Brief History of LP

• Calculus invented in 17th century to study mechanics.
  LP invented in 20th century to study "Operations/Production Planning".
• As early as 1826: Fourier studied system of inequalities.
• 1947: Dantzig (USA), "Faster q LP": Simplex Algorithm to solve USA Airforce planning problem.
• 1947: Koopmans (USA): Applications of LP in Economic System.
• 1954: Frisch (USA): Logarithmic barrier fn. for LP.
• 1975: Kantorovich & Koopmans: Nobel prize in Economics (Dantzig considered only a "Mathematician").
• 1979: L.G. Khachian (USSR): Ellipsoidal Method for LP (proved LP can be solved polynomially; Simplex method although exponential in complexity, works faster on most practical problems).
• 1984: Karmarkar (USA/India): Interior point method for LP. (This method also polynomial and also faster than simplex in practice.)

Note: Theory of computational complexity developed in 60's, after simplex was invented. So when it was invented, notion of computational complexity did not exist.
• 1972: Klee & Minty: Simplex is exponential.
• 1979: O(n^5) algo. by Khachian (ellipsoidal method).
• 1984: O(n^{3.5}) algo. by Karmarkar (interior-point method).
Linear Mixed/Integer Programming

**Programming**: Optimization subject to "static" constraints

**Optimize**: \( f(x_1, \ldots, x_n) \) s.t. \( g_i(x_1, \ldots, x_n) \leq 0 \) \( i = 1, \ldots, m \)

**Linear prog. (LP)**: \( f \) & \( g_i \) linear fn.

**Integer prog. (IP)**: decision var. \( x_j \) take integer value.

**Mixed integer prog. (MIP)**: Some decision var. integer-valued, others real-valued

**Linear Mixed Integer Prog (LMIP)**: LP + MIP

**Having integer-valued decision variables makes problem harder.**

**We begin with study of LIP (= LP + IP)**

**Example (Knapsack problem)**: Pack items in a knapsack to maximize utility subject to weight limit

<table>
<thead>
<tr>
<th>Item</th>
<th>Weight</th>
<th>Utility</th>
<th>Utility/Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5</td>
<td>16</td>
<td>3.23</td>
</tr>
<tr>
<td>2</td>
<td>7</td>
<td>22</td>
<td>3.14</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>12</td>
<td>3.00</td>
</tr>
<tr>
<td>4</td>
<td>8</td>
<td>8</td>
<td>2.67</td>
</tr>
</tbody>
</table>

\[
\text{max } z = 16x_1 + 22x_2 + 12x_3 + 8x_4 \\
\text{st. } 5x_1 + 7x_2 + 4x_3 + 3x_4 \leq 14 \quad x_j = 0 \text{ or } 1.
\]

- This is an instance of LBP (linear binary prog.), a special case of LIP.
- One would suspect that optimal solution will include item that has maximum utility per unit weight. But contrary to this intuition, optimal solution is: \( x_1 = 0, x_2 = x_3 = x_4 = 1 \).

**Example (Set-Covering)**: Given two sets, cover each element of set 1 by an element of set 2 optimally.

<table>
<thead>
<tr>
<th>City 1</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>10</td>
<td>20</td>
<td>30</td>
<td>30</td>
<td>20</td>
</tr>
<tr>
<td>2</td>
<td>10</td>
<td>0</td>
<td>25</td>
<td>35</td>
<td>20</td>
<td>10</td>
</tr>
<tr>
<td>3</td>
<td>20</td>
<td>25</td>
<td>0</td>
<td>15</td>
<td>30</td>
<td>20</td>
</tr>
<tr>
<td>4</td>
<td>30</td>
<td>35</td>
<td>15</td>
<td>0</td>
<td>15</td>
<td>25</td>
</tr>
<tr>
<td>5</td>
<td>30</td>
<td>20</td>
<td>30</td>
<td>15</td>
<td>0</td>
<td>14</td>
</tr>
<tr>
<td>6</td>
<td>20</td>
<td>10</td>
<td>25</td>
<td>14</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Locate fire-stations in as few cities as possible so that each city is within 15 mins of at least one fire-station.
LIP (Examples)

• Fire-station location problem is an instance of set-covering (each city to be covered by some fire-station).

• \( x_j = 0 \) or 1; \( x_j = 1 \) if fire-station located at city \( j \).

• \( \min z = x_1 + x_2 + x_3 + x_4 + x_5 + x_6 \)

\[ \text{s.t. } \begin{align*}
    x_1 + x_2 &\geq 1 \quad (\text{city 1 can be covered by city 1 or 2}) \\
    x_1 + x_2 + x_6 &\geq 1 \\
    x_3 + x_4 &\geq 1 \\
    x_3 + x_4 + x_5 &\geq 1 \\
    x_4 + x_5 + x_6 &\geq 1 \\
    x_5 + x_6 &\geq 1 \quad (\text{city 6 can be covered by city 2 or 5 or 6})
\end{align*} \]

**Graphical Illustration:**

\[
\text{max } z = 21x_1 + 11x_2 \]

\[ \text{s.t. } \begin{align*}
    7x_1 + 4x_2 &\leq 13 \\
    x_1, x_2 &\geq 0 \text{, integer}
\end{align*} \]

From the graph, 6 feasible solutions:

\[
\begin{align*}
    (0,0) &\Rightarrow z = 0 \\
    (1,0) &\Rightarrow z = 21 \\
    (0,1) &\Rightarrow z = 11 \\
    (1,1) &\Rightarrow z = 32 \\
    (0,3) &\Rightarrow z = 22 \\
    (0,3) &\Rightarrow z = 33
\end{align*} \]

If real-valued solutions allow, then infinitely many feasible solutions exist. But from theory of LP, we know optimal occurs at a corner point.

From graph only 3 corner points:

\[
\begin{align*}
    (0,0) &\Rightarrow z = 0 \\
    \left( \frac{13}{7}, 0 \right) &\Rightarrow z = 39 \\
    (0, \frac{13}{4}) &\Rightarrow z = 35.75
\end{align*} \]

• Graph illustrates (i) LIP optimal value \( \leq \) LP optimal value

(ii) LIP optimal location very different from LP optimal location.

"LP relaxation" can be used to bound optimal value, but not to approximate the optimal solution.