F. N. Correia L. V. Tavares I. H. Themido Rui. C. Oliveira

Minister of Environment (2007)

Luís Valadares Tavares, professor catedrático do IST, doutor em Ciências de Engenharia (UTL). Sócio fundador da Associação Portuguesa de Investigação Operacional (APDIO). Vice-presidente da *International Federation of Operational Research Societies* (88-89) e presidente do grupo Europeu de *Project Management and Scheduling* (EURO). Dedica-se especialmente às áreas de Gestão de Projectos, Teoria de Decisão e da Negociação, Desenvolvimento e avaliação de sistemas públicos.

Isabel Hall Themido, doutorada em Investigação Operacional (Universidade de Lancaster, R.U.) é professora associada do IST, presidente da Associação Portuguesa de Investigação Operacional (APDIO), tendo sido vice-presidente da Federação Europeia de Associações de Investigação Operacional - EURO (1988-92). Tem orientado a sua actividade profissional no sentido de conciliar os interesses de investigação com os problemas reais vividos pelas organizações, nos domínios da Logística, *Marketing* Quantitativo e modelação estatística multivariável.

Rui Carvalho Oliveira é doutorado em Engenharia de Sistemas pela UTL, sendo actualmente professor associado do IST. Tem desenvolvido investigação sobre metodologias e técnicas de I.O. (nomeadamente de simulação, optimização combinatória e sistemas de apoio à decisão) aplicadas a sistemas produtivos, logísticos e ferroviários, áreas em que tem também realizado projecto e consultoria para diversas empresas públicas e privadas.

Francisco Nunes Correia é professor associado do IST e doutor em Recursos Hídricos (Universidade de Fort Collens, EUA). É director de projectos europeus sobre desenvolvimento e gestão de sistemas hídricos e foi coordenador do Plano Nacional do Ambiente. Dedica-se à aplicação da Análise de Sistemas ao estudo de políticas ambientais e de valorização dos recursos naturais.

9 789728 298081
ISBN: 972-8298-08-0

14. Casos particulares da programação linear

14.1. Introdução

Os algoritmos correntes de programação linear e, em particular, o Simplex, permitem resolver problemas muito diversos que podem ser formulados de acordo com a forma canónica de um problema de PL. Existem, contudo, problemas que apresentam uma estrutura particular da qual pode ser tirado partido para desenvolver algoritmos mais simples e mais eficientes que se aplicam apenas a esse tipo de problemas. Esses algoritmos correspondem, em geral, a versões simplificadas ou expeditas do Simplex.

São exemplos de situações deste tipo os problemas de transporte, transexpedição e afectação que correspondem precisamente a problemas que podem ser formulados em termos de PL, mas para os quais foram desenvolvidos algoritmos resolutivos específicos que podem ser utilizados com grande vantagem.

Não existe a preocupação de formalizar matematicamente os algoritmos apresentados, nem de fazer a sua demonstração. Procura-se apenas relacionar cada tipo de problema com a formulação geral da PL, sistematizar os passos do respectivo algoritmo resolutivo e ilustrar as técnicas apresentadas com exemplos numéricos. Com efeito, de um ponto de vista conceptual, estes casos particulares não vêm aduzir contributos especialmente relevantes, antes correspondendo a versões simplificadas dos algoritmos básicos.

14.2. Transportes

O chamado "problema de transportes", embora nem sempre envolva transportes no sentido real, é um dos casos particulares do modelo geral de PL com utilização corrente. Comecemos por dar um exemplo de enunciado:

Em relação a determinado produto de grande necessidade existem numa dada região 2 fábricas, A e B, e 3 centros de recepção, 1, 2 e 3.

As necessidades diárias dos centros de recepção 1, 2 e 3 são respectivamente de 1 000, 1 500 e 2 500 unidades, sendo as disponibilidades das fábricas A e B, também por dia, de respectivamente 2 000 e 3 000 unidades. Os custos de transporte por unidade de cada fábrica, ou origem, para cada centro, ou destino, resumem-se no seguinte quadro:

	Centros de recepção		
Fábricas	1	2	3
A	2	1	0.5
B	1	0.5	2

Pretende-se saber qual a melhor forma de efectuar os transportes para, por um lado, conseguir satisfazer integralmente as necessidades e, por outro, assegurar que o custo total do transporte seja mínimo.

INVESTIGAÇAO OPERACIONAL

Luís Valadares Tavares
Rui Carvalho Oliveira
Isabel Hall Themido
Francisco Nunes Correia

McGRAW-HILL 1996

LISBOA • SÃO PAULO • BOGOTÁ • BUENOS AIRES • GUATEMALA
MADRID • MÉXICO • NOVA IORQUE • PANAMÁ • SAN JUAN • SANTIAGO
AUCKLAND • HAMBURGO • KUALA LAMPUR • LONDRES
MILÃO • MONTREAL • NOVA DELI • PARIS • SINGAPURA • SYDNEY
TÓQUIO • TORONTO

3 e $c'_{22} = 0.5$, correspondendo,

5

valor obtido quando não se impõe a restrição adicional de x_{13} ser nula.

14.3. Transexpedição

Nos problemas de transportes apresentados em 14.2 consideram-se apenas origens e destinos. É frequente, contudo, existirem não apenas origens e destinos mas também armazéns ou entrepostos onde os produtos são temporariamente armazenados ou, de forma mais geral, locais que não produzem nem consomem, mas por onde os produtos podem, eventualmente, passar.

Na formulação mais geral pode mesmo admitir-se que todos os pontos podem funcionar simultaneamente como origens, entrepostos e destinos, constituindo assim uma rede cujos fluxos podem ter origem, destino ou transitar por qualquer ramo entre dois pontos.

Assim, se considerarmos m origens, n destinos e p entrepostos, podemos formular um problema de transportes, tal como foi apresentado no capítulo anterior, em que se consideram $(m + p + n)$ origens e $(m + p + n)$ destinos. O número total de variáveis é, naturalmente, $(m + p + n) \cdot (m + p + n)$. O número de variáveis básicas é $2(m + p + n) - 1$. É, naturalmente, necessário conhecer a matriz de custos, que tem a dimensão $(m + p + n) \cdot (m + p + n)$.

Para simplificar a notação pode considerar-se que x_{ij} é a quantidade transportada de i para j, fazendo-se:

$$i = 1, 2, ..., m, m + 1, m + 2, ..., m + p, m + p + 1, m + p + 2, ..., m + p + n$$
$$j = 1, 2, ..., m, m + 1, m + 2, ..., m + p, m + p + 1, m + p + 2, ..., m + p + n$$

sendo m o número de origens; p, o número de entrepostos, e n, o número de destinos.

Cada local que funcione como entreposto tem de ser capaz de ceder aos outros locais uma quantidade de produtos que, no seu limite superior, só pode ser igual a Q, sendo Q dado por:

$$Q = \sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j$$

Assim, cada origem deverá ser capaz de ceder um valor inferior a $a_i + Q$, e cada destino deverá ser capaz de absorver um valor inferior a $b_j + Q$, admitindo que, no limite, todos os produtos transitavam por todas as origens e por todos os destinos.

Deve ainda ser considerado que cada entreposto e cada destino funcionando como origem devem ser capazes de ceder Q. Analogamente, cada origem e cada entreposto funcionando como destino devem ser capazes de absorver Q.

O problema de transexpedição pode, assim, ser formulado em termos de programação linear:

$$\text{Min } F = \sum_{i=1}^{m+p+n} \sum_{j=1}^{m+p+n} C_{ij}\, x_{ij}$$

Sujeito a:

$$\sum_{j=1}^{m+p+n} x_{ij} = a_i + Q \qquad (i = 1, 2, \ldots, m)$$

$$\sum_{i=1}^{m+p+n} x_{ij} = Q \qquad (i = m+1, m+2, \ldots, m+p+n)$$

$$\sum_{i=1}^{m+p+n} x_{ij} = Q \qquad (j = 1, 2, \ldots, m+p)$$

$$\sum_{i=1}^{m+p+n} x_{ij} = b_j + Q \qquad (j = m+p+1, \ldots, m+p+n)$$

$$x_{ij} \geq 0 \qquad (i, j = 1, 2, \ldots, m+p+n)$$

Esta formulação corresponde, afinal, a um problema de transportes e é, portanto, um caso particular de PL. A sua resolução baseia-se no algoritmo dos transportes que precisa ser "expandido" para acomodar todas as origens, destinos e entrepostos.

Para exemplificar a resolução de problemas de transexpedição considere-se o problema apresentado no 14.2 com a consideração adicional de um entreposto representado por E. Em vez de duas origens e três destinos temos agora um total de seis locais que podem desempenhar o papel de origens ou destinos.

A matriz de custos é dada por:

	A	B	E	1	2	3	Disponibilidades
A	0	2	2.5	2	1	0.5	2 + 5
B	2	0	0.5	1	0.5	2	3 + 5
E	2.5	0.5	0	2	2.5	0.5	5
1	2	1	2	0	2	2.5	5
2	1	0.5	2.5	2	0	2	5
3	0.5	2	0.5	2.5	2	0	5
Necessidades	5	5	5	1 + 5	1.5 + 5	2.5 + 5	35