Lecture 11: Integer programming

David C. Parkes
SEAS

Lesson Plan

• Integer programs
• Examples: Packing, Covering, TSP problems
• Modeling approaches
  – fixed costs
  – penalizing a violated constraints
  – alternate constraints
  – conditional constraints
  – n-fold constraints
• Formulation strength

Jensen & Bard: 7.1 – 7.7
What’s an IP?

- \( \max c^T x \) (IP)
  \[
  \text{s.t. } \begin{align*}
  Ax & \leq b \\
  x & \geq 0 \text{ and integer}
  \end{align*}
  \]

- \( \max c^T x \) (Binary IP)
  \[
  \text{s.t. } \begin{align*}
  Ax & \leq b \\
  x & \in \{0,1\}^n
  \end{align*}
  \]

- \( \max c^T x + h^T y \) (Mixed IP)
  \[
  \text{s.t. } \begin{align*}
  Ax + Gy & \leq b \\
  x & \geq 0 \\
  y & \geq 0 \text{ and integer (or binary)}
  \end{align*}
  \]

Applications

- Scheduling problems
  - e.g., crew and fleet scheduling (assign each crew and plane to a particular route)
- Procurement
  - e.g., hospital system determining which suppliers to use for sourcing of medical/surgical equipment
- Electricity generation
  - e.g., decide when to start-up plants, and what levels to run each plant at
- Kidney matching
  - e.g., find swaps or chains of patient-donor pairs
- Facility location
  - e.g., locate points of distribution during a disaster relief operation
Will IPs be easier or harder to solve?

\[
\begin{align*}
\text{max } & \quad 5x_1 + 8x_2 \\
\text{s.t. } & \quad x_1 + x_2 \leq 6 \\
& \quad 5x_1 + 9x_2 \leq 45 \\
& \quad x_1, \quad x_2 \geq 0, \text{ integer}
\end{align*}
\]

<table>
<thead>
<tr>
<th></th>
<th>cont</th>
<th>round off</th>
<th>nearest feas</th>
<th>integer</th>
</tr>
</thead>
<tbody>
<tr>
<td>(x_1)</td>
<td>2.25</td>
<td>2</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>(x_2)</td>
<td>3.75</td>
<td>4</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>(z)</td>
<td>41.25</td>
<td>infeas</td>
<td>34</td>
<td>40</td>
</tr>
</tbody>
</table>

optimal integer solution \(x^*=\langle 0,5 \rangle, \; z^*=40\)

optimal continuous solution \(x^*=\langle 9/4, \; 15/4 \rangle, \; z^*=41.25\)
Modeling IPs

• Step 1: Define decision variables
• Step 2: Define constraints to capture the various elements of the problem
  – introduce additional variables as necessary
• Step 3: Define the objective function

Example: Project Selection

• Budget $b$ to invest in projects.
• Project $j \in \{1, \ldots, n\}$ has cost $a_j$ and value $c_j$.
• Let $x_j = 1$ if project $j$ selected, $x_j = 0$ otherwise
• \[ \text{max } \sum_j c_j x_j \quad \text{ (maximize value)} \]
  \[ \text{s.t. } \sum_j a_j x_j \leq b \quad \text{ (budget constraint)} \]
  \[ x_j \in \{0,1\} \quad \text{ (no fractional projects)} \]
Example: Facility Location

• Locate emergency response centers. Location \( j \in \{1, \ldots, n\} \) has cost \( c_j \), and services regions \( R_j \subseteq R \). Regions \( i \in \{1, \ldots, m\} \).

• **Goal**: min total cost, but serve each region.

• \( a_{ij} = 1 \) if region \( i \) can be served by center \( j \), with \( a_{ij} = 0 \) otherwise.

• Let \( x_j = 1 \) if center \( j \) selected, \( x_j = 0 \) otherwise.

\[
\begin{align*}
\text{min } \sum_j c_j x_j & \quad \text{(min cost)} \\
\text{s.t. } \sum_i a_{ij} x_j \geq 1, & \quad i \in R \quad \text{(cover each region)} \\
& \quad x_j \in \{0, 1\} \quad \text{(no fractional centers)} 
\end{align*}
\]

Example: Traveling Salesperson

• A “tour” visits each of \( N = \{1, \ldots, n\} \) cities once and returns to start. Cost \( c_{ij} \geq 0 \) to travel \( i \) to \( j \).

• **Goal**: find tour that minimizes total cost.

A feasible tour in a seven-city TSP

Examples: FedEx pick-up, robot placing modules on a circuit board, student visiting colleges.
Example: Traveling Salesperson

- How many solutions?
- Starting at city 1, there are n-1 choices
- For the next city, n-2 choices,…
- (n-1)! feasible tours

<table>
<thead>
<tr>
<th>n</th>
<th>log n</th>
<th>n^0.5</th>
<th>n^2</th>
<th>2^n</th>
<th>n!</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>3.32</td>
<td>3.16</td>
<td>10^2</td>
<td>1.02x10^3</td>
<td>3.6x10^6</td>
</tr>
<tr>
<td>100</td>
<td>6.64</td>
<td>10.00</td>
<td>10^4</td>
<td>1.27x10^30</td>
<td>9.33x10^157</td>
</tr>
<tr>
<td>1000</td>
<td>9.97</td>
<td>31.62</td>
<td>10^6</td>
<td>1.07x10^301</td>
<td>4.02x10^2567</td>
</tr>
</tbody>
</table>

Example: Traveling Salesperson

Proctor & Gamble ran a contest in 1962. The contest required solving a TSP on 33 cities.

http://www.tsp.gatech.edu/history/pictorial/pictorial.html
Groetschel (1977) found the optimal tour of 120 cities from what was then West Germany.

Applegate, Bixby, Chvátal, and Cook (2001) found the optimal tour of 15,112 cities in unified Germany.
Applegate, Bixby, Chvátal, Cook, and Helsgaun (2004) found the optimal tour of 24,978 cities in Sweden. This was at the time the largest solved TSP problem.

Example: Traveling Salesperson

- \( x_{ij} = 1 \) if tour visits \( j \) immediately after \( i \), \( x_{ij} = 0 \) otherwise
- Objective: \( \min \sum_i \sum_j c_{ij} x_{ij} \)
- \( \sum_{j:j \neq i} x_{ij} = 1, \ \forall \ i \) “leave \( i \) once”
- \( \sum_{i:i \neq j} x_{ij} = 1, \ \forall \ j \) “enter \( j \) once”

Not quite right: this allows infeasible subtours:
Example: Traveling Salesperson

• “Subtour elimination” constraints
• Must not make more than $|S|-1$ trips between any strict subset $S$ of cities:
• $\sum_{i \in S} \sum_{j \in S} x_{ij} \leq |S|-1$ for $S \subseteq N$

• A new problem: this requires #constraints exponential in $n$.

A Succinct TSP formulation

• New variables: $t_j$ for $j \in \{1, \ldots, n\}$, denote the position in sequence at which city $j$ is visited (indexed by $i$ or by $j$)
• Let $t_1=1$ (wlog). Valid tour requires $t_j \geq t_i+1$ if $x_{ij}=1$ except for $j=1$.

Capture this way:

$t_j \geq t_i+1-n(1-x_{ij})$ for all $i$ and $j$ ($j \neq 1$)

$\Rightarrow x_{ij}=0$: $t_j \geq t_i-6$
$\Rightarrow x_{ij}=1$: $t_j \geq t_i+1$
Example: Radiation treatment

- Beams $b \in B$, pixels $(i,j) \in I$, tumor $T$, critical $C$
- $x_b = \text{power on beam } b$; $d_{ij}^b$ relative intens $b$
- $\min \varepsilon_T + \varepsilon_C$
  s.t. $D_{ij} + \varepsilon_T \geq \gamma_L \quad \forall (i,j) \in T$
  $D_{ij} - \varepsilon_C \leq \gamma_U \quad \forall (i,j) \in C$
  $D_{ij} = \sum_{b \in B} d_{ij}^b x_b \quad \forall (i,j) \in I$
  $x_b \geq 0$, $D_{ij} \geq 0$, $\varepsilon_T, \varepsilon_C \geq 0$

- What if each beam used also has “cost” $w$?
- Goal: $\min \varepsilon_T + \varepsilon_C + \text{“total cost of beams used”}$

Modified Radiation Formulation

- $\min \varepsilon_T + \varepsilon_C + \sum_{b \in B} w \alpha_b$
  s.t. $D_{ij} + \varepsilon_T \geq \gamma_L \quad \forall (i,j) \in T$
  $D_{ij} - \varepsilon_C \leq \gamma_U \quad \forall (i,j) \in C$
  $D_{ij} = \sum_{b \in B} d_{ij}^b x_b \quad \forall (i,j) \in I$
  $x_b \leq M \alpha_b \quad (*)$
  $x_b \geq 0$, $D_{ij} \geq 0$, $\varepsilon_T, \varepsilon_C \geq 0$
  $\alpha_b \in \{0,1\}$
  “indicator variable”

- Pick constant $M$ so (*) is satisfied when $\alpha_b = 1$.
- Need $M \geq \max$ possible power. A “big M” 😊
Modeling approaches

1. Fixed costs
2. Penalizing a Violated Constraint
3. Alternate constraints
4. Conditional constraints
5. n-fold constraints

1. Modeling Fixed costs

Cost of $K>0$ if used, and additional cost $c \times x$ per unit of power.
1. Modeling Fixed costs

\[
\text{cost} = \begin{cases} 
0, & \text{if } x_b = 0 \\
K + c x_b, & \text{if } x_b > 0 
\end{cases}
\]

\( \alpha_b \) = indicator variable for when the fixed cost is incurred, so \( \alpha_b = 1 \) when \( x_b > 0 \) and \( \alpha_b = 0 \) when \( x = 0 \)

- Can write cost = \( K \alpha_b + c x_b \)
- Constraints:
  \[ x_b \leq M \alpha_b \]
  \[ x_b \geq 0, \text{ and } \alpha_b \in \{0, 1\} \]

when \( x_b > 0 \), then \( \alpha_b = 1 \).

2. Penalizing a Violated Constraint

- Penalize the objective if “\( f(x) > b \)”.
- “the average power to critical region should not be more than 40% above \( \gamma_U \)”

- Let \( N_C = \#\text{pixels in C} \)
- Require:
  \[
  \sum_{ij \in C} D_{ij} \leq 1.4 N_C \gamma_U 
  \]

Add: \( \sum_{ij \in C} D_{ij} - M \alpha \leq 1.4 N_C \gamma_U \) where \( \alpha \in \{0, 1\} \).
Penalize \( \alpha \) in objective. Need M bigger than biggest possible difference.
2. Penalizing a Violated Constraint

• Penalize the objective if “f(x) > b”.

• Introduce an **indicator variable** $\alpha \in \{0, 1\}$. Write:
  
  $f(x_1, \ldots, x_n) - M\alpha \leq b$

• Define constant $M$ to be larger than $\max_x [f(x) - b]$, so that $(x, \alpha = 1)$ feasible for any $x$.

• Penalize $\alpha$ in objective.

3. Modeling alternate constraints

• “The power to critical region must be 20% or more below $\gamma_U$ or the power to tumor region must be 30% or more above $\gamma_L$ (or both).”

\[
\begin{align*}
\sum_{ij \in C} D_{ij} - M_1\alpha_1 & \leq 0.8 N_C \gamma_U & \text{(if violated, alpha1=1)} \\
\sum_{ij \in T} D_{ij} + M_2\alpha_2 & \geq 1.3 N_T \gamma_L & \text{(if violated, alpha2=1)} \\
\alpha_1 + \alpha_2 & \leq 1 & \text{(can't have both 1)} \\
\alpha_1, \alpha_2 & \in \{0, 1\}
\end{align*}
\]
3. Modeling alternate constraints

- Need at least one of:
  \[ f_1(x) \leq b_1 \quad (*) \]
  \[ f_2(x) \leq b_2 \quad (**) \]

- Introduce indicator variables \( \alpha_1, \alpha_2 \in \{0,1\} \)
  \[ f_1(x) - M_1 \alpha_1 \leq b_1 \quad (\text{if } * \text{ violated, need } \alpha_1=1) \]
  \[ f_2(x) - M_2 \alpha_2 \leq b_2 \quad (\text{if } ** \text{ violated, need } \alpha_2=1) \]
  \[ \alpha_1 + \alpha_2 \leq 1 \]

Big M constants:
\[ M_1 \geq \max_x [f_1(x)-b_1] \]
\[ M_2 \geq \max_x [f_2(x)-b_2] \]

- Can also use only one \( \alpha \) by defining \( \alpha_2=1-\alpha_1 \)

4. Modeling conditional constraints

- “If worst-case power to critical region is more than 10% above \( \gamma_U \) then the average power to tumor should be at least 30% above \( \gamma_L \).”
- \((\varepsilon_C > 0.1 \gamma_U) \Rightarrow (\sum_{ij \in T} D_{ij} \geq 1.3 N_T \gamma_L)\)
- \((A \Rightarrow B)\)
  \[ \varepsilon_C - M_1 \alpha_1 \leq 0.1 \gamma_U \]
  \[ \sum_{ij \in T} D_{ij} + M_2 \alpha_2 \geq 1.3 N_T \gamma_L \]
  \[ \alpha_1 + \alpha_2 \leq 1 \]

\[ \alpha_1, \alpha_2 \in \{0,1\} \]

... for suitable big-M values of \( M_1 \) and \( M_2 \)
4. Modeling conditional constraints

- \(f_1(x) > b_1 \Rightarrow f_2(x) \leq b_2\)
- \((A \Rightarrow B) \equiv (\neg A \vee B)\)

Because of this, can model as alternate constraints.

At least one of:
- \(f_1(x) \leq b_1\)
- \(f_2(x) \leq b_2\)

5. Modeling n-fold constraints

- “Divide the critical region into 10 sections \((C_1,...,C_{10})\). The average power must be less than \(\gamma_U\) on 4 or more sections.”
- Let \(N_k\) = number of pixels in \(C_k\)
- \(\sum_{ij \in C_k} D_{ij} \leq N_k \gamma_U, \ \forall k \in \{1,...,10\}\)
- Solve as:
  \[
  \sum_{ij \in C_k} D_{ij} M_k \alpha_k \leq N_k \gamma_U, \ \forall k \in \{1,...,10\} \\
  \sum_k (1-\alpha_k) \geq 4 \\
  \alpha_k \in \{0,1\} \ \forall k \in \{1,...,10\}
  \]
  “at least four alpha's must be zero”
5. Modeling n-fold constraints

- Require **at least r of p** constraints:
  \[ f_k(x) \leq b_k \quad k\in\{1,\ldots,p\} \]
- Introduce **indicator variables** \[ \alpha_k\in\{0,1\} \]
- Solve as:
  \[
  f_k(x) - M_k\alpha_k \leq b_k \\
  \sum_{k=1}^{p} (1-\alpha_k) \geq r
  \]
  with \[ M_k \geq \max_x [f_k(x) - b_k] \]

Summary

- Integer programs are very flexible and can model many real-world problems.
- The key new component is “0/1” variables (and integer variables more generally).
- Lots of modeling tricks using indicator variables and “big M” constants.
- Next class: interactive modeling exercises!