Optimsation & Operations Research

Haide College, Spring Semester

Tutorial 5

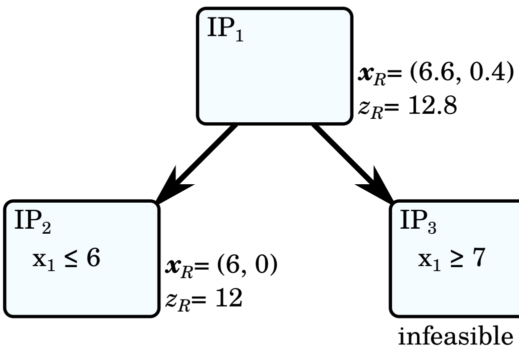
These questions are on Topic 4: Integer programming problems, and Topic 5: More sophisticated algorithms

---

1. **Calculations:** Perform Branch and Bound on the following problem:

$$
\begin{array}{lrcl}
\text{maximise} & 2x_1 - x_2 & & \\
\text{subject to} & x_1 + x_2 & \leq & 7 \\
 & 2x_1 - 3x_2 & \leq & 12 \\
 & x_1 \geq 0, x_2 \geq 0 & & \\
 & x_1, x_2 \text{ integer} & &
\end{array}
$$

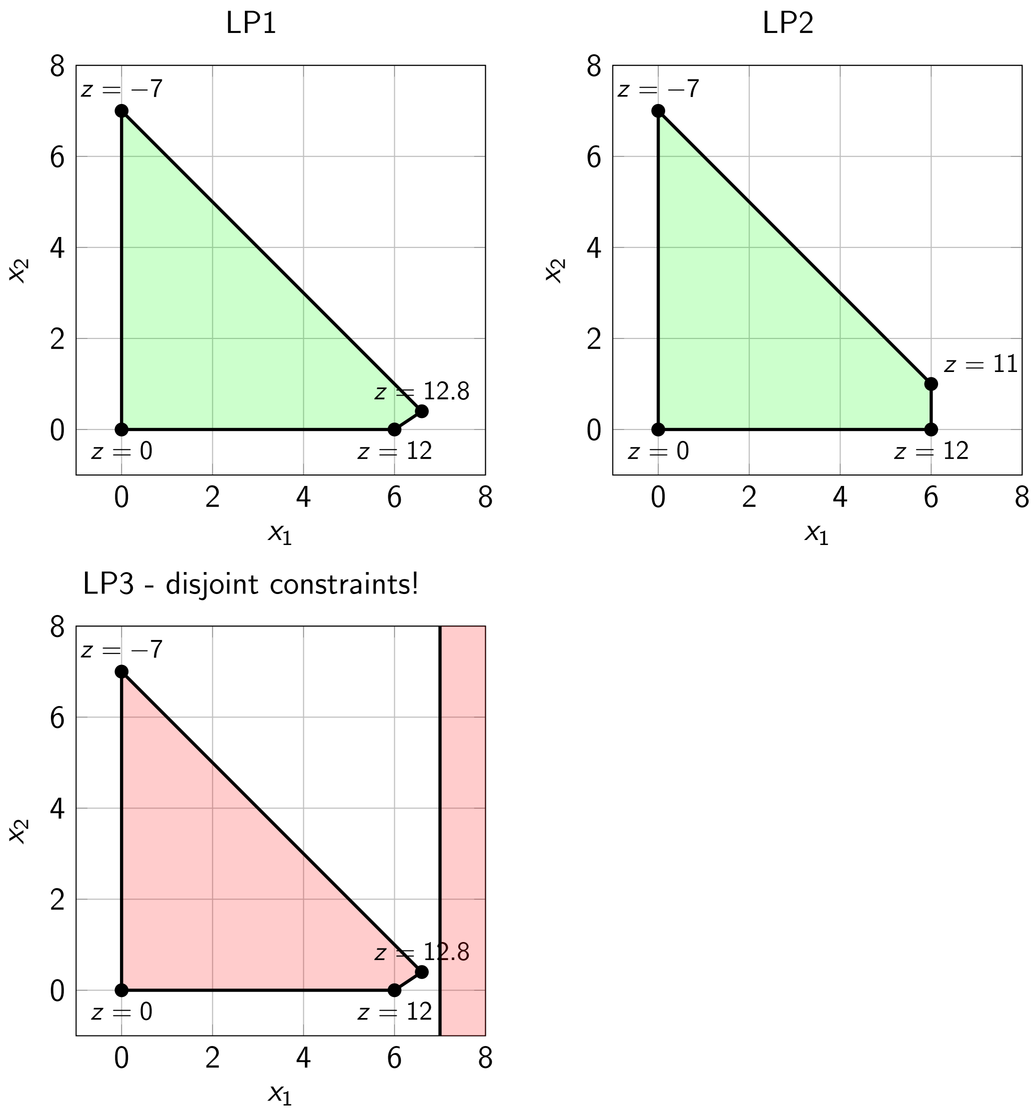
Solution: The solution is illustrated in the figure below. The figure shows each branch, but only shows the *extra* constraints of each branch problem.



- The initial problem LP_1 is derived by relaxing the integer constraints and solve using Simplex (or otherwise). It has solution $\mathbf{x}^T = (6.6, 0.4)$.
- We branch on x_1.
- LP_2 has the extra constraint that $x_1 \leq 6$, and solution $\mathbf{x}^T = (6, 0)$ which is integer feasible.

  Set $z_{ip} = 12$, and this branch is fathomed.
- LP_3 has $x_1 \geq 7$, which is infeasible, so this branch is also fathomed.

All branches are fathomed so the final solution is $\mathbf{x}^T = (6, 0)$, with $z^* = 12$
Can illustrate solution to relaxed problems graphically:

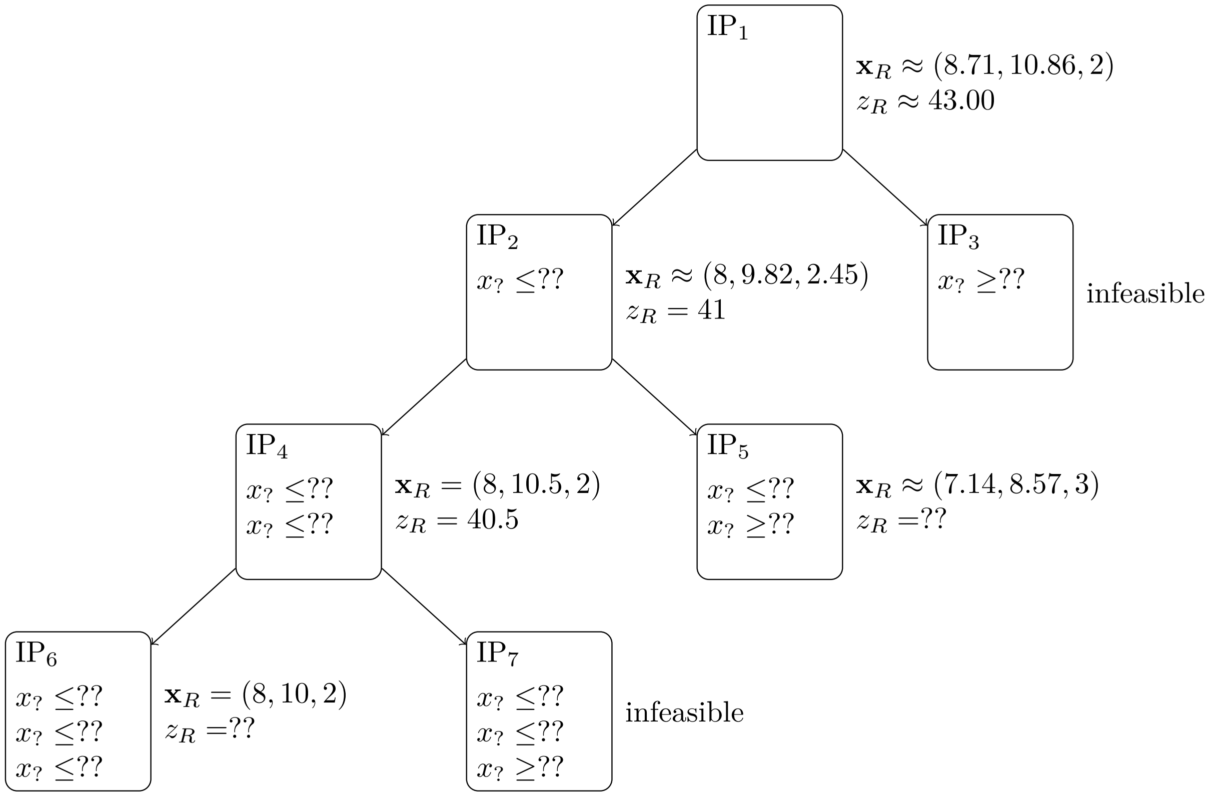


---

2. The following ILP (Integer Linear Program)

$$
\begin{array}{rcrcrcrcl}
\max z & = & 3x_1 & + & x_2 & + & 3x_3 & & \\
 & & -2x_1 & + & 4x_2 & - & 4x_3 & \leq & 18 \\
 & & 5x_1 & - & 3x_2 & + & x_3 & \leq & 13 \\
 & & -x_1 & + & 2x_2 & + & 3x_3 & \leq & 19
\end{array}
$$

with the $x_i \geq 0$ and integer, has been solved using Branch and Bound. The steps in the solution tree are shown in the figure below.

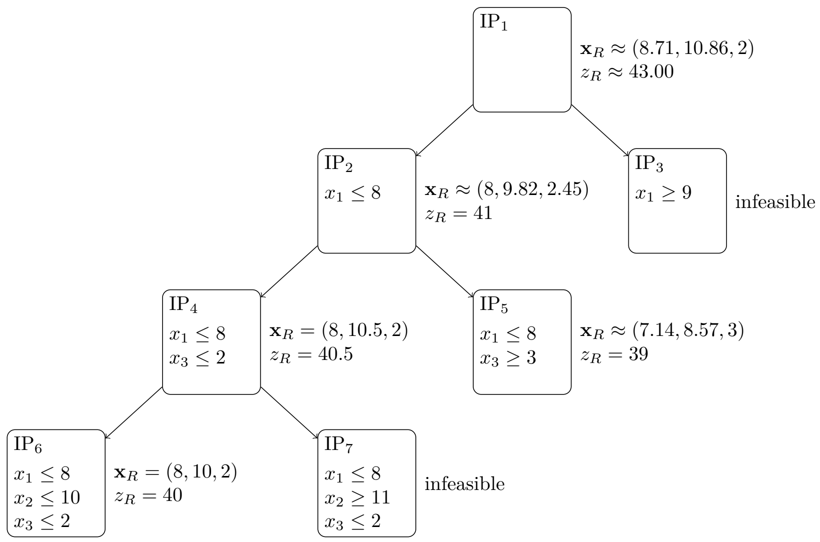


Note that the subproblems in the figure only show additional constraints added to the subproblem in addition to the original constraints of the ILP. Note also that the figure does not show all solutions to relaxed versions of the IPs (you will have to do some working yourself).

a) Write out a completed version of the tree diagram above by filling in :

- the constraints for all nodes
- the value of the objective z_R for the relaxed version of IP$_5$ and IP$_6$.

Solution: The figure below provides complete information about the B&B tree.



b) Explain why the tree is fathomed at IP$_3$, IP$_5$, IP$_6$ and IP$_7$.

> **Solution:** There are four fathomed leaf nodes:
>
> - IP_3 & IP_7 are fathomed because their relaxations are infeasible
> - IP_6 is fathomed because its relaxation is integer feasible.
> - IP_5 is fathomed because it is bounded by IP_6, which sets $z_{ip} = 40 > 39$.

c) Use the figure (and your working) to derive the optimal solution to the ILP.

> **Solution:** The best solution is given in IP_6, whose relaxation is integer feasible. The optimal solution is $z^* = 40$ at $\vec{x}^T = (8, 10, 2)$.

Bonus questions

1. **Translation:** *Ambulance depot location.* The ambulance service needs to locate a number of depots around the city so that it can get to any potential accident within 15 minutes.

   - There are M possible locations where we could place a depot.
   - Each location i can reach a subset S_i of the population within the required time.

   In the ambulance depot problem, the subsets might be described by regions represented say by circles around the hub, but in general we simply use sets, which makes the problem formulation more general.

   Each location also has a cost $c_i > 0$. We need choose which potential depot sites to use to:

   - minimise the cost of the overall system, and
   - ensure that every person is covered by at least one depot.

   Formulate the problem of minimising the cost of the ambulance depots as an Integer Linear Program.

> **Solution:** We have M subsets $S_1, S_2, \ldots, S_M$ of the population, where S_i represents the group that can be reached in a given time from potential depot i. Note that the total population $S = \cup_{i=1}^{M} S_i$ or the problem is infeasible.
> Define a_{ij} such that
> $$a_{ij} = \begin{cases} 1, & \text{if } i \in S_j, \\ 0, & \text{otherwise.} \end{cases}$$
> Take variables x_i where
> $$x_j = \begin{cases} 1, & \text{if we use depot location } j, \\ 0, & \text{otherwise.} \end{cases}$$
> Then the ILP expressing the problem is
> $$\min \left\{ \sum_{j=1}^{M} c_j x_j \middle| \sum_{j=1}^{M} a_{ij} x_j \geq 1, \forall i = 1, \ldots, N, \; x_j \in \{0, 1\}, \forall j = 1, \ldots, M \right\}$$

NB: this is a special instance of the *set-covering problem* (though sometimes in that problem the costs are all equal). It applies to many other optimisation problems: most obviously other similar problems such as placement of other emergency services, but also to such diverse areas as optimised supply chains (the sets represent potential suppliers, and the population are the components that need to be supplied); working out how to obtain required expertise for a project; construction of optimal logical circuits; and air-crew scheduling for airlines.
It is interesting because it was one of the 21 early problems shown to be NP complete in "Reducibility Among Combinatorial Problems", by Richard Karp from Complexity of Computer Computations, pp. 85-103, 1972.