Overview

Decomposition based approach.
Start with
- Easy constraints
- Complicating Constraints.

Put the complicating constraints into the objective and delete them from the constraints.

We will obtain a lower bound on the optimal solution for minimization problems.

In many situations, this bound is close to the optimal solution value.
An Example: Constrained Shortest Paths

Given: a network $G = (N,A)$
- $c_{ij}$: cost for arc $(i,j)$
- $t_{ij}$: traversal time for arc $(i,j)$

$$z^* = \text{Min} \quad \sum_{(i,j) \in A} c_{ij} x_{ij}$$

s. t. $$\sum_j x_{ij} - \sum_j x_{ji} = \begin{cases} 1 & \text{if } i = 1 \\ -1 & \text{if } i = n \\ 0 & \text{otherwise} \end{cases}$$
$$\sum_{(i,j) \in A} t_{ij} x_{ij} \leq T$$

Complicating constraint

$$x_{ij} = 0 \text{ or } 1 \quad \text{for all } (i,j) \in A$$
Find the shortest path from node 1 to node 6 with a transit time at most 10
Shortest path problems are easy.

Shortest path problems with transit time restrictions are NP-hard.

We say that constrained optimization problem Y is a relaxation of problem X if Y is obtained from X by eliminating one or more constraints.

We will “relax” the complicating constraint, and then use a “heuristic” of penalizing too much transit time. We will then connect it to the theory of Lagrangian relaxations.
Step 1. (A Lagrangian relaxation approach). Penalize violation of the constraint in the objective function.

\[ z(\lambda) = \min \sum_{(i,j) \in A} c_{ij} x_{ij} + \lambda \left( \sum_{(i,j) \in A} t_{ij} x_{ij} - T \right) \]

\[ \sum_j x_{ij} - \sum_j x_{ji} = \begin{cases} 1 & \text{if } i = s \\ -1 & \text{if } i = t \\ 0 & \text{otherwise} \end{cases} \]

\[ \sum_{(i,j) \in A} t_{ij} x_{ij} \leq T \]  

Complicating constraint

\[ x_{ij} = 0 \text{ or } 1 \text{ for all } (i,j) \in A \]

Note: \( z^*(\lambda) \leq z^* \ \forall \lambda \geq 0 \)
Step 2. Delete the complicating constraint(s) from the problem. The resulting problem is called the *Lagrangian relaxation*.

\[ L(\lambda) = \text{Min} \sum_{(i,j) \in A} \left( c_{ij} + \lambda t_{ij} \right) x_{ij} - \lambda T \]

\[ \sum_j x_{ij} - \sum_j x_{ji} = \begin{cases} 
 1 & \text{if } i = 1 \\
 -1 & \text{if } i = n \\
 0 & \text{otherwise} 
\end{cases} \]

\[ \sum_{(i,j) \in A} t_{ij} x_{ij} \leq T \]

Complicating constraint

\[ x_{ij} = 0 \text{ or } 1 \text{ for all } (i,j) \in A \]

**Note:** \( L(\lambda) \leq z(\lambda) \leq z^* \) \( \forall \lambda \geq 0 \)
What is the effect of varying $\lambda$?

Case 1: $\lambda = 0$

$P = \begin{bmatrix}
1 & 1 & 10 \\
1 & 2 & 12 \\
5 & 10 & 1 \\
10 & 2 & 2
\end{bmatrix}$

c($P$) = 3

t($P$) = 18

$P = (c_{ij} + \lambda t_{ij})$
If $\lambda = 0$, the min cost path is found.

What happens to the (real) cost of the path as $\lambda$ increases from 0?

What path is determined as $\lambda$ gets VERY large?

What happens to the (real) transit time of the path as $\lambda$ increases from 0?
Let $\lambda = 1$

Case 2: $\lambda = 1$

$P = 1 - 2 - 5 - 6$

$c(P) = 5$

$t(P) = 15$
Let $\lambda = 2$

Case 3: $\lambda = 2$

Graph with nodes and edges labeled with weights and costs.

$P =$

$c(P) =$ $

$t(P) =$ / $

P =

\[ c(P) = \]

\[ t(P) = / \]
And alternative shortest path when $\lambda = 2$

$P = 1 - 3 - 5 - 6$

$c(P) = 15$
$t(P) = 10$

$1, 10$
$2, 3$
$10, 1$
$2, 2$

$1, 1$
$1, 2$
$5, 7$
$12, 3$
Let $\lambda = 5$

**Case 4: $\lambda = 5$**

$P = 1 - 3 - 2 - 4 - 5 - 6$

$c(P) = 24$

$t(P) = 8$
A parametric analysis

<table>
<thead>
<tr>
<th>Toll</th>
<th>modified cost</th>
<th>Cost</th>
<th>Transit Time</th>
<th>Modified cost -10λ</th>
</tr>
</thead>
<tbody>
<tr>
<td>$0 \leq \lambda \leq \frac{2}{3}$</td>
<td>$3 + 18\lambda$</td>
<td>3</td>
<td>18</td>
<td>$3 + 8\lambda$</td>
</tr>
<tr>
<td>$\frac{2}{3} \leq \lambda \leq 2$</td>
<td>$5 + 15\lambda$</td>
<td>5</td>
<td>15</td>
<td>$5 + 3\lambda$</td>
</tr>
<tr>
<td>$2 \leq \lambda \leq 4.5$</td>
<td>$15 + 10\lambda$</td>
<td>15</td>
<td>10</td>
<td>15</td>
</tr>
<tr>
<td>$4.5 \leq \lambda &lt; \infty$</td>
<td>$24 + 8\lambda$</td>
<td>24</td>
<td>8</td>
<td>$24 - 2\lambda$</td>
</tr>
</tbody>
</table>

The best value of $\lambda$ is the one that maximizes the lower bound.
Costs
Modified Cost – 10\(\lambda\)
Transit Times

Modified cost
The Lagrangian Multiplier Problem

\[ L(\lambda) = \min \sum_{(i,j) \in A} (c_{ij} + \lambda t_{ij}) x_{ij} - \lambda T \]

s.t. \[ \sum_j x_{ij} - \sum_j x_{ji} = \begin{cases} 1 & \text{if } i = 1 \\ -1 & \text{if } i = n \\ 0 & \text{otherwise} \end{cases} \]

\[ x_{ij} = 0 \text{ or } 1 \quad \text{for all } (i,j) \in A \]

\[ L^* = \max \{ L(\lambda) : \lambda \geq 0 \}. \quad \text{Lagrangian Multiplier Problem} \]

Theorem. \[ L(\lambda) \leq L^* \leq z^*. \]
Application to constrained shortest path

\[ L(\lambda) = \min \sum_{(i, j) \in A} (c_{ij} + \lambda t_{ij})x_{ij} - \lambda T \]

Let \( c(P) \) be the cost of path \( P \) that satisfies the transit time constraint.

**Corollary.** For all \( \lambda \), \( L(\lambda) \leq L^* \leq z^* \leq c(P) \).

If \( L(\lambda’) = c(P) \), then \( L(\lambda’) = L^* = z^* = c(P) \). In this case, \( P \) is an optimal path and \( \lambda’ \) optimizes the Lagrangian Multiplier Problem.