de investimento,

$$5D.20 \quad \xi = \xi_r = - \sum_{i=1}^M \sum_{s=1}^{NS(i)} \sum_{p=1}^{NP(i)} c_{isp} y_{isp}, \quad \forall r$$

Sendo o valor de  $\xi$  cenário-independente, não se verificarão desvios nesta variável, pelo que se obterá em todo e qualquer cenário,  $dsvn_r = 0$ ; dado que o propósito desta optimização robusta é a satisfação das procuras incertas, não se penalizam as folgas de capacidade,  $\lambda flg=0$ , pelo que a função objectivo em *sp.13-a* se reduz à minimização robusta dos custos de investimento, incluindo-se apenas as penalidades pela procura não-satisfeita,

$$5D.21 \quad \begin{aligned} [max] \Phi &= \xi \cdot \sum_{r=1}^{NR} prob_r - \lambda qns \sum_{r=1}^{NR} \frac{prob_r}{NC \cdot NT} \left( \sum_{j=1}^{NC} \sum_{t=1}^{NT} Qns_{jtr} \right) \\ &= - \sum_{i=1}^M \sum_{s=1}^{NS(i)} \sum_{p=1}^{NP(i)} c_{isp} y_{isp} - \lambda qns \sum_{r=1}^{NR} \frac{prob_r}{NC \cdot NT} \left( \sum_{j=1}^{NC} \sum_{t=1}^{NT} Qns_{jtr} \right) \end{aligned}$$

isto é,

$$5D.22 \quad [min] \Psi = \sum_{j=1}^M \sum_{s=1}^{NS(i)} \sum_{p=1}^{NP(i)} c_{isp} y_{isp} + \lambda qns \sum_{r=1}^{NR} \frac{prob_r}{NC \cdot NT} \left( \sum_{j=1}^{NC} \sum_{t=1}^{NT} Qns_{jtr} \right)$$

Assim, utiliza-se o modelo definido pelo conjunto das equações em *sb.13*, onde se pode negligenciar: o índice dos períodos temporais,  $t$ , já que se considera apenas um período de tempo; os grupos de restrições referentes à definição das variáveis-valor,  $\xi$ , que apresentarão valor cenário-independente, e dos respectivos desvios,  $dsvn$ , que consequentemente serão nulos.

### Exemplo numérico *spbatch\_ex2*

Trata-se a situação generalizada do dimensionamento e sequenciamento dos processos descontínuos, visando não só o objectivo parcial da satisfação da procura dos diferentes componentes, mas antes a usual maximização do valor actual líquido, VLA, considerando vários períodos temporais (problema “dinâmico”). A procura incerta é definida por um conjunto discreto de cenários, com uma dada probabilidade, sendo penalizados: os desvios por variabilidade do VAL; as folgas por excesso de capacidade.

Quanto à penalização da procura não-satisfeita, atente-se que

$$5D.23 \quad Qns_{jtr} = Q_{jtr} - W_{jtr} \geq 0, \quad \forall j, t, r$$

e se na função objectivo robusta considerarmos, sem perda de generalidade e por economia textual, apenas as parcelas referentes ao VLA e à penalização por procura não-satisfeita, obter-se-á

$$5D.24 \quad \begin{aligned} [max] \Phi &= \sum_{r=1}^{NR} prob_r \xi_r - \lambda qns \sum_{r=1}^{NR} \frac{prob_r}{NC \cdot NT} \left( \sum_{j=1}^{NC} \sum_{t=1}^{NT} Qns_{jtr} \right) \\ &= \sum_{r=1}^{NR} prob_r \left( \sum_{j=1}^{NC} \sum_{t=1}^{NT} ret_{jtr} W_{jtr} - \sum_{j=1}^M \sum_{s=1}^{NS(i)} \sum_{p=1}^{NP(i)} c_{isp} y_{isp} \right) - \lambda qns \sum_{r=1}^{NR} \frac{prob_r}{NC \cdot NT} \left( \sum_{j=1}^{NC} \sum_{t=1}^{NT} (Q_{jtr} - W_{jtr}) \right) \\ &= - \sum_{j=1}^M \sum_{s=1}^{NS(i)} \sum_{p=1}^{NP(i)} c_{isp} y_{isp} + \sum_{r=1}^{NR} prob_r \left( \sum_{j=1}^{NC} \sum_{t=1}^{NT} \left( ret_{jtr} + \frac{\lambda qns}{NC \cdot NT} \right) W_{jtr} \right) - \lambda qns \sum_{r=1}^{NR} \frac{prob_r}{NC \cdot NT} \left( \sum_{j=1}^{NC} \sum_{t=1}^{NT} Q_{jtr} \right) \end{aligned}$$

ou seja, a solução obtida é equivalente à que seria obtida por maximização de:

$$5D.25 \quad [\max] \Phi = -\sum_{j=1}^M \sum_{s=1}^{NS(i)NP(i)} \sum_{p=1}^{NP(i)} c_{isp} \mathcal{Y}_{isp} + \sum_{r=1}^{NR} prob_r \left( \sum_{j=1}^{NC} \sum_{t=1}^{NT} \left( ret_{jtr} + \frac{\lambda qns}{NC \cdot NT} \right) W_{jtr} \right)$$

De facto, atendendo a que os parâmetros aleatórios  $Q_{jtr}$  não são passíveis de optimização, pois são dados do problema sob incerteza, a penalização da procura não-satisfeita traduz-se num aumento efectivo dos valores de retorno  $ret_{jtr}$  envolvidos. Donde, optou-se por não penalizar a procura não-satisfeita neste exemplo *spbatch\_ex2*, fazendo-se  $\lambda qns=0.$ , embora se avalie a respeitante evolução no decorrer das diversas instâncias numéricas.

### Estimadores técnicos e económicos

$$5D.26 \quad Eproc = \sum_{r=1}^{NR} \frac{prob_r}{NC \cdot NT} \left( \sum_{j=1}^{NC} \sum_{t=1}^{NT} Qns_{jtr} \right)$$

$$5D.27 \quad Efolg = \sum_{r=1}^{NR} \frac{prob_r}{M \cdot NC \cdot NT} \sum_{j=1}^{NC} \sum_{t=1}^{NT} \left\{ \sum_{i=1}^M \sum_{s=1}^{NS} \sum_{p=1}^{NP} p(i) \cdot y_{isp} \cdot \left( dv_{js} - S_{ij} \cdot \frac{W_{jtr}}{n_{jtr}} \right) \right\}$$

$$5D.28 \quad \%Eproc = \frac{Eproc}{Qmed} \cdot 100 \quad , \quad \text{com } Qmed = \sum_{r=1}^{NR} \frac{prob_r}{NC \cdot NT} \left( \sum_{j=1}^{NC} \sum_{t=1}^{NT} Q_{jtr} \right)$$

$$5D.29 \quad \%Efolg = \frac{Efolg}{Vtotal} \cdot 100 \quad , \quad \text{com } Vtotal = \sum_{i=1}^M \sum_{s=1}^{NS} \sum_{p=1}^{NP} (y_{isp} \cdot dv_{is})$$

$$5D.30 \quad \%Benef = \frac{Ecsi + Ctotal}{Ctotal} \cdot 100 \quad , \quad \text{com } Ctotal = \sum_{i=1}^M \sum_{s=1}^{NS} \sum_{p=1}^{NP} (y_{isp} \cdot c_{isp})$$

$$payback = t'-1 + \frac{Ctotal - \sum_{t=1}^{t'-1} Ecash(t)}{Ecash(t')} \quad ,$$

$$5D.31 \quad \text{com } \begin{cases} Ecash(t) = \sum_{j=1}^{NC} \sum_{r=1}^{NR} (prob_r \cdot ret_{jtr} \cdot W_{jtr}) \\ \exists^1 t' : \sum_{t=1}^{t'} Ecash(t) \geq Ctotal > \sum_{t=1}^{t'-1} Ecash(t) \end{cases}$$

Obtem-se uma estimativa para a TIR, actualizando os fluxos de retorno ao período inicial,  $Ecash0$ , obtendo-se esta estimativa logo que,

$$5D.32 \quad \sum_{t=1}^{NT} Ecash0(t) \leq Ctotal,$$

$$\text{com } Ecash0(t) = Ecash(t) \cdot \frac{(1 + txint)^t}{(1 + TIR)^t}$$

■

# Dados dos exemplos para o dimensionamento e sequenciamento de processos descontínuos

1. Dados para instâncias do exemplo *spbatch\_ex1*;
2. Estimadores económicos para instâncias do exemplo *spbatch\_ex2*.

### 1. Dados para instâncias do exemplo *spbatch\_ex1*

Compreensivamente, descrevem-se as funções geradoras dos dados aleatórios em ambiente GAMS, para o exemplo *spbatch\_ex1*. Apresentam-se, de forma complementar, os dados referentes à primeira instância amostral de EX1A.

```
*
* SPfmsz3_vdata_ex1MVP.inc
* modulo para definicao dos parametros
OPTION SEED = 08012007
H = UNIFORM (6000, 8000);
alpha(j)= UNIFORM (300, 700);
beta(j) = UNIFORM (0.5, 0.7);
LOOP (j,
    dv(j,"1") = UNIFORM(800,1300);
    dv(j,"2") = UNIFORM(1500,2000);
    dv(j,"3") = UNIFORM(2200,2700);
    dv(j,"4") = UNIFORM(2800,3300);
    dv(j,"5") = UNIFORM(3500,4000);
);
SF(i,j) = UNIFORM (1,5);
tau(i,j) = UNIFORM (2,9);
Q_med = 150000; Q_dsv = Q_med / 6.;
ret_med = 0.3; ret_dsv = ret_med / 6.;
Q(i,t,r) = NORMAL (Q_med, Q_dsv);
ret(i,t,r) = NORMAL (ret_med, ret_dsv);
txint = 0.09 ;
*****
```

```

*
* SPfmsz_dados_ex1A.txt
  4  3  5  3  ni, nj, ns, np
  1  1  nt, nr
1.0000  prob(r)
6061.11  H
817.75  ncupp
1092.48  1632.34  2564.37  3194.99  3571.68  dv(j,s)
1272.02  1805.56  2603.39  3029.56  3769.17  dv(j,s)
1086.97  1601.54  2391.81  2963.96  3965.08  dv(j,s)
  1.03  4.70  2.50  SF(i,j)
  4.85  1.45  1.34  SF(i,j)
  1.79  1.59  4.36  SF(i,j)
  1.19  3.41  3.82  SF(i,j)
  3.32  2.48  4.41  tau(i,j)
  8.29  8.38  6.48  tau(i,j)
  5.09  3.83  7.11  tau(i,j)
  6.57  8.27  2.55  tau(i,j)
68342.42  136684.85  205027.27  c(j,s,p)
89765.79  179531.58  269297.36  c(j,s,p)
121988.15  243976.30  365964.44  c(j,s,p)
141630.92  283261.85  424892.77  c(j,s,p)
152765.65  305531.31  458296.96  c(j,s,p)
27405.83  54811.66  82217.49  c(j,s,p)
33309.58  66619.16  99928.73  c(j,s,p)
40840.11  81680.22  122520.33  c(j,s,p)
44438.35  88876.70  133315.05  c(j,s,p)
50187.58  100375.15  150562.73  c(j,s,p)
28847.83  57695.66  86543.48  c(j,s,p)
36880.34  73760.69  110641.03  c(j,s,p)
47555.07  95110.14  142665.20  c(j,s,p)
54479.38  108958.76  163438.14  c(j,s,p)
65513.64  131027.28  196540.92  c(j,s,p)
181176.93  Q(i,t,r)
183270.76  Q(i,t,r)
153835.98  Q(i,t,r)
161498.55  Q(i,t,r)
  0.32  ret(i,t,r)
  0.30  ret(i,t,r)
  0.32  ret(i,t,r)
  0.37  ret(i,t,r)
*****

```

## 2. Estimadores económicos para instâncias do exemplo sbatch\_ex2

Apresentam-se os resultados comparáveis dos estimadores económicos para as instâncias EX2A, com 5 períodos de tempo, e EX2B, com 10 períodos de tempo,

para os seguintes parâmetros de penalização:  $\lambda_{dsv} = 1.0$ ;  $\lambda_{flg} = 0.1$ ;  $\lambda_{qns}$  de 0. a 20. em “passo” de 2.;

## EX2A

* Ord(s);	Vtotal;	Ctotal;	%Eproc;	%Efolg;	%Benef.;	Payback;	TIR ***
4 , 3 , 3,	7905.19	145840.19	7.6	7.7	120.7	4.0	16.5
4 , 3 , 4,	8353.83	157249.94	2.1	6.2	118.4	4.0	15.5
4 , 3 , 4,	8353.83	157249.94	2.1	6.2	118.4	4.0	15.5
5 , 4 , 4,	9807.69	164863.35	0.4	8.6	114.7	4.2	14.5
5 , 4 , 4,	9807.69	164863.35	0.4	8.6	114.7	4.2	14.5
5 , 4 , 4,	9807.69	164863.35	0.4	8.6	114.7	4.2	14.5
5 , 5 , 4,	10363.31	167983.17	0.1	9.2	112.9	4.3	13.5
5 , 5 , 4,	10363.31	167983.17	0.1	9.2	112.9	4.3	13.5
5 , 5 , 4,	10363.31	167983.17	0.1	9.2	112.9	4.3	13.5
5 , 5 , 4,	10363.31	167983.17	0.1	9.2	112.9	4.3	13.5
5 , 5 , 4,	10363.31	167983.17	0.1	9.2	112.9	4.3	13.5

## EX2B

* Ord(s);	Vtotal;	Ctotal;	%Eproc;	%Efolg;	%Benef.;	Payback;	TIR ***
4 , 3 , 4,	8353.83	157249.94	2.5	6.2	204.9	3.8	29.5
4 , 3 , 4,	8353.83	157249.94	2.4	6.2	204.7	3.8	29.5
5 , 4 , 4,	9807.69	164863.35	0.5	8.6	198.8	4.0	28.5
5 , 4 , 4,	9807.69	164863.35	0.5	8.6	198.8	4.0	28.5
5 , 4 , 4,	9807.69	164863.35	0.5	8.6	198.8	4.0	28.5
5 , 5 , 4,	10363.31	167983.17	0.1	9.2	195.7	4.1	27.5
5 , 5 , 4,	10363.31	167983.17	0.1	9.2	195.7	4.1	27.5
5 , 5 , 4,	10363.31	167983.17	0.1	9.2	195.7	4.1	27.5
5 , 5 , 4,	10363.31	167983.17	0.1	9.2	195.7	4.1	27.5
5 , 5 , 4,	10363.31	167983.17	0.1	9.2	195.7	4.1	27.5
5 , 5 , 4,	10363.31	167983.17	0.1	9.2	195.7	4.1	27.5





## Introdução ao modelo ELS

No problema de dimensionamento óptimo de lotes (*economic lot size*, ELS), procura determinar-se as quantidades a produzir em cada período, para satisfazer a procura ao longo de um determinado horizonte de planeamento, visando a minimização global de custos, considerando custos de produção, de posse e de pronto.

É bastante conhecida a abordagem por programação dinâmica de Wagner e Whitin [1958], interessa aqui analisar a resolução da correspondente formulação MILP, cujo tratamento pode ser acelerado através de desagregação de variáveis [Krarup e Bilde, 1977], ou de reformulação adequada [Martin, 1987], ambas conducentes à solução óptima no primeiro nó do *branch-and-bound*, ou seja, à exactidão da relaxação linear. Através da abordagem via dual —Erlenkotter [1978], Krarup e Bilde [1977]— ou por formulação em rede, visando o percurso de rota mínima [Eppen e Martin, 1987], são também conseguidos desempenhos de grande eficiência.

De facto, evidencia-se a eficiência de resolução do problema de ELS, decorrente das propriedades da estrutura matemática do problema e caracterizada por grande rapidez de cálculo, mas esta qualidade revela-se insusceptível de uma generalização directa. Embora este problema permaneça atractivo para se constituir como elemento ou subestrutura de problemas mais complexos — de *location*, *capacity expansion*, *batch scheduling*, *long range planning* —, aquelas características de eficiência apresentam dificuldades de extrapolação para situações aplicadas (multiproduto, limitação de capacidade, variação e estrutura dos custos), pelo que se justifica a contínua investigação a que se encontra submetido. Wolsey [1995], Kuik [1994], apresentam um guia dos progressos efectuados no âmbito do problema de ELS.

Neste conhecido problema, pretende minimizar-se o custo total associado à satisfação da procura ao longo do horizonte temporal, considerando custos fixos —de pronto— e custos variáveis, dependendo dos níveis de produção e posse. Os dados do problema, necessariamente conhecidos para cada um dos  $T$  períodos de tempo ( $t=1..T$ ), são:

- $p_t$  – custo (marginal) unitário de produção;
- $h_t$  – custo (marginal) unitário de posse (*hold*);
- $f_t$  – custo fixo de pronto (“setup”) de cada unidade de produção;
- $d_t$  – quantidade de produto procurada (*demand*).

Os referidos custos correspondem, respectivamente, às seguintes variáveis:

- $x_t$  – quantidade produzida;
- $s_t$  – quantidade de produto em armazém (*stock*);
- $y_t$  – decisão binária quanto à produção (0, não; 1, sim).

Nomeadamente, através de MILP, este problema pode formular-se no modelo que designamos por *els\_milp* onde, sequencialmente:

- a função objectivo exprime o total dos custos, a minimizar;
- a procura do produto deve ser satisfeita através das quantidades produzidas ou oportunamente armazenadas;
- a desigualdade estabelece o limite superior de produção, que será nulo quando a variável binária se anular;
- o somatório de procuras define o valor para esse limite superior;
- é suposto que as quantidades armazenadas, no início e fim do horizonte de planeamento, sejam nulas.

**Modelo *els\_milp*:**

$$1A.1-a \quad [min] \quad z = \sum_{t=1}^T (p_t x_t + h_t s_t + f_t y_t)$$

sujeito a,

$$s_{t-1} + x_t - s_t = d_t, \quad \forall t$$

$$1A.1-b \quad x_t - W_t y_t \leq 0, \quad \forall t$$

$$1A.1-c \quad W_t = \sum_{\tau=t}^T d_\tau, \quad \forall t$$

$$1A.1-d \quad s_0 = s_T = 0$$

$$1A.1-e, f \quad x_t, s_t \geq 0 \quad y_t \in \{0,1\}, \quad \forall t$$

Desta forma, o modelo *els\_milp* apresenta  $(2T)$  restrições (em cada período: balanço e limite superior de produção) e  $(3T)$  variáveis (em cada período: quantidade produzida, quantidade armazenada, variável de decisão binária). Note-se que  $W_t$  é um parâmetro definido directamente pelo total das procuras nos períodos  $\tau$  subsequentes, não constituindo pois uma variável a otimizar.

Algumas generalizações intrínsecas ao modelo apresentado, além do enquadramento multiproduto, da imposição activa de diferentes constrangimentos, associam-se com:

- maximização de lucros como objectivo último, em lugar de minimização dos custos;

- 
- ocorrência de custos, de produção ou posse, não proporcionais às quantidades, pelo que se configura a não-linearidade da função objectivo;
  - procuras (ou vendas) como função do preço considerado.

A generalização do modelo de ELS aos problemas tratados na Engenharia Química apresenta como fundamento inicial a adequação deste modelo quanto aos processos de produção descontínuos. No entanto, através de algoritmos de decomposição ou de reformulações, que permitam desenvolver as propriedades que o caracterizam (por um lado, a unimodularidade da matriz das restrições; por outro lado, constância da relação hierárquica entre custos), pretende-se também aprofundar a aplicabilidade do problema de dimensionamento óptimo de lotes a outros problemas. Particularmente, releva-se a aplicação deste modelo em termos da problemática resultante da coordenação entre as funções de produção, de distribuição e inventário (*production-distribution networks*), na problemática de localização de unidades (*location*) ou na de expansão de capacidades (*capacity expansion*). Eventualmente, as igualdades nos balanços materiais e a imposição dos limites superiores à produção configuram uma dupla de relações matemáticas que é facilmente reconhecida.





# Complexidade computacional

A teoria da complexidade computacional proporciona uma visão da dificuldade de um certo tipo de problema, classificando-o face a algoritmos de tempo polinomial existentes para a sua resolução [Garey *et al.*, 1979; Papadimitriou *et al.*, 1982].

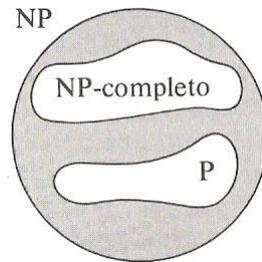
Um problema é definido como uma questão geral a ser respondida, possuindo parâmetros e visando especificar os valores das suas variáveis livres. Uma descrição do problema inclui uma lista geral de todos os seus parâmetros e uma definição das propriedades e constrangimentos que a solução deve satisfazer. Uma instância do problema consiste numa especificação de valores particulares para todos os seus parâmetros. Um problema de decisão é um problema cuja solução consiste numa resposta “sim” ou “não”. Um problema de optimização tem o objectivo de determinar o valor máximo ou mínimo de uma função objectivo específica, sujeita a restrições.

Um algoritmo é um procedimento para resolver um problema. Um algoritmo é de tempo polinomial se o seu tempo de resolução é limitado superiormente por uma função polinomial da dimensão do problema e da dos seus dados. Um algoritmo de tempo exponencial é um algoritmo cujo tempo de execução não é limitado por uma função polinomial da dimensão do problema. Um algoritmo de tempo pseudo-polinomial consiste num algoritmo com tempo de execução polinomialmente limitado, mas de dimensão de dados de maior grandeza. Um algoritmo não-determinístico consiste em duas fases: uma fase em que é produzida uma possível solução; e uma fase, de validação determinística, em que se verifica se a solução produzida é válida. Um algoritmo não-determinístico opera em tempo polinomial se, para uma dada escolha da solução a uma instância do problema, a validade dessa solução pode ser verificada em tempo polinomial.

A teoria da complexidade clássica classifica os problemas de decisão, isto é, problemas com respostas “sim” ou “não”, em duas categorias: aqueles para os quais existem algoritmos de tempo polinomial, que pertencem à classe de complexidade P; outros, para os quais existem algoritmos de tempo polinomial não-determinísticos, que pertencem à classe NP. No referente a este último caso, a classe NP consiste no grupo de problemas de decisão cuja solução pode ser verificada em tempo polinomial para determinar se é ou não válida.

Observa-se (*Figura 4A.1*) que  $P \subseteq NP$  e supõe-se que  $P \neq NP$ , embora ainda não o esteja provado. De facto, a classe P é um subconjunto da classe NP: todo o problema de decisão resolúvel por um algoritmo determinístico de tempo polinomial é resolúvel por um algoritmo não-determinístico de tempo polinomial,

bastando para tal ignorar a primeira fase de escolha da solução e utilizar o algoritmo determinístico de tempo polinomial na segunda fase de verificação. Quanto à questão  $P = NP$ , não se conhece qualquer método para converter algoritmos não-determinísticos de tempo polinomial nos algoritmos determinísticos de tempo polinomial.



**Figura 4A.1** – Classes NP e P.

Um problema  $Q$  em NP diz-se NP-completo caso um qualquer problema em NP pode ser transformado em tempo polinomial no problema  $Q$ . Se um único problema NP-completo pode ser resolvido em tempo polinomial, então todos os problemas NP-completos são resolúveis em tempo polinomial. Do mesmo modo, se qualquer problema do conjunto NP-completo for computacionalmente intratável, então todos os problemas NP-completos serão também intratáveis. Assim, um problema NP-completo apresenta a propriedade  $Q \in P$  se e apenas se  $P = NP$ .

Um algoritmo em tempo pseudo-polinomial apresenta um tempo de execução polinomialmente limitado na dimensão dos dados e na dimensão do maior valor dos dados. De facto, um algoritmo em tempo polinomial tem um tempo de execução polinomialmente limitado na dimensão dos dados e no logaritmo do maior valor dos dados. Problemas NP-completos para os quais não existe qualquer algoritmo de tempo pseudo polinomial dizem-se problemas NP-completos em sentido forte.

Para provar que um problema particular de decisão é NP-completo, terá de provar-se que o problema  $Q \in NP$  e que qualquer problema reconhecido como NP-completo é transformável polinomialmente nesse próprio problema. Para provar que um problema  $Q$  é NP-completo em sentido forte, não só terá de provar-se que  $Q \in NP$  e é viável a redução polinomial a um problema reconhecido como NP-completo, mas também uma redução polinomial a um problema NP-completo em sentido forte. Note-se que a dimensão dos valores dos dados deve manter-se polinomialmente limitada aquando da redução.

Uma redução (transformação) polinomial do problema  $Q$  no problema  $R$  existe se, para qualquer conjunto de dados de  $Q$ , o conjunto de dados de  $R$  pode ser construído em tempo polinomial, de forma que, uma vez resolvido o problema  $R$ , possa ser obtida a solução para o problema  $Q$  em tempo polinomial. Por consequência, uma redução do problema  $Q$  no problema  $R$  implica que  $R$  pode ser considerado um caso especial de  $Q$ , de forma que  $R$  é pelo menos tão difícil como  $Q$ .

Em regra, existem três técnicas comuns de provar que um problema é NP-completo: restrição, substituição local e concepção de componentes. Frequentemente aplicada é a demonstração por restrição: consiste em mostrar que o problema  $Q$  em análise contém algum problema reconhecido como NP-completo como um seu caso específico, colocando-se restrições às instâncias particulares de  $Q$ .

A complexidade computacional de um problema é caracterizada pela complexidade do algoritmo mais eficiente que o resolve. Note-se que a análise de complexidade computacional apenas considera o pior caso possível do tempo de resolução do melhor possível algoritmo ao longo de todas as possíveis instâncias do problema. Ocorrem casos onde o tempo médio de execução de um algoritmo se apresenta polinomial, embora a limitação de pior caso possível possa ser exponencial: na situação específica da LP, já foram encontrados algoritmos de tempo polinomial, mas o algoritmo do *simplex* funciona melhor em média, não obstante ser um algoritmo de tempo exponencial no pior caso possível.

Ora, o problema de otimização  $Q$  da forma  $\max\{cx|x \in X\}$  encontra-se associado a um problema de decisão: o problema de existência de *lower bound feasibility* (admissibilidade), no qual se pretende determinar quando  $\{x \in X, cx \geq z_0\} \neq \emptyset$ , onde  $z_0$  é um escalar. Por conseguinte, se o problema de *feasibility* é NP-completo, então um algoritmo de tempo polinomial para o problema de otimização  $Q$  iria implicar que  $P = NP$ . No entanto, como o problema de otimização  $Q$  não se encontra necessariamente em NP, não é NP-completo. Designa-se esse problema de otimização  $Q$  como NP-hard. Ou seja, um problema é NP-hard caso exista um problema NP-completo que possa ser polinomialmente reduzido àquele, sendo essa redução em tempo polinomial. Donde, um problema NP-hard é pelo menos tão difícil como um problema NP-completo, e um algoritmo polinomial para um problema NP-hard implica que  $P = NP$ .

Dado que a questão de saber se um problema  $Q$  se encontra em NP é irrelevante quando se pretende mostrar que ele é NP-hard, deve ser realizada uma descrição da complexidade computacional para os problemas de otimização. Na realidade, a complexidade de um problema de otimização geral não é imediatamente decorrente de uma inspeção. Por exemplo, parece óbvio que todos os problemas de MILP sejam NP-hard, mas ocorrem casos, como o do ELS reformulado, em que a resolução do LP-rel apresenta valores binários (0, 1) para as correspondentes variáveis de decisão binárias. Consequentemente, o problema ELS reformulado pode ser resolvido em tempo polinomial utilizando técnicas de LP. Atente-se que a formulação MILP de um problema de otimização é meramente uma representação matemática do problema, considerando-se que a complexidade computacional trata do pior caso possível do melhor tratamento matemático do problema — especificamente, assumindo que um problema  $Q \in NP$ , aquela implica que não existe nenhuma representação polinomialmente limitada de LP para o problema  $Q$ .

A abordagem geral para comprovar que um problema particular de otimização  $Q$  é NP-hard consiste em identificar problemas reconhecidamente NP-hard como casos especiais do problema  $Q$ . Consiste também em identificar um problema reconhecido como NP-completo que seja passível de redução polinomial para o problema de *feasibility*, o qual consistirá num caso especial do problema  $Q$ .

Um elevado número de problemas NP-completos são de importância prática significativa para serem abandonados apenas porque obter uma solução ótima é um processo intratável. Mas existem algumas abordagens possíveis: se a dimensão dos dados do problema é diminuta, então mesmo um algoritmo com tempo de execução exponencial poderá ser satisfatório; pode ser viável especificar casos particulares importantes, os quais sejam resolúveis em tempo polinomial; podem ser obtíveis soluções subótimas em tempo polinomial, através de algoritmos de aproximação.





# Heurísticas

Dado o interesse suscitado por um significativo número de problemas NP-completos, ou NP-hard, entre os quais se encontram importantes problemas de Engenharia Química formulados com variáveis inteiras ou binárias, pode recorrer-se à obtenção de soluções admissíveis e próximas do óptimo através de algoritmos de aproximação que, nomeadamente, sejam executados em tempo polinomial.

A qualidade do algoritmo de aproximação para um problema pode ser representada através da razão de aproximação, a qual relaciona o valor da solução obtida através da aproximação com o valor da respectiva solução óptima, para qualquer instância finita que se considere para o problema. Assim, um algoritmo com razão de aproximação 1 produz uma solução óptima, enquanto que um algoritmo que apresente um valor elevado para a razão produzirá uma solução de fraca qualidade.

Atente-se no facto de, para alguns problemas, se poder desenvolver algoritmos de aproximação em tempo polinomial cuja razão é constante e diminuta, embora para muitos outros problemas o que ocorre é que a razão de aproximação aumenta em função da dimensão dos dados. Alguns problemas NP-completos permitem algoritmos de aproximação em tempo polinomial, atingindo menores valores para a razão de aproximação a expensas de um maior tempo de computação: ocorre um natural balanceamento entre o tempo de execução e a qualidade da aproximação.

Uma heurística corresponde a um procedimento de aproximação: visa a obtenção de soluções de qualidade satisfatória, num intervalo de tempo também aceitável. Porém, é conveniente ter presente que o objectivo da heurística aproximativa consiste em substituir a solução exacta do problema por uma solução de recurso. O procedimento de aproximação trata-se de um conceito da área da algoritmia, enquanto que a heurística, método ou procedimento heurístico, apresenta uma abrangência lata, sendo aplicada nas ciências matemáticas e de engenharia.

Relativamente aos métodos exactos, os métodos heurísticos permitem a abordagem a problemas de grandes dimensões, não tratáveis pelos primeiros, se os problemas forem NP-completos; ainda, os métodos heurísticos permitem uma maior flexibilidade de formulação, recorrem a conceitos simples e empíricos, bem como possibilitam a utilização directa de informação sobre o problema. No entanto, os métodos heurísticos não garantem a optimalidade da solução obtida, pelo que interessa quantificar ou majorar o desvio inerente.

Genericamente, podem distinguir-se as heurísticas quanto ao objectivo do seu desenvolvimento: heurísticas de construção, quando se pretende obter uma solução admissível, adicionando sucessivamente novos elementos à solução inicial

(inadmissível); heurísticas de melhoria, quando se deseja melhorar uma solução admissível, já obtida entretanto, por exemplo, através de um método de construção.

Nas heurísticas de construção podemos considerar, principalmente: a heurística *greedy*, onde se selecciona a melhor opção em cada iteração, não se levando em consideração as iterações subseqüentes; a construção aleatória, sendo que a solução é definida ao acaso; a construção por simplificação, quando se pretende separar o problema em diversos módulos ou subproblemas, para os quais haja procedimentos já desenvolvidos, ou que permitam a generalização de casos especiais. No entanto, estas heurísticas de construção podem não produzir uma solução admissível, pelo que se revela benéfico considerar a integração de aspectos das diferentes heurísticas.

A pesquisa ou busca local (*Local Search*, LS) constitui uma ferramenta básica que, a partir de uma solução admissível, avalia a vizinhança (constituída por soluções admissíveis semelhantes àquela), seleccionando a melhor solução vizinha de acordo uma função de avaliação especificada. Baseia-se no princípio de que soluções próximas da solução ótima apresentarão estruturas também próximas ou semelhantes. Neste método, além de se definir a estrutura da vizinhança a pesquisar e a função de avaliação das soluções alternativas, é também necessário seleccionar a estratégia para alteração da solução, bem como especificar o critério de paragem. Naturalmente, estes aspectos do método permitem a introdução de variantes, quer na definição da vizinhança, quer na avaliação das alternativas ou na selecção das alterações a introduzir.

Com a iteração repetida do método de busca local, encontra-se um método de busca local iterada (*Iterated Local Search*, ILS), onde se combina a geração rápida de boas soluções com a convergência assintótica para o ótimo.

Quando a busca local permite continuar a procura utilizando soluções de pior qualidade, podemos classificá-la como busca Tabu (*Tabu Search*, TS). Este método possibilita a geração rápida de boas soluções, de melhor qualidade (em média) do que os métodos anteriores, pois pode evitar mínimos locais, mas é necessário armazenar a lista de soluções já encontradas (*tabu list*). Esta lista, com uma dimensão predefinida e gerida por uma regra arbitrária (por ex., *first in first out*), regista as alterações que se vão seleccionando, pelo que se evita o reciclo de soluções. Este método pode ser combinado com um procedimento de re-arranque quando não se verificar avanço na qualidade da solução após um dado número de iterações, terminando-se o método quando se observar o critério de paragem. Ainda, esse re-arranque pode também ser burilado através de um procedimento aleatório. Cavin *et al.* [2004] utilizam busca Tabu para a optimização combinatória e multicritério de um problema não linear, visando o dimensionamento de processos descontínuos numa instalação multiobjectivo (*multipurpose*). Investigam-se os parâmetros do algoritmo e da implementação computacional efectuada, supondo uma gama ampla de instâncias do caso de dimensionamento considerado.

O método de “arrefecimento” (“cristalização”) simulado(a) (*Simulated Annealing*, SA) vai gerar soluções na vizinhança da solução corrente, cruzando o espaço de procura na busca do ótimo global. Uma solução vizinha de qualidade superior é aceite, enquanto uma de qualidade inferior é avaliada de forma probabilística, em termos de um parâmetro de “temperatura”. A parametrização respeitante pode ser problemática dependendo, por exemplo, da quantidade de informação necessária, variando tal parametrização com o avanço do processo de busca. Athier *et al.* [1996] implementam um método de SA, conjugado a um nível

inferior com NLP para otimizar as variáveis operacionais, de forma a sistematizar a geração de redes de permutadores de calor (*heat exchanger network*), em busca do custo mínimo global. Os testes numéricos revelaram elevadas taxas de convergência, acompanhadas de tempos de resolução diminutos.

A busca por *ant colony* usualmente produz resultados de qualidade inferior às outras técnicas de busca local, mas é aplicável a problemas de otimização combinatória que não mostram ser passíveis de uma otimização global por outros métodos. Este método utiliza agentes para cruzar o espaço de procura e determinar ótimos locais, a sua implementação é relativamente simples, produzindo soluções de boa qualidade de forma eficiente. Jayaraman *et al.* [2000] aplicam um enquadramento de *ant colony* para o dimensionamento de processos químicos descontínuos, obtendo resultados com 100% de optimalidade para problemas de pequena dimensão.

Na heurística designada por Algoritmo Genético (*Genetic Algorithm*, GA), trata-se um conjunto dado de soluções como se fosse uma população de diferentes indivíduos, onde se verifica a actuação dos seguintes de aspectos do contexto genético para produzir um novo conjunto de soluções (população): selecção, onde uma solução permanece na população com uma probabilidade decorrente da sua qualidade; recombinação, quando duas soluções são combinadas para dar origem a uma ou duas novas soluções; mutação, quando ocorre uma alteração aleatória e mínima num elemento duma dada solução. Neste enquadramento, é crucial o método de representação de cada solução, tal como acontece na busca Tabu, especialmente, no registo da lista de alterações efectuadas. Bernal-Haro *et al.* [1998] desenvolvem e implementam um algoritmo genético para o dimensionamento de uma instalação química multiobjectivo (*multipurpose, jobshop*), para minimização dos custos do equipamento direccionado para um definido nível de produção. Obtiveram-se, de forma rápida e eficiente, soluções próximas do óptimo, as quais também revelaram boa qualidade no ponto de vista da aplicação de engenharia. Lohl *et al.* [1998] aplicam um algoritmo genético ao sequenciamento de operações *batch*, obtendo soluções de boa qualidade e bom desempenho numérico, quando se comparou com os resultantes, soluções e desempenho, de MP.

De uma forma geral, na resolução de um problema através de heurísticas deve ser usualmente especificado: o critério de terminação (usualmente, o máximo de tempo e/ou iterações); o método de construção (por exemplo, *greedy* ou aleatório); o tipo de vizinhança; o método ou tipo de heurística a utilizar (sendo o mais simples, a busca local).

Especialmente para problemas lineares com variáveis inteiras (IP, MILP), são utilizados também diversos procedimentos heurísticos, como os que se descrevem [Wolsey, 1998] em seguida (supondo que todas as variáveis inteiras são binárias):

- *dive-and-fix* — selecciona-se a solução de um LP obtido num dado nó da árvore de *branch-and-bound*; selecciona-se a variável de interesse cujo valor fraccionário esteja mais próximo de um valor binário (0 ou 1), fixando-se nesse valor binário; resolve-se este novo LP, terminando-se o método em caso de inadmissibilidade; perante admissibilidade, repete-se o método, até não ocorrerem quaisquer variáveis com valor fraccionário;
- *relax-and-fix* — repartem-se as variáveis binárias em dois grupos, segundo a sua importância; relaxando-se as restrições de integralidade para as variáveis binárias menos importante, resolve-se um problema

---

MILP1 mais fácil; então, num segundo problema, MILP2, fixam-se as variáveis mais importantes nos valores óptimos obtidos do primeiro problema (MILP1), obtendo-se assim valores inteiros para as variáveis menos importantes (para MILP2 admissível);

- *cut-and-fix* — procedimento similar ao anterior, mas a fixação de valores para as variáveis binárias de interesse, resultam da aplicação de planos de corte (*strong cutting planes*).

Exemplificando, Kelly e Mann [2004] utilizam o método de *relax-and-fix* num procedimento heurístico para decomposição de fluxogramas (*flowsheet decomposition heuristic*) dirigidas ao sequenciamento de grupos de processos industriais. O método consiste em resolver sequencialmente uma série de problemas MILP (um problema para cada grupo de processos equacionado), relaxando-se uma série de variáveis binárias que não correspondem ao actual grupo, fixando-se o valor para as variáveis binárias desse grupo. Reportam-se exemplos numéricos que mostram a efectividade deste método de *relax-and-fix*.

Também é de interesse a possibilidade de hibridação entre diferentes heurísticas, sendo que a vantagem na utilização dessas regras heurísticas reside na sua adequação à especificidade de cada problema. Nomeadamente, um aspecto a explorar é a integração de heurísticas com abordagens exactas: a partir de uma solução admissível obtida por heurística, podem ser eliminados numerosos ramos numa busca enumerativa por *branch-and-bound*; iniciando-se com uma solução parcial através de *branch-and-bound*, pode prosseguir-se com uma busca local. Assim, é promissora a possibilidade de desenvolvimento de um motor de cálculo baseado em heurísticas, o qual poderia ser acedido a partir de um sistema ou ambiente de modelação por MP.

