Programmazione Lineare Intera

Massimo Pappalardo

University of Pisa, Italy
massimo.pappalardo@unipi.it

Ricerca Operativa
Placing ambulances in Milan

Ambulances in Milan
  ▶ They are placed in given positions waiting for a call.
  ▶ Serious ill patients must be reached at most in 8 minutes

Problem
How to place ambulances to guarantee a good service with a minimal number of them?
What to do?

We can represent the problem with a mathematical model

**Define the problem that we want to model**

- Use the minimum number of ambulances
- Guarantee to arrive to any place at most in 8 minutes
Structure of the mathematical model

- Data ➔ parameters

\[ I = \text{set of possible locations of ambulances} \]

\[ J = \text{set of demand positions} \]

\[ a_{ij} = \begin{cases} 
1 & \text{if it is possible to go from } i \text{ to } j \text{ at most in 8 minutes} \\
0 & \text{otherwise} 
\end{cases} \]
Structure of the mathematical model

- Decisions (where to put ambulances) → variables
  
  \[ x_i = \begin{cases} 
  1 & \text{if an ambulance is located at place } i \\
  0 & \text{otherwise} 
  \end{cases} \]

- The goal/objective (use the smallest number of ambulances as possible) → objective function

- Requirements that a solution must satisfy (everyone can be reached at most in 8 minutes) → constraints that variables must satisfy
Mathematical model

**Objective function**
Minimize the number of ambulances

$$\min \sum_{i \in I} x_i$$

**Constraints**
For each location $j$ at least one ambulance can reach it at most in 8 minutes

$$\sum_{i \in I} a_{ij} x_i \geq 1, \quad \forall j \in J$$

**Variables domain**

$$x_i \in \{0, 1\}, \quad \forall i \in I$$
Relations between ILP and LP

Consider an Integer Linear Programming (ILP) in canonical form:

\[
\begin{align*}
\max & \quad c^T x \\
A x & \leq b \\
x & \in \mathbb{Z}^n
\end{align*}
\]  

(\(P\))

**Definition**

The LP problem

\[
\begin{align*}
\max & \quad c^T x \\
A x & \leq b
\end{align*}
\]  

(CR)

is said *continuous relaxation* of the problem (\(P\)).
What is the relation between (\(P\)) and (CR)?

**Theorem**

- The optimal value of (CR) is an upper bound for the optimal value of (\(P\)).
- If an optimal solution of (CR) is feasible for (\(P\)), then it is optimal for (\(P\)).

... but usually the optimal solution of (CR) is unfeasible for (\(P\)) ...
Relations between ILP and LP

In order to solve \((P)\), is it sufficient to solve \((CR)\) and round the solution? NO

Example

\[
\begin{align*}
\text{max} & \quad x_1 + 3x_2 \\
x_1 + 5x_2 & \leq 21 \\
8x_1 + 2x_2 & \leq 35 \\
x & \geq 0 \\
x & \in \mathbb{Z}^2
\end{align*}
\]

\[
\left(\frac{7}{2}, \frac{7}{2}\right)
\]

optimal for \((CR)\)

rounding \(\rightarrow (3, 3)\)

\((3, 3)\) is NOT optimal for \((P)\)

\((1, 4)\) is optimal for \((P)\)
Relations between ILP and LP

In order to solve (P), is it sufficient to solve (CR) and find the closest feasible integer solution? NO

Example

\[
\begin{aligned}
\text{max } & \ x_1 + 3 x_2 \\
& x_1 + 5 x_2 \leq 21 \\
& 8 x_1 + 2 x_2 \leq 35 \\
& x \geq 0 \\
& x \in \mathbb{Z}^2
\end{aligned}
\]

\[\left(\frac{7}{2}, \frac{7}{2}\right)\] optimal for (CR)

the closest feasible integer solution is (3, 3)

(3, 3) is not optimal for (P)

(1, 4) is optimal for (P)
Explicit enumeration

Consider the ILP problem

\[
\begin{aligned}
\max & \quad 5x_1 + 6x_2 \\ 3x_1 + 4x_2 & \leq 7 \\
& x_1 \geq 0, \quad x_2 \geq 0 \\
& x \in \mathbb{Z}^2
\end{aligned}
\]  

(P)

Constraints imply \(x_1 = 0\) or \(1\) or \(2\).

Write a partition of the feasible region \(\Omega\) into three subsets:

\[
\Omega = (\Omega \cap \{x_1 = 0\}) \cup (\Omega \cap \{x_1 = 1\}) \cup (\Omega \cap \{x_1 = 2\})
\]

corresponding to the first level of the decision tree:

```
\begin{tikzpicture}
  \node (P) at (0,0) {P};
  \node (P11) at (-3,-1) {P_{1,1}}; 
  \node (P12) at (0,-1) {P_{1,2}}; 
  \node (P13) at (3,-1) {P_{1,3}};
  \draw (P) -- (P11) node [midway, above] {$x_1 = 0$};
  \draw (P) -- (P12) node [midway, above] {$x_1 = 1$};
  \draw (P) -- (P13) node [midway, above] {$x_1 = 2$};
\end{tikzpicture}
```
Explicit enumeration

Similarly, $x_2 = 0$ or $1$. Therefore, the complete decision tree is

Nodes $P_{2,1}, \ldots, P_{2,5}$ correspond to feasible solutions of $(\mathcal{P})$, while node $P_{2,6}$ corresponds to $x = (2, 1)$ which is unfeasible.

The objective function value for the nodes $P_{2,1}, \ldots, P_{2,5}$ are $0, 6, 5, 11, 10$. Thus, the optimal solution is given by $P_{2,4}$, i.e., $x^* = (1, 1)$. 
Implicit enumeration

- **Explicit** enumeration is very computational demanding
- Sets of solutions instead of single solution can be evaluated (and discarded)
- **Implicit** enumeration
Implicit enumeration

\[
\begin{align*}
\text{max} & \quad x_1 + 3x_2 \\
& \quad x_1 + 5x_2 \leq 21 \\
& \quad 8x_1 + 2x_2 \leq 35 \\
& \quad x \geq 0, \quad x \in \mathbb{Z}^2
\end{align*}
\]

We know that (3, 3) is feasible with value 12.

Consider the additional constraint \(x_1 \geq 4\): the optimum of the continuous relaxation of the subproblem is (4, 3/2) with value 8.5, hence feasible solutions of the subproblem are worse than (3, 3) → implicit enumeration.
Reduced decision tree

\[ PB = 12 \]

\[ P_{1,1} \]
\[ x_1 \leq 3 \]

\[ P \]
\[ x_1 \geq 4 \]

\[ UB(P_{1,2}) = 8 \]
The basic idea of the *Branch and Bound* method is:

- The feasible region is partitioned, generating a searching or branching tree
- For each subregion (corresponding to a subproblem) the optimal value is approximated by evaluating a bound
- The regions that cannot contain the global optimum are discarded (pruned) and are not further investigated
Branch and bound method

Main components

- **Branching**: how to partition the feasible region to generate the subregions associated with subproblems
- **Bounding**: how to estimate the value of the optimum in one subregion
- **Fathoming**: how to prune the subregions that do not contain the optimal solution (or how to close nodes of the searching tree)
- **Tree search**: in which order the subproblems are visited
Branching rules

How to partition the feasible region

- Subregions are generated by adding constraints
- Subregions must be a partition of the integer feasible solution sets, so as to guarantee that no integer solution is discarded
Branching rules

Bipartite branching
Each region is partitioned into two subregions, adding two constraints
E.g. \( x_1 \leq 3 \) and \( x_1 \geq 4 \)
Corresponding branching tree

\[ P \]

\[ x_1 \leq 3 \quad x_1 \geq 4 \]

\[ P_1 \quad P_2 \]
**k-partite branching**

Each region is divided in $k$ subregions, adding $k$ constraints

E.g. $x_1 = 0$, $x_1 = 1$, $x_1 = 2$, $x_1 = 3$, and $x_1 = 4$
Corresponding branching tree

\[
x_1 = 0 \quad x_1 = 1 \quad x_1 = 2 \quad x_1 = 3 \quad x_1 = 4
\]

- \( P_1 \)
- \( P_2 \)
- \( P_3 \)
- \( P_4 \)
- \( P_5 \)
Bounds

Lower bound LB given by a feasible solution.

Upper bound UB (of each subproblem) given by a relaxation:

- Continuous relaxation (integrality constraints are discarded)
  \[ x \in \{0, 1\} \Rightarrow 0 \leq x \leq 1 \]
  \[ x \in \mathbb{Z}_+ \Rightarrow x \geq 0 \]
- Constraint elimination
- sum of two constraints
- \ldots
Fathoming or pruning criteria

A node of the decision tree can be pruned if any of the following conditions holds:

- the subproblem is unfeasible
- $UB$ of the subproblem is $\leq LB$ of $(P)$
- $UB$ of the subproblem is $> LB$ and the optimal solution of the subproblem is feasible for $(P)$. In such a case we update $LB = UB$
Tree search strategies

In which order are the subregions investigated?

**Depth first**

- The next node to be explored is one of the children of the currently explored node
- this strategy quickly finds a feasible integer solution
- it needs limited memory
- the obtained solution may be a poor quality one

**Best first**

- The next node to be visited is the most promising one (e.g. the one with the best UB)
- it requires a lot of memory
- it focuses on the solution quality

**Breadth first**

- All the nodes of the same level are explored
- it is not commonly used, as it does not performs well
Method scheme

1. Compute a lower bound $LB$ of the optimal value of $(P)$
2. if all the nodes have been explored then STOP
3. select the node (subproblem) to be investigated
4. compute an upper bound $UB$ of the optimal value of the subproblem
5. Check fathoming criteria:
   - if the subproblem is unfeasible, then close the node
   - if $UB \leq LB$, then close the node
   - if $UB > LB$ and the optimal solution of the subproblem is feasible for $(P)$,
     then close the node and update $LB = UB$
6. if the node is not closed, then branch
7. go to Step 2
Example

Apply the Branch and Bound method for solving the problem

\[
\begin{align*}
\max & \quad x_1 + 3 x_2 \\
x_1 + 5 x_2 & \leq 21 \\
8 x_1 + 2 x_2 & \leq 35 \\
x \geq 0, \quad x \in \mathbb{Z}^2.
\end{align*}
\]

(\(P\))

We use a bipartite branching and a “depth first” search strategy.

We know that the optimal solution of the continuous relaxation is

\((7/2, 7/2)\) hence \(UB(P) = 14\).

We know the feasible solution \((3, 3)\) hence \(LB = 12\).
Example

The optimal solution of the continuous relaxation of $P_1$ is $(3, 18/5)$, it is unfeasible with value 13.8, hence $UB(P_1) = 13 > 12 = LB$. Node $P_1$ remains open.
Branch $x_2 \leq 3$ and $x_2 \geq 4$. 
Example

The optimal solution of the continuous relaxation of $P_2$ is $(3, 3)$, hence $UB(P_2) = 12 = LB$, we close node $P_2$.
The optimal solution of the continuous relaxation of $P_3$ is $(1, 4)$ and $UB(P_3) = 13 > 12 = LB$. Since $(1, 4)$ is feasible for $P$, we update $LB = 13$ and close node $P_3$.
The optimal solution of the continuous relaxation of $P_4$ is $(4, 3/2)$ with value 8.5, hence $UB(P_4) = 8 < 13 = LB$. We close node $P_4$.

Since all the nodes are closed, the optimal solution of $P$ is $(1, 4)$ and the optimal value is 13.