Gavin Hudnall
Michail Liaroutsos
Kaythan Somavarapu

Homework 2 - Report

To solve this problem, we first defined two binary variables as follows:

$x_{ij} = \{1, if\ truck\ travels\ from\ node\ i\ to\ j\ \ 0, otherwise\ ,\qquad \forall i = \{0, \dots, 9\}$

$y_i = \{1, if\ truck\ visits\ node\ i\ \ 0, otherwise\ ,\qquad \forall i = \{0, \dots, 9\}$

Our model uses the above decision variables and the predefined parameters for profit (p_i), capacity (k_i) and time between points (t_{ij}).

Model:

Objective Function: max $\sum_{i=1}{}^{n} p_i * y_i$

S.t.

$\sum_{i=0}{}^{n} k_i * y_i \leq C$ Total capacity must not exceed C

$\sum_{i=0}{}^{n}\ \sum_{j=1}\ {}^{n} x_{ij} * t_{ij} \leq M$ Total length of trip must not exceed M

$\sum_{i=0}{}^{n} X_{ij} \leq 1, \qquad \forall j = \{0, \dots, N\}$ For each node entered (j), there should be at most 1 leaving node (i)

$\sum_{j=0}{}^{n} X_{ij} \leq 1, \qquad \forall i = \{0, \dots, N\}$ For each node left (i), there should be at most 1 entering node (j)

$\sum_{j=0}{}^{n} X_{0j} = 1$ Starting node i = 0 must be left once.

$\sum_{i=1}{}^{n} X_{i0} = 1$ Ending node i = 0 must be entered once.

$\sum_{i=0}{}^{n} x_{ij} = y_i, \qquad \forall j = \{0, \dots, N\}$ In order to leave node i on path ij, you must have visited node i

$\sum_{j=0}{}^{n} x_{ij} = y_j, \qquad \forall j = \{0, \dots, N\}$ In order to enter node j on path ij, you must visit node j$X_{ii} =$

$0, \qquad \forall i = \{0, \dots, 9\}$ Truck cannot travel from node i directly back to node i

$\sum_{j=0}{}^{n} x_{i,j} + x_{j,i} \leq 1 \qquad \forall i = \{0, \dots, 9\}$ A path ij that has already been taken cannot be retraced from j to i

Subtour Elimination

Our initial code returned the subtours 0-6-1-0 and 4-5-4. To eliminate these subtours, we added the following constraints:

x01 + x16 + x60 <= 2

x10 + x61 + x06 <= 2

x60 + x06 <= 1

x54 + x45 <= 1

in order to ensure that each edge could only be traveled once (because xij and xji denote the same edge), and to keep the tour from making a full cycle of 0-6-1-0.

Gavin Hudnall
Michail Liaroutsos
Kaythan Somavarapu
After adding these constraints, we had another subtour 6-7-6. To eliminate this subtour, we added the constraint:

$$x67+x76 <= 1.$$

Results

After running the optimization model and implementing our subtour constraints, the optimal path returned by our program was:

0 - 4 - 5 - 2 - 0

With an objective value (profit) of 20.