~~**tabulation is helpful in clarifying thought and detecting incorrect mathematical descriptions.**~~

Other problems

The main areas in which IP is used in practice include:

- imposition of logical conditions in LP problems (such as the either/or condition dealt with above)
- blending with a limited number of ingredients
- depot location
- job shop scheduling
- assembly line balancing
- airline crew scheduling
- timetabling

Integer programming example

Recall the blending problem dealt with before under linear programming. To remind you of it we reproduce it below.

Blending problem **Problem 1**

Consider the example of a manufacturer of animal feed who is producing feed mix for dairy cattle. In our simple example the feed mix contains two active ingredients and a filler to provide bulk. One kg of feed mix must contain a minimum quantity of each of four nutrients as below:

Nutrient	A	B	C	D
gram	90	50	20	2

The ingredients have the following nutrient values and cost

	A	B	C	D	Cost/kg
Ingredient 1 (gram/kg)	100	80	40	10	40
Ingredient 2 (gram/kg)	200	150	20	-	60

What should be the amounts of active ingredients and filler in one kg of feed mix?

Blending problem solution

Variables

In order to solve this problem it is best to think in terms of one kilogram of feed mix. That kilogram is made up of three parts - ingredient 1, ingredient 2 and filler so: let
x_1 = amount (kg) of ingredient 1 in one kg of feed mix
x_2 = amount (kg) of ingredient 2 in one kg of feed mix
x_3 = amount (kg) of filler in one kg of feed mix
where $x_1 >= 0$, $x_2 >= 0$ and $x_3 >= 0$.

Constraints

- balancing constraint (an *implicit* constraint due to the definition of the variables)

$x_1 + x_2 + x_3 = 1$

- nutrient constraints

$100x_1 + 200x_2 >= 90$ (nutrient A)
$80x_1 + 150x_2 >= 50$ (nutrient B)
$40x_1 + 20x_2 >= 20$ (nutrient C)
$10x_1 >= 2$ (nutrient D)

Note the use of an inequality rather than an equality in these constraints, following the rule we put forward in the Two Mines example, where we assume that the nutrient levels we want are lower limits on the amount of nutrient in one kg of feed mix.

Objective

Presumably to minimise cost, i.e.

minimise $40x_1 + 60x_2$

which gives us our complete LP model for the blending problem.

Suppose now we have the additional conditions:

- if we use any of ingredient 2 we incur a fixed cost of 15
- we need not satisfy all four nutrient constraints but need only satisfy three of them (i.e. whereas before the optimal solution required all four nutrient constraints to be satisfied now the optimal solution could (if it is worthwhile to do so) only have three (any three) of these nutrient constraints satisfied and the fourth violated.

Give the complete MIP formulation of the problem with these two new conditions added.

Solution

To cope with the condition that if $x_2>=0$ we have a fixed cost of 15 incurred we have the standard trick of introducing a zero-one variable y defined by

```
y = 1 if x2>=0
  = 0 otherwise
```

and

- add a term +15y to the objective function

and add the additional constraint

- $x_2 <=$ [largest value x_2 can take]y

In this case it is easy to see that x_2 can never be greater than one and hence our additional constraint is $x_2<= y$.

To cope with condition that need only satisfy three of the four nutrient constraints we introduce four zero-one variables z_i (i=1,2,3,4) where

```
zi = 1 if nutrient constraint i (i=1,2,3,4) is satisfied
```

```
   = 0 otherwise
```

and add the constraint

- $z_1+z_2+z_3+z_4$ >= 3 (at least 3 constraints satisfied)

and alter the nutrient constraints to be

- $100x_1 + 200x_2 >= 90z_1$
- $80x_1 + 150x_2 >= 50z_2$
- $40x_1 + 20x_2 >= 20z_3$
- $10x_1 >= 2z_4$

The logic behind this change is that if a z_i=1 then the constraint becomes the original nutrient constraint which needs to be satisfied. However if a z_i=0 then the original nutrient constraint becomes

- same left-hand side >= zero

which (for the four left-hand sides dealt with above) is always true and so can be neglected - meaning the original nutrient constraint need not be satisfied. Hence the complete (MIP) formulation of the problem is given by

```
minimise 40x1 + 60x2 + 15y
subject to
x1 + x2 + x3 = 1
100x1 + 200x2 >= 90z1
80x1 + 150x2 >= 50z2
40x1 + 20x2 >= 20z3
10x1 >= 2z4
z1 + z2 + z3 + z4 >= 3
x2 <= y
zi = 0 or 1    i=1,2,3,4
y = 0 or 1
xi >= 0    i=1,2,3
```

Integer programming example **Problem 2**

In the planning of the monthly production for the next six months a company must, in each month, operate either a normal shift or an extended shift (if it produces at all). A normal shift costs £100,000 per month and can produce up to 5,000 units per month. An extended shift costs £180,000 per month and can produce up to 7,500 units per month. Note here that, for either type of shift, the cost incurred is fixed by a union guarantee agreement and so is independent of the amount produced.

It is estimated that changing from a normal shift in one month to an extended shift in the next month costs an extra £15,000. No extra cost is incurred in changing from an extended shift in one month to a normal shift in the next month.

The cost of holding stock is estimated to be £2 per unit per month (based on the stock held at the end of each month) and the initial stock is 3,000 units (produced by a normal shift). The amount in stock at the end of month 6 should be at least 2,000 units. The demand for the company's product in each of the next six months is estimated to be as shown below:

Month	1	2	3	4	5	6
Demand	6,000	6,500	7,500	7,000	6,000	6,000

Production constraints are such that if the company produces anything in a particular month it must produce at least 2,000 units. If the company wants a production plan for the next six months that avoids stockouts, formulate their problem as an integer program.

Hint: first formulate the problem allowing non-linear constraints and then attempt to make all the constraints linear.

Solution

Variables

The decisions that have to be made relate to:

- whether to operate a normal shift or an extended shift in each month; and
- how much to produce each month.

Hence let:

```
xt = 1 if we operate a normal shift in month t (t=1,2,...,6)
   = 0 otherwise

yt = 1 if we operate an extended shift in month t (t=1,2,...,6)
   = 0 otherwise

Pt (>= 0) be the amount produced in month t (t=1,2,...,6)
```

In fact, for this problem, we can ease the formulation by defining three additional variables - namely let:

```
zt = 1 if we switch from a normal shift in month t-1 to an extended
       shift in month t (t=1,2,...,6)
   = 0 otherwise

It be the closing inventory (amount of stock left) at the end
   of month t (t=1,2,...,6)

wt = 1 if we produce in month t, and hence from the production
       constraints Pt >= 2000 (t=1,2,...,6)
   = 0 otherwise (i.e. Pt = 0)
```

The motivation behind introducing the first two of these variables (z_t, I_t) is that in the objective function we will need terms relating to shift change cost and inventory holding cost. The motivation behind introducing the third of these variables (w_t) is the production constraint "either P_t = 0 or P_t >= 2000", which needs a zero-one variable so that it can be dealt with using the standard trick for "either/or" constraints.

In any event formulating an IP tends to be an iterative process and if we have made a mistake in defining variables we will encounter difficulties in formulating the constraints/objective. At that point we can redefine our variables and reformulate.

Constraints

We first express each constraint in words and then in terms of the variables defined above.

- only operate (at most) one shift each month

```
xt + yt <= 1    t=1,2,...,6
```

Note here that we could not have made do with just one variable (x_t say) and defined that variable to be one for a normal shift and zero for an extended shift (since in that case what if we decide not to produce in a particular month?).

Although we could have introduced a variable indicating no shift (normal or extended) operated in a particular month this is not necessary as such a variable is equivalent to $1-x_t-y_t$.

- production limits not exceeded

$P_t <= 5000x_t + 7500y_t \quad t=1,2,...,6$

Note here the use of addition in the right-hand side of the above equation where we are making use of the fact that at most one of x_t and y_t can be one and the other must be zero.

- no stockouts

$I_t >= 0 \quad t=1,2,...,6$

- we have an inventory continuity equation of the form

closing stock = opening stock + production - demand

where $I_0 = 3000$. Hence letting D_t = demand in month t (t=1,2,...,6) (a known constant) and assuming

- that opening stock in period t = closing stock in period t-1 and
- that production in period t is available to meet demand in period t

we have that

$I_t = I_{t-1} + P_t - D_t \quad t=1,2,...,6$

As noted above this equation assumes that we can meet demand in the current month from goods produced that month. Any time lag between goods being produced and becoming available to meet demand is easily incorporated into the above equation. For example for a 2 month time lag we replace P_t in the above equation by P_{t-2} and interpret I_t as the number of goods in stock at the end of month t which are available to meet demand i.e. goods are not regarded as being in stock until they are available to meet demand. Inventory continuity equations of the type shown are common in production planning problems.

- the amount in stock at the end of month 6 should be at least 2000 units

$I_6 >= 2000$

- production constraints of the form "either $P_t = 0$ or $P_t >= 2,000$".

Here we make use of the standard trick we presented for "either/or" constraints. We have already defined appropriate zero-one variables w_t (t=1,2,...,6) and so we merely need the constraints

$P_t <= Mw_t \quad t=1,2,...,6$

$P_t >= 2000w_t \quad t=1,2,...,6$

Here M is a positive constant and represents the most we can produce in any period t (t=1,2,...,6). A convenient value for M for this example is M = 7500 (the most we can produce irrespective of the shift operated).

- we also need to relate the shift change variable z_t to the shifts being operated

The obvious constraint is

$z_t = x_{t-1}y_t \quad t=1,2,...,6$

since as both x_{t-1} and y_t are zero-one variables z_t can only take the value one if both x_{t-1} and y_t are one (i.e. we operate a normal shift in period t-1 and an extended shift in period t). Looking back to the verbal description of z_t it is clear that the mathematical description given above is equivalent to that verbal description. (Note here that we define $x_0 = 1$ ($y_0 = 0$)).

This constraint is non-linear. However we are told that we can first formulate the problem with non-linear constraints and so we proceed. We shall see later how to linearise (generate equivalent linear constraints for) this equation.

Objective

We wish to minimise total cost and this is given by

SUM{t=1,...,6}($100000x_t + 180000y_t + 15000z_t + 2I_t$)

Hence our formulation is complete.

Note that, in practise, we would probably regard I_t and P_t as taking fractional values and round to get integer values (since they are both large this should be acceptable). Note too here that this is a non-linear integer program.

Comments

In practice a model of this kind would be used on a "rolling horizon" basis whereby every month or so it would be updated and resolved to give a new production plan.

The inventory continuity equation presented is quite flexible, being able to accommodate both time lags (as discussed previously) and wastage. For example if 2% of the stock is wasted each month due to deterioration/pilfering etc then the inventory continuity equation becomes $I_t = 0.98I_{t-1} + P_t - D_t$. Note that, if necessary, the objective function can include a term related to $0.02I_{t-1}$ to account for the loss in financial terms.

In order to linearise (generate equivalent linear constraints) for our non-linear constraint we again use a standard trick. Note that that equation is of the form

- A = BC

where A, B and C are zero-one variables. The standard trick is that a non-linear constraint of this type can be replaced by the two linear constraints

- A <= (B+C)/2 and
- A >= B+C-1

To see this we use the fact that as B and C take only zero-one values there are only four possible cases to consider:

B	C	A = BC becomes	A <= (B+C)/2 becomes	A >= B+C-1 becomes
0	0	A = 0	A <= 0	A >= -1
0	1	A = 0	A <= 0.5	A >= 0

```
1   0   A = 0      A <= 0.5        A >= 0
1   1   A = 1      A <= 1          A >= 1
```

Then, recalling that A can also only take zero-one values, it is clear that in each of the four possible cases the two linear constraints (A <= (B+C)/2 and A >= B+C-1) are equivalent to the single non-linear constraint (A=BC).

Returning now to our original non-linear constraint

- $z_t = x_{t-1}y_t$

this involves the three zero-one variables z_t, x_{t-1} and y_t and so we can use our general rule and replace this non-linear constraint by the two linear constraints

$$z_t \le (x_{t-1} + y_t)/2 \quad t=1,2,...,6$$
$$\text{and } z_t \ge x_{t-1} + y_t - 1 \quad t=1,2,...,6$$

Making this change transforms the non-linear integer program given before into an equivalent linear integer program.

Integer programming example 1996 MBA exam **Problem 3**

A toy manufacturer is planning to produce new toys. The setup cost of the production facilities and the unit profit for each toy are given below. :

Toy	Setup cost (£)	Profit (£)
1	45000	12
2	76000	16

The company has two factories that are capable of producing these toys. In order to avoid doubling the setup cost only *one* factory could be used.

The production rates of each toy are given below (in units/hour):

	Toy 1	Toy 2
Factory 1	52	38
Factory 2	42	23

Factories 1 and 2, respectively, have 480 and 720 hours of production time available for the production of these toys. The manufacturer wants to know *which* of the new toys to produce, *where* and *how many* of each (if any) should be produced so as to maximise the total profit.

- Introducing 0-1 decision variables *formulate* the above problem as an integer program. (Do *not* try to solve it).
- Explain briefly how the above mathematical model can be used in production planning.

Solution

Variables

We need to decide whether to setup a factory to produce a toy or not so let f_{ij} = 1 if factory i (i=1,2) is setup to produce toys of type j (j=1,2), 0 otherwise

We need to decide how many of each toy should be produced in each factory so let x_{ij} be the number of

toys of type j (j=1,2) produced in factory i (i=1,2) where $x_{ij} \ge 0$ and integer.

Constraints

- at each factory cannot exceed the production time available

$x_{11}/52 + x_{12}/38 \le 480$
$x_{21}/42 + x_{22}/23 \le 720$

- cannot produce a toy unless we are setup to do so

$x_{11} \le 52(480)f_{11}$
$x_{12} \le 38(480)f_{12}$
$x_{21} \le 42(720)f_{21}$
$x_{22} \le 23(720)f_{22}$

Objective

The objective is to maximise total profit, i.e.

maximise $12(x_{11}+ x_{21}) + 16(x_{12}+ x_{22}) - 45000(f_{11} + f_{21}) - 76000(f_{12} + f_{22})$

Note here that the question says that in order to avoid doubling the setup costs only one factory could be used. This is not a constraint. We can argue that if it is only cost considerations that prevent us using more than one factory these cost considerations have already been incorporated into the model given above and the model can decide for us how many factories to use, rather than we artificially imposing a limit via an explicit constraint on the number of factories that can be used.

The above mathematical model could be used in production planning in the following way:

- enables us to maximise profit, rather than relying on an ad-hoc judgemental approach
- can be used for sensitivity analysis - for example to see how sensitive our production planning decision is to changes in the production rates
- enables us to see the effect upon production of (say) increasing the profit per unit on toy 1
- enables us to easily replan production in the event of a change in the system (e.g. a reduction in the available production hours at factory one due to increased work from other products made at that factory)
- can use on a rolling horizon basis to plan production as time passes (in which case we perhaps need to introduce a time subscript into the above model)

Integer programming example 1995 MBA exam **Problem 4**

A project manager in a company is considering a portfolio of 10 large project investments. These investments differ in the estimated long-run profit (net present value) they will generate as well as in the amount of capital required.

Let P_j and C_j denote the estimated profit and capital required (both given in units of millions of £) for investment opportunity j (j=1,...,10) respectively. The total amount of capital available for these investments is Q (in units of millions of £)

Investment opportunities 3 and 4 are mutually exclusive and so are 5 and 6. Furthermore, neither 5 nor 6 can be undertaken unless either 3 or 4 is undertaken. At least two and at most four investment opportunities have to be undertaken from the set {1,2,7,8,9,10}.

The project manager wishes to select the combination of capital investments that will maximise the total estimated long-run profit subject to the restrictions described above.

Formulate this problem using an *integer programming* model and comment on the difficulties of solving this model. (Do not actually solve it).

What are the advantages and disadvantages of using this model for portfolio selection?

Solution

Variables

We need to decide whether to use an investment opportunity or not so let $x_j = 1$ if we use investment opportunity j (j=1,...,10), 0 otherwise

Constraints

- total amount of capital available for these investments is Q

SUM{j=1,...,10}$C_j x_j$ <= Q

- investment opportunities 3 and 4 are mutually exclusive and so are 5 and 6

$x_3 + x_4 <= 1$
$x_5 + x_6 <= 1$

- neither 5 nor 6 can be undertaken unless either 3 or 4 is undertaken

$x_5 <= x_3 + x_4$
$x_6 <= x_3 + x_4$

- at least two and at most four investment opportunities have to be undertaken from the set {1,2,7,8,9,10}

$x_1 + x_2 + x_7 + x_8 + x_9 + x_{10} >= 2$
$x_1 + x_2 + x_7 + x_8 + x_9 + x_{10} <= 4$

Objective

The objective is to maximise the total estimated long-run profit i.e.

maximise SUM{j=1,...,10}$P_j x_j$

The model given above is a very small zero-one integer programming problem with just 10 variables and 7 constraints and should be very easy to solve. For example even by complete (total) enumeration there are just 2^{10} = 1024 possible solutions to be examined.

The advantages and disadvantages of using this model for portfolio selection are:

- enables us to maximise profit, rather than relying on an ad-hoc judgemental approach
- can be easily extended to deal with a larger number of potential investment opportunities

- can be used for sensitivity analysis, for example to see how sensitive our portfolio selection decision is to changes in the data
- the model fails to take into account any statistical uncertainly (risk) in the data, it is a completely deterministic model, for example project j might have a (known or estimated) statistical distribution for its profit P_j and so we might need a model that takes this distribution into account

---

Integer programming example 1994 MBA exam

Problem 5

A food is manufactured by refining raw oils and blending them together. The raw oils come in two categories:

- Vegetable oil:
  - VEG1
  - VEG2
- Non-vegetable oil:
  - OIL1
  - OIL2
  - OIL3

The prices for buying each oil are given below (in £/tonne)

```
VEG1  VEG2  OIL1  OIL2  OIL3
115   128   132   109   114
```

The final product sells at £180 per tonne. Vegetable oils and non-vegetable oils require different production lines for refining. It is not possible to refine more than 210 tonnes of vegetable oils and more than 260 tonnes of non-vegetable oils. There is no loss of weight in the refining process and the cost of refining may be ignored.

There is a technical restriction relating to the hardness of the final product. In the units in which hardness is measured this must lie between 3.5 and 6.2. It is assumed that hardness blends linearly and that the hardness of the raw oils is:

```
VEG1  VEG2  OIL1  OIL2  OIL3
8.8   6.2   1.9   4.3   5.1
```

It is required to determine what to buy and how to blend the raw oils so that the company maximises its profit.

- *Formulate* the above problem as a *linear program*. (Do not actually solve it).
- What assumptions do you make in solving this problem by linear programming?

The following extra conditions are imposed on the food manufacture problem stated above as a result of the production process involved:

- the food may never be made up of more than 3 raw oils
- if an oil (vegetable or non-vegetable) is used, at least 30 tonnes of that oil must be used
- if either of VEG1 or VEG2 are used then OIL2 must also be used

Introducing 0-1 integer variables extend the linear programming model you have developed to encompass these new extra conditions.

Solution

Variables

We need to decide how much of each oil to use so let x_i be the number of tonnes of oil of type i used (i=1,...,5) where i=1 corresponds to VEG1, i=2 corresponds to VEG2, i=3 corresponds to OIL1, i=4 corresponds to OIL2 and i=5 corresponds to OIL3 and where $x_i >= 0$ i=1,...,5

Constraints

- cannot refine more than a certain amount of oil

$x_1 + x_2 <= 210$
$x_3 + x_4 + x_5 <= 260$

- hardness of the final product must lie between 3.5 and 6.2

$(8.8x_1 + 6.2x_2 + 1.9x_3 + 4.3x_4 + 5.1x_5)/(x_1 + x_2 + x_3 + x_4 + x_5) >= 3.5$
$(8.8x_1 + 6.2x_2 + 1.9x_3 + 4.3x_4 + 5.1x_5)/(x_1 + x_2 + x_3 + x_4 + x_5) <= 6.2$

Objective

The objective is to maximise total profit, i.e.

maximise $180(x_1 + x_2 + x_3 + x_4 + x_5) - 115x_1 - 128x_2 - 132x_3 - 109x_4 - 114x_5$

The assumptions we make in solving this problem by linear programming are:

- all data/numbers are accurate
- hardness does indeed blend linearly
- no loss of weight in refining
- can sell all we produce

Integer program

Variables

In order to deal with the extra conditions we need to decide whether to use an oil or not so let $y_i = 1$ if we use any of oil i (i=1,...,5), 0 otherwise

Constraints

- must relate the amount used (x variables) to the integer variables (y) that specify whether any is used or not

$x_1 <= 210y_1$
$x_2 <= 210y_2$
$x_3 <= 260y_3$
$x_4 <= 260y_4$
$x_5 <= 260y_5$

- the food may never be made up of more than 3 raw oils

$y_1 + y_2 + y_3 + y_4 + y_5 <= 3$

- if an oil (vegetable or non-vegetable) is used, at least 30 tonnes of that oil must be used

$x_i >= 30y_i$ i=1,...,5

- if either of VEG1 or VEG2 are used then OIL2 must also be used

$y_4 >= y_1$
$y_4 >= y_2$

Objective

The objective is unchanged by the addition of these extra constraints and variables.

Integer programming example 1985 UG exam

Problem 6

A factory works a 24 hour day, 7 day week in producing four products. Since only one product can be produced at a time the factory operates a system where, throughout one day, the same product is produced (and then the next day either the same product is produced or the factory produces a different product). The rate of production is:

Product	1	2	3	4
No. of units produced per hour worked	100	250	190	150

The only complication is that in changing from producing product 1 one day to producing product 2 the next day five working hours are lost (from the 24 hours available to produce product 2 that day) due to the necessity of cleaning certain oil tanks.

To assist in planning the production for the next week the following data is available:

Product	Current stock (units)	Demand (units) for each day of the week 1	2	3	4	5	6	7
1	5000	1500	1700	1900	1000	2000	500	500
2	7000	4000	500	1000	3000	500	1000	2000
3	9000	2000	2000	3000	2000	2000	2000	500
4	8000	3000	2000	2000	1000	1000	500	500

Product 3 was produced on day 0. The factory is not allowed to be idle (i.e. one of the four products must be produced each day). Stockouts are not allowed. At the end of day 7 there must be (for each product) at least 1750 units in stock.

If the cost of holding stock is £1.50 per unit for products 1 and 2 but £2.50 per unit for products 3 and 4 (based on the stock held at the end of each day) formulate the problem of planning the production for the next week as an integer program in which all the constraints are linear.

Solution

Variables

The decisions that have to be made relate to the type of product to produce each day. Hence let:

- x_{it} = 1 if produce product i (i=1,2,3,4) on day t (t=1,2,3,4,5,6,7) = 0 otherwise

In fact, for this problem, we can ease the formulation by defining two additional variables - namely let:

- I_{it} be the closing inventory (amount of stock left) of product i (i=1,2,3,4) on day t (t=1,2,...,7)
- P_{it} be the number of units of product i (i=1,2,3,4) produced on day t (t=1,2,...,7)

Constraints

- only produce one product per day

$$x_{1t} + x_{2t} + x_{3t} + x_{4t} = 1 \quad t=1,2,...,7$$

- no stockouts

$$I_{it} >= 0 \quad i=1,...,4 \ t=1,...,7$$

- we have an inventory continuity equation of the form

closing stock = opening stock + production - demand

Letting D_{it} represent the demand for product i (i=1,2,3,4) on day t (t=1,2,...,7) we have

$$I_{10} = 5000$$
$$I_{20} = 7000$$
$$I_{30} = 9000$$
$$I_{40} = 8000$$

representing the initial stock situation and

$$I_{it} = I_{it-1} + P_{it} - D_{it} \quad i=1,...,4 \ t=1,...,7$$

for the inventory continuity equation.

Note here that we assume that we can meet demand in month t from goods produced in month t and also that the opening stock in month t = the closing stock in month $_{t-1}$.

- production constraint

Let R_i represent the work rate (units/hour) for product i (i=1,2,3,4) then the production constraint is

$$P_{it} = x_{it}[24R_i] \quad i=1,3,4 \ t=1,...,7$$

which covers the production for all except product 2 and

$$P_{2t} = [24R_2]x_{2t} - [5R_2]x_{2t}x_{1t-1} \quad t=1,...,7$$

i.e. for product 2 we lose 5 hours production if we are producing product 2 in period t and we produced product 1 the previous period. Note here that we initialise by

$$x_{10} = 0$$

since we know we were not producing product 1 on day 0. Plainly the constraint involving P_{2t} is non-linear as it involves a term which is the product of two variables. However we can linearise it by using the trick that given three zero-one variables (A,B,C say) the non-linear constraint

- A = BC

can be replaced by the two linear constraints

- A <= (B + C)/2 and
- A >= B + C - 1

Hence introduce a new variable Z_t defined by the verbal description

Z_t = 1 if produce product 2 on day t and product 1 on day t-1
= 0 otherwise

Then

$$Z_t = x_{2t}x_{1t-1} \quad t=1,...,7$$

and our non-linear equation becomes

$$P_{2t} = [24R_2]x_{2t} - [5R_2]Z_t \quad t=1,...,7$$

and applying our trick the non-linear equation for Z_t can be replaced by the two linear equations

$$Z_t <= (x_{2t} + x_{1t-1})/2 \quad t=1,...,7$$

$$Z_t >= x_{2t} + x_{1t-1} - 1 \quad t=1,...,7$$

- closing stock

$$I_{i7} >= 1750 \quad i=1,...,4$$

- all variables >= 0 and integer, (x_{it}) and (Z_t) zero-one variables

Note that, in practise, we would probably regard (I_{it}) and (P_{it}) as taking fractional values and round to get integer values (since they are both quite large this should be acceptable).

Objective

We wish to minimise total cost and this is given by

$$SUM\{t=1,...,7\}(1.50I_{1t} + 1.50I_{2t} + 2.50_{I3t} + 2.50I_{4t})$$

Note here that this program may not have a feasible solution, i.e. it may simply not be possible to satisfy all the constraints. This is irrelevant to the process of constructing the model however. Indeed one advantage of the model may be that it will tell us (once a computational solution technique is applied) that the problem is infeasible.

Integer programming example 1987 UG exam **Problem 7**

A company is attempting to decide the mix of products which it should produce next week. It has seven products, each with a profit (£) per unit and a production time (man-hours) per unit as shown below:

Product	Profit (£ per unit)	Production time (man-hours per unit)
1	10	1.0
2	22	2.0
3	35	3.7
4	19	2.4

5	55	4.5
6	10	0.7
7	115	9.5

The company has 720 man-hours available next week.

- Formulate the problem of how many units (if any) of each product to produce next week as an integer program in which all the constraints are linear.

Incorporate the following additional restrictions into your integer program (retaining linear constraints and a linear objective):

- If any of product 7 are produced an additional fixed cost of £2000 is incurred.
- Each unit of product 2 that is produced over 100 units requires a production time of 3.0 man-hours instead of 2.0 man-hours (e.g. producing 101 units of product 2 requires 100(2.0) + 1(3.0) man-hours).
- If both product 3 and product 4 are produced 75 man-hours are needed for production line set-up and hence the (effective) number of man-hours available falls to 720 - 75 = 645.

Solution

Let x_i (integer >=0) be the number of units of product i produced then the integer program is

maximise

$10x_1 + 22x_2 + 35x_3 + 19x_4 + 55x_5 + 10x_6 + 115x_7$

subject to

$1.0x_1 + 2.0x_2 + 3.7x_3 + 2.4x_4 + 4.5x_5 + 0.7x_6 + 9.5x_7 <= 720$

$x_i >= 0$ integer i=1,2,...,7

Let

z_7 = 1 if produce product 7 ($x_7 >= 1$)
= 0 otherwise

then

- subtract $2000z_7$ from the objective function and
- add the constraint $x_7 <=$ [most we can make of product 7]z_7

Hence

$x_7 <= (720/9.5)z_7$
i.e. $x_7 <= 75.8z_7$
so $x_7 <= 75z_7$ (since x_7 is integer)

Let y_2 = number of units of product 2 produced in excess of 100 units then add the constraints

- $x_2 <= 100$
- $y_2 >= 0$ integer
- and amend the work-time constraint to be $1.0x_1 + [2.0x_2 + 3.0y_2] + 3.7x_3 + 2.4x_4 + 4.5x_5 + 0.7x_6 + 9.5x_7 <= 720$

- and add $+22y_2$ to the objective function.

This will work because x_2 and y_2 have the same objective function coefficient but y_2 requires longer to produce so will always get more flexibility by producing x_2 first (up to the 100 limit) before producing y_2.

Introduce

Z = 1 if produce both product 3 and product 4 ($x_3 >= 1$ and $x_4 >= 1$)
= 0 otherwise

z_3 = 1 if produce product 3 ($x_3 >= 1$)
= 0 otherwise

z_4 = 1 if produce product 4 ($x_4 >= 1$)
= 0 otherwise

and

- subtract from the rhs of the work-time constraint 75Z

and add the two constraints

- $x_3 <=$ [most we can make of product 3]z_3
- $x_4 <=$ [most we can make of product 4]z_4

i.e.

$x_3 <= 194z_3$ and $x_4 <= 300z_4$

and relate Z to z_3 and z_4 with the non-linear constraint

- $Z = z_3z_4$

which we linearise by replacing the non-linear constraint by the two linear constraints

- $Z >= z_3 + z_4 - 1$
- $Z <= (z_3 + z_4)/2$