Integer Programming Problems.

These problems include decisions that are inherently *discrete* in nature.

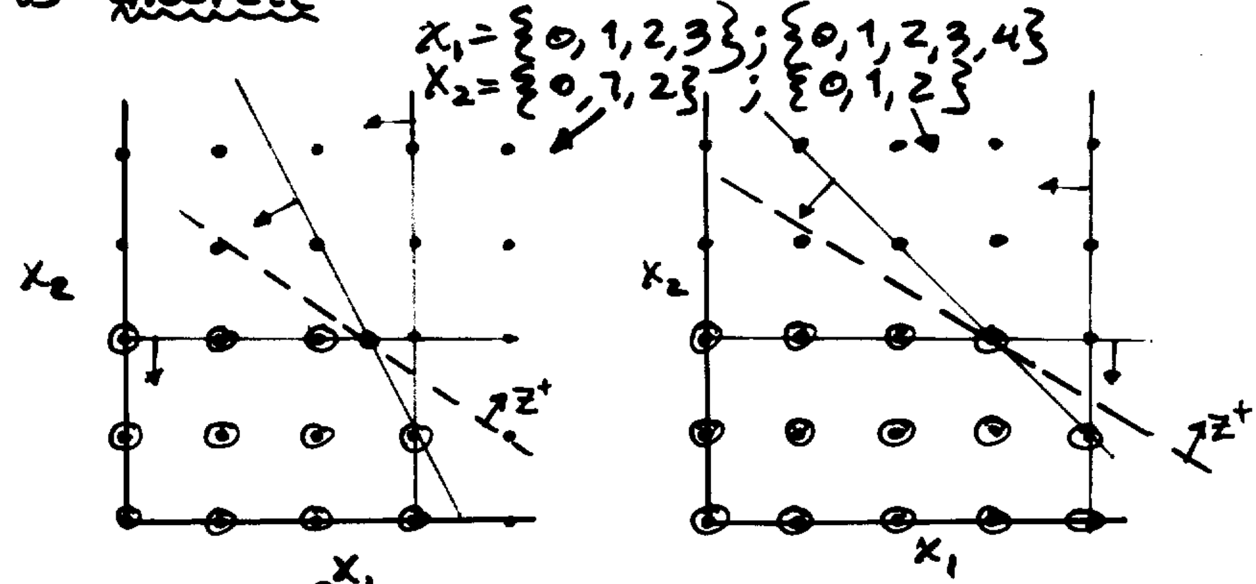
E.g. "Should the plant be built at location A or location B?"

A particularly common (and useful) category of integer programming problems are those which contain *binary* variables.

$$x = \{0, 1\}$$

where $x = 0$ suggests option A (e.g. "off")
$x = 1$ suggests option B (e.g. "on")

Note that the decision space for these types of problems is *discrete*



o - Feasible Region

"Integer Unfriendly"

(Optimal solution of 'Relaxed' LP problem, where integer variables are allowed to be continuous, is fractional)

"Integer Friendly"

(Optimal solution of Relaxed problem is 'naturally' Integer - just solve by LP Simplex Method)

So, by "Integer FRIENDLY", we mean that, if the integer restrictions on the variables are replaced by simple lower and upper bounds, allowing the variable to vary continuously, e.g.

LP 'relaxation' A Binary variable : $x = \{0, 1\}$
$\longrightarrow$ Replaced with : $0 \leq x \leq 1$

An Integer variable : $x = \{0, 1, 2, \ldots r\}$
LP 'relaxation' Replaced with : $0 \leq x \leq r$

Then the result from solution of the LP is integer-valued. If the result were fractional, then the problem would be "integer-unfriendly".

(*) The 1st step in solving any Integer Programming problem is to try to express it in an known integer friendly form, and then to attempt solution of the resulting 'relaxed' problem using the simplex algorithm.

These ideas are best illustrated using example problems...

LP Shipping Problem.

Determine the minimum cost shipping schedule for the following manufacturer of mouse traps at 3 factories with 4 Distribution centers:

\# mouse traps manufactured:

Factory	1	2	3
Loads/week	5	3	7

\# Truckloads required /week:

Dist Center	1	2	3	4
Loads/week	4	6	3	2

Shipping Costs:

	To Center:			
From Factory:	1	2	3	4
1	3	6	7	5
2	2	4	3	7
3	8	2	9	3

FORMULATION: Let X_{ij} = # truckloads shipped from $i \to j$

$$\text{Minimize } Z = \left.\begin{array}{l} 3X_{11} + 6X_{12} + 7X_{13} + 5X_{14} \\ + 2X_{21} + 4X_{22} \quad 3X_{23} \quad 7X_{24} \\ + 8X_{31} \quad 2X_{32} \quad 9X_{33} \quad 3X_{34} \end{array}\right\} \text{Cost of Shipping}$$

Subject to:

$$\left.\begin{array}{l} X_{11} + X_{21} + X_{31} = 4 \\ X_{12} + X_{22} + X_{32} = 6 \\ X_{13} + X_{23} + X_{33} = 3 \\ X_{14} + X_{24} + X_{34} = 2 \end{array}\right\} \text{Distrib. Center Requirements.}$$

$$\left.\begin{array}{l} X_{11} + X_{12} + X_{13} + X_{14} = 5 \\ X_{21} + X_{22} + X_{23} + X_{24} = 3 \\ X_{31} + X_{32} + X_{33} + X_{34} = 7 \end{array}\right\} \text{Factory Product Requirements.}$$

This problem can be formulated more generally:

Let: X_{ij} – Qty. shipped from source i to dest. j
C_{ij} – Cost of shipping unit qty. from $i \rightarrow j$
S_i – supply available from source i
d_j – demand of destination j
m – # of sources
n – # of destinations.

Minimize $$Z = \sum_{i=1}^{m} \sum_{j=1}^{n} C_{ij} X_{ij}$$

subject to: $$\sum_{i=1}^{m} X_{ij} = d_j \qquad j = 1, 2, \ldots, n$$

$$\sum_{j=1}^{n} X_{ij} = S_i \qquad i = 1, 2, \ldots, m$$

<u>Question</u>: What if the sources had not yet been built? The problem is, in part, to determine the <u>number</u> & <u>locations</u> of sources, from among a set of <u>potential</u> locations – and in addition to determine the optimal shipping pattern.

Define a new set of <u>binary</u> variables:

$y_i = \{0, 1\}$

$y_i = 0$ If a source is <u>not</u> constructed @ i

$y_i = 1$ If a source <u>is</u> constructed @ i

Also define f_i – the cost to <u>open</u> source i ('fixed' cost)

e_i – the cost to manufacture <u>each</u> <u>additional unit</u> @ i ('variable' cost)

Note: We no longer know the supply from each source, since it isn't constructed yet! → M – the maximum possible supply from any one source.

What is M?

$$M = \sum_{j=1}^{n} d_j = \text{Total demand.}$$

Now, the formulation...

$$\text{Minimize } Z = \underbrace{\sum_{i=1}^{m} f_i y_i}_{\substack{\text{Fixed Cost to} \\ \text{Construct Sources}}} + \underbrace{\sum_{i=1}^{m} \sum_{j=1}^{n} (c_{ij} + e_i) x_{ij}}_{\substack{\text{Variable shipping +} \\ \text{manufacturing Costs.}}}$$

Subject to:

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

Total shipped to Destination j

(Still need to satisfy the destination Requirement)

$$\sum_{j=1}^{n} x_{ij} \leq M y_i \qquad i = 1, 2, \ldots, m \qquad (2)$$

Total shipped & Manufactured at source i

Note how (2) 'Shuts off' source i if it is not constructed.
- A typical usage of binary variables.

Equal to zero when source i is not built, and equal to max. possible demand from source i if it is built.

$$x_{ij} \geq 0 \qquad i = 1, 2, \ldots, m; \; j = 1, 2, \ldots, n$$

$$y_i = \{0, 1\} \qquad i = 1, 2, \ldots, m$$

Constraint (2) is often written in a 'Tighter' form, which place more precise (and logical) restrictions on supply from i to each individual destination:

$$x_{ij} \leq d_j y_i \qquad \begin{matrix} i = 1, 2, \ldots, m \\ j = 1, 2, \ldots, n \end{matrix} \qquad (2a)$$

This is more 'integer friendly' than (2), but neither of these formulations is particularly 'integer friendly'.