IE 400 Principles of Engineering Management
Project Report
43

Cahit Ediz Civan - 22003206
Efe Kaan Fidancı - 22102589
Görkem Kadir Solun - 22003214

Question 1: Optimal City Selection for Tan-Tech's Promotional Tour

A tech startup, Tan-Tech, is planning a promotional tour to launch their latest product across multiple cities. The goal is to maximize exposure and attract both potential investors and customers, but their budget and time are limited. Each city has a different level of market potential, represented as benefit points. However, visiting every city is not feasible due to budget and time constraints.

Travel Costs Between Cities:

	A	B	C	D	E	F	G	H	I	J
A	0	300	450	150	225	330	495	600	315	405
B	300	0	180	330	315	375	510	195	420	240
C	450	180	0	210	495	135	240	435	180	345
D	150	330	210	0	360	195	300	405	270	285
E	225	315	495	360	0	240	465	180	285	225
F	330	375	135	195	240	0	270	300	345	390
G	495	510	240	300	465	270	0	135	255	210
H	600	195	435	405	180	300	135	0	285	315
I	315	420	180	270	285	345	255	285	0	150
J	405	240	345	285	225	390	210	315	150	0

Benefit Points of cities:

City	Benefit Points
A	350
B	420
C	270
D	300
E	380
F	410
G	320
H	450
I	330
J	400

To ensure the tour is impactful, Tan-Tech's objectives and constraints are as follows:

- The total travel cost for the tour cannot exceed $1,500.
- Due to logistical and team limitations, Tan-Tech can visit a maximum of 7 cities in total.
- The tour must start and end in city A.

Using Gurobi or a similar optimization software, formulate and solve the problem to generate an optimal route.

1.1. Decision Variables

- $x_{ij} \in \{0, 1\}$: binary variable that determines if the tour travels from city i to city j.
  - $x_i = 1$: if the tour travels from city i to city j.
  - $x_i = 0$: if the tour does not travel from city i to city j.
- $y_i \in \{0, 1\}$: binary variable that determines if city i is visited.
  - $y_i = 1$: city i is visited.
  - $y_i = 0$: city i is not visited.
- u_i: Auxiliary continuous variable used to eliminate subtours. Represents the order of visiting each city.

1.2. Parameters

- b_i is the benefit points associated with city i.
- c_{ij} is the travel cost between city i and city j.
- $n = 10$ is the total number of the cities.

1.3. Constraints

- **Budget constraint:** The total travel cost cannot exceed $1,500.

$$\sum_i \sum_j c_{ij} x_{ij} \leq 1500$$

- **Start and end in city A:** The tour starts and ends at city A.

$$\sum_j x_{Aj} = 1 \; and \; \sum_i x_{iA} = 1$$

- **Cannot visit the a city from itself**

$$x_{ii} = 0, \forall i$$

- **City visit limit:** The total number of cities visited is at most 7.

$$\sum_i y_i \leq 7$$

- **Inbound and outbound conservation:** For each city i, ensure that if a city is visited, there is an inbound and outbound connection.

$$\sum_j x_{ij} = y_i \;, \forall i \; and \; \sum_i x_{ij} = y_j \,, \forall j$$

- **Subtour elimination:** To avoid subtours, use the Miller-Tucker-Zemlin (MTZ) formulation.

$$u_i - u_j + (n-1)x_{ij} \leq n-2 \qquad \forall\, i, j \neq A,\ i \neq j$$

1.4. Objective Function

Maximize the total benefit points collected: $Max \sum_i b_i y_i$

1.5. Comments

The problem mirrors real world challenges companies face when planning promotional campaigns with limited resources. It integrates business goals to maximize exposure with logistical constraints (budget, time, and maximum cities). The requirement to start and end the tour in city A introduces a classic Traveling Salesperson Problem element, which is computationally challenging yet fascinating to solve. Balancing benefit points against travel costs creates a trade-off scenario, prompting strategic city selections to maximize impact. Only visiting up to 7 cities adds a layer of complexity. It requires formulating a binary integer program to make discrete decisions.

1.6 Solution

Path: A → B → J → I → G → H → E → A
Total benefit points: 2650.0
Total travel cost: 1485.0

Question 2: Optimal Meal Planning for Hiking

Item	Weight (kg)	Calories	Cost ($)
Granola Bars	2	300	5
Trail Mix	1	800	10
Dried Fruit	2	200	4
Canned Beans	6	800	7
Rice	4	1100	8
Energy Drink	6	150	3
Pasta	5	1200	9
Jerky	2	500	6

2.1. Decision Variables

- $x_i \in \{0, 1\}$: binary variable that determines if item i is assigned to Ece's bag.
    - $x_i = 1$: item i is in Ece's bag.
    - $x_i = 0$: item i is NOT in Ece's bag.
- $y_i \in \{0, 1\}$: binary variable that determines if item i is assigned to Arda's bag.
    - $y_i = 1$: item i is in Arda's bag.
    - $y_i = 0$: item i is NOT in Arda's bag.

2.2. Parameters

- Weights ($weight_i$): The weight of each item i (in kilograms).
    - Granola Bars ($i = 1$): 2 kg,
    - Trail Mix ($i = 2$): 1 kg,
    - Dried Fruit ($i = 3$): 2 kg,
    - Canned Beans ($i = 4$): 6 kg,
    - Rice ($i = 5$): 4 kg,
    - Energy Drink ($i = 6$): 6 kg,
    - Pasta ($i = 7$): 5 kg,
    - Jerky ($i = 8$): 2 kg,

- Calories ($calories_i$): The calorie value provided by each item i.
    - Granola Bars ($i = 1$): 300 cal,
    - Trail Mix ($i = 2$): 800 cal,
    - Dried Fruit ($i = 3$): 200 cal,
    - Canned Beans ($i = 4$): 800 cal,
    - Rice ($i = 5$): 1100 cal,

- Energy Drink ($i = 6$): 150 cal,
- Pasta ($i = 7$): 1200 cal,
- Jerky ($i = 8$): 500 cal.

- Costs ($cost_i$): The cost of each item i in dollars.
  - Granola Bars ($i = 1$): $5,
  - Trail Mix ($i = 2$): $10,
  - Dried Fruit ($i = 3$): $4,
  - Canned Beans ($i = 4$): $7,
  - Rice ($i = 5$): $8,
  - Energy Drink ($i = 6$): $3,
  - Pasta ($i = 7$): $9,
  - Jerky ($i = 8$): $6,

- Ece's Capacity (W_{Ece}): Maximum weight that Ece can carry: 10 kg.
- Arda's Capacity (W_{Arda}): Maximum weight that Arda can carry: 8 kg.
- Total Calorie Requirement (C_{Total}): Combined calorie needs of Ece and Arda: 2000 + 1500 = 3500 cal.

2.3. Constraints

- **Capacity (Weight) Constraint for Ece's Bag:** The following ensures that the total weight of items assigned to Ece's bag does not exceed 10 kg.

$$\sum_{i=1}^{8} weight_i \cdot x_i \leq W_{Ece}\,(10\,kg)$$

- **Capacity (Weight) Constraint for Arda's Bag:** The following ensures that the total weight of items assigned to Arda's bag does not exceed 10 kg.

$$\sum_{i=1}^{8} weight_i \cdot y_i \leq W_{Arda}\,(8\,kg)$$

- **Total Calorie Constraint:** The following ensures the combined calorie contribution of all selected items meets or exceeds 3500 cal.

$$\sum_{i=1}^{8} calories_i \cdot (x_i + y_i) \geq C_{Total}\,(3500\,cal)$$

- **Item Assignment Consistency:** The following ensures that if an item i is selected ($x_i = 1$), it must be assigned to exactly one of the two bags (either Ece's or Arda's). The item can either be in Ece's bag ($y_i = 1$) or Arda's bag ($z_i = 1$), but not both.

$$x_i + y_i \leq 1, \ \forall i$$

- **Binary Variables:** The following enforces that each of the variables is binary.

$$x_i, y_i \in \{0, 1\}, \ \forall i$$

2.4. Objective Function

The total cost (in dollars) is calculated by the following formula.

$$Total\ Cost\ = \sum_{i=1}^{8} cost_i \cdot (x_i + y_i)$$

The objective of the problem is to minimize the total cost of the selected items. Hence, the objective function is defined as:

$$Minimize\ \sum_{i=1}^{8} cost_i \cdot (x_i + y_i)$$

2.5. Comments

Important Aspects of the Problem

There are two important aspects in the problem that significantly affect the formulation of the constraints:

- **“They only have one item from each type of food.”**

This aspect implies that each food item can only be used once in the meal plan. If an item is selected, it must be included fully in either Ece's or Arda's backpack, but not both. This restriction is enforced through the binary decision variables x_i, and y_i.

- **“They can share the items in their backpacks once they reach their destination.”**

This means that the calorie intake requirement applies to both Ece and Arda together, not separately. The total calories from the selected food items must meet the combined calorie requirement of 3500 calories, but how the items are divided between the two backpacks does not matter for calorie intake, as they can share the items. The calorie constraint ensures that the total calories from the selected food items are enough to satisfy both Ece’s and Arda’s combined needs. Each selected item contributes to the total calorie intake regardless of which backpack it is placed in.

2.6. Comparison Between Gurobi and Custom Branch-and-Bound Algorithm

When comparing the performance of our custom branch-and-bound (BnB) algorithm with Gurobi for solving the binary integer programming problem in a meal planning scenario, we noticed a slight difference in execution times. Gurobi completed the optimization in approximately 0.007 seconds, while our BnB algorithm took around 0.017 seconds. Although the ratio between them fluctuated, Gurobi consistently executed the code slightly faster than our custom BnB algorithm. This disparity can be attributed to several key factors. Firstly, Gurobi is a highly sophisticated, commercially developed solver that leverages multiple advanced solving strategies which allow it to efficiently navigate the search space more effectively.

In contrast, our custom BnB algorithm, while optimized, lacks these advanced features and relies on a more straightforward approach to branching and bounding. It is also worth noting that although both solutions identified the same set of items to include in the meal plan, the distribution of these items between Ece's and Arda's backpacks differed. This highlights that while both methods achieve the same optimal solution in terms of item selection, the custom BnB algorithm may not be as efficient as Gurobi in determining the secondary aspects of the solution, such as the specific allocation of items to each backpack.

2.7. Solution to The Problem

Gurobi Solution:
- Ece’s bag: Rice, Pasta → (9 kg)
- Arda’s bag: Canned Beans, Jerky → (8 kg)
- Total Cost: $30

Custom BnB Solution:
- Ece’s bag: Canned Beans, Rice → (10 kg)
- Arda’s bag: Pasta, Jerky → (7 kg)
- Total Cost: $30