Optimization I

Janne Kettunen

1

1

Course Evaluation

- Please take time to provide course evaluation
- Evaluation available 12 Oct – 20 Oct
- I value and use your responses
- If you have suggestions how to improve the course or even if you are happy how things have been taught, I would appreciate to hear that!
- Incentive: if at least 70% of students provide feedback I will grant everybody 15 additional points

Thanks!

2

2

Feedback Survey Responses: 14 Oct

3

3

Preliminaries: Optimization in Networks

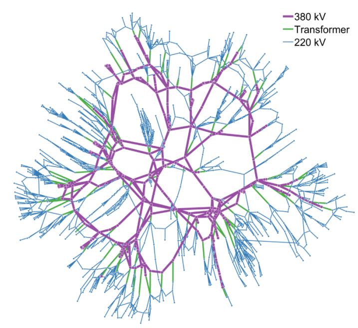
Janne Kettunen

4

4

Networks in Real Life

- Transportation of physical goods
- Communication networks
- Electricity and water networks
- Road/railroad/airline networks
- Social networks
- Biological networks

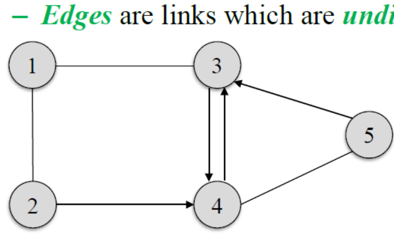


5

5

Terminology

- ***Graphs*** are used to model *travel*, *flow* and *adjacency patterns*.
- The ***nodes*** or ***vertices*** of a graph represent *entities*, *intersections* and *points of transfer* for the graph.
- Nodes are connected by links representing ***flow*** or *movement*:
  - ***Arcs*** are links which are ***directed*** ("one-way")
  - ***Edges*** are links which are ***undirected*** ("two-way")



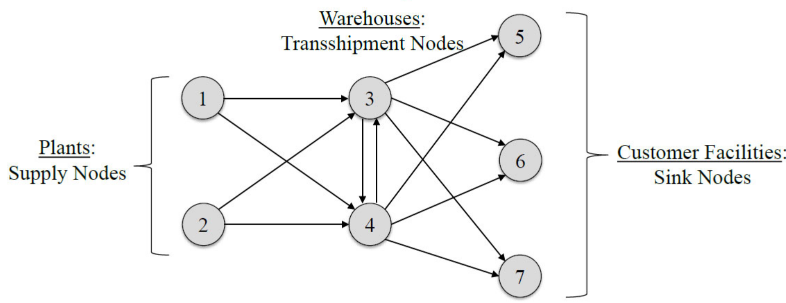
  - For an arc, the direction of flow matters; e.g., arc (3, 4) is not the same as arc (4, 3).
  - For an edge, such as (1, 3) the relationship between the two nodes is *independent* of direction.
- ***Directed graphs***, or ***digraphs***, are graphs that consist of arcs *only*. 6

6

Network Models

- ***Network flow models*** are defined on directed graphs.
- The nodes are of three types:
  - ***Source*** or ***supply*** nodes that ***create flow***,
  - ***Sink*** or ***demand*** nodes that ***consume flow***, and
  - ***Transshipment*** nodes that ***pass along flow***.

A Product Distribution Example



7

7

Formulating Network Flow Problems

Janne Kettunen

8

8

General Network Flow Model Formulation

Decision variables:
x_{ij}= flow from node i to node j

Constants:
c_{ij}= unit cost of flow for arc (i, j)
u_{ij}= flow capacity for arc (i, j)
b_k= total net demand for node k
(positive for sink, negative for source, zero for transshipment nodes)

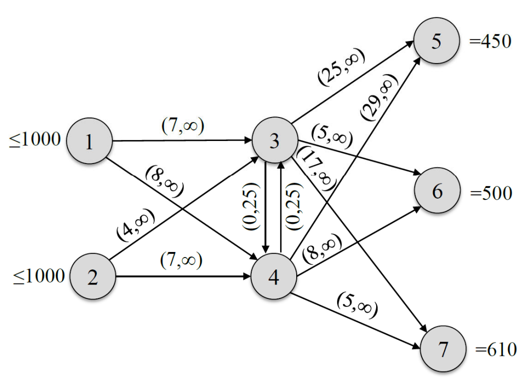
Sets and indices:
V= set of nodes or vertices in the network
A = set of arcs in the network
i, j, k = indices for nodes
(i, j) = arc from node i to node j

Minimize the overall cost of the flow on all the arcs
$$\min \sum_{(i,j)\in A} c_{ij} x_{ij}$$

Flow balance constraints
$$\text{s.t.} \sum_{(i,k)\in A} x_{ik} - \sum_{(k,i)\in A} x_{ki} = b_k, \text{ for all } k \in V$$

Bounds on the flow
$$0 \le x_{ij} \le u_{ij}, \text{ for all } (i, j) \in A$$

A Network Flow Model Example 1/2

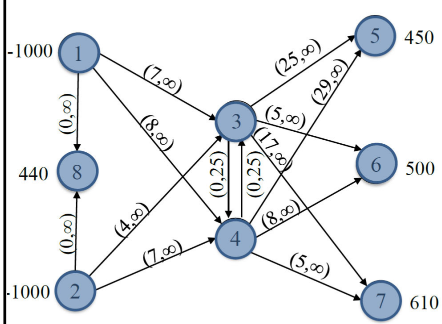


- Unit costs and capacities are indicated next to each arc (i, j) as (c_{ij}, u_{ij}).
- Source nodes have their associated maximum supply amounts indicated next to them.
- Sink nodes have their associated demand amounts indicated next to them.

- Note that supply capacity exceeds demand $(2000 > 450 + 500 + 610)$
- To deal with this, we add a dummy node and let excess demand to flow in it

A Network Flow Model Example 2/2

Network Representation ↔ ***LP Formulation***



"Network Flow.ipynb"

$$\min 7x_{13}+8x_{14}+4x_{23}+7x_{24}+25x_{35}+5x_{36}+17x_{37}+29x_{45}+8x_{46}+5x_{47}$$

Minimize cost

$$-x_{13}-x_{14}-x_{18}=-1000 \quad \text{(Node 1)}$$
$$-x_{23}-x_{24}-x_{28}=-1000 \quad \text{(Node 2)}$$
$$+x_{13}+x_{23}+x_{43}-x_{34}-x_{35}-x_{36}-x_{37}=0 \quad \text{(Node 3)}$$
$$+x_{14}+x_{24}+x_{34}-x_{43}-x_{45}-x_{46}-x_{47}=0 \quad \text{(Node 4)}$$
$$+x_{35}+x_{45}=450 \quad \text{(Node 5)}$$
$$+x_{36}+x_{46}=500 \quad \text{(Node 6)}$$
$$+x_{37}+x_{47}=610 \quad \text{(Node 7)}$$
$$+x_{18}+x_{28}=440 \quad \text{(Node 8)}$$

Flow balance constraints

$$x_{34}\le 25, x_{43}\le 25$$
$$x_{ij}\ge 0, \forall (i,j)\in A$$

Bounds on the flow

11

Solving Network Flow Models

- Most network flow models can be formulated and solved as LPs using the simplex algorithm (although they are inherently integer programming problems)
- Minimum network flow models can be solved as LPs when (i) supplies, (ii) demands, and (iii) capacities are all integer valued
  - This is a very powerful result as solving an LP problem is much faster than solving an IP problem

12

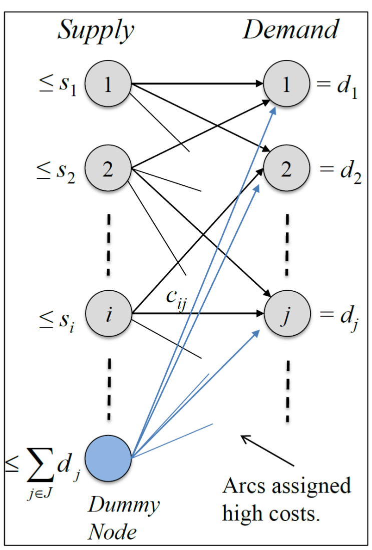
Network Problems

Janne Kettunen

13

13

Transportation Models



I = set of supply nodes i
J = set of demand nodes j
J_i = set of demand nodes linked for supply node i
s_i = supply at node i
d_j = demand at node j

$$\min \sum_{i \in I} \sum_{j \in J_i} c_{ij} x_{ij}$$ minimize transportation cost

$$\text{s.t. } \sum_{j \in J_i} x_{ij} \le s_i, \text{ for all } i \in I$$ Supply limit for transported amount (also for dummy node)

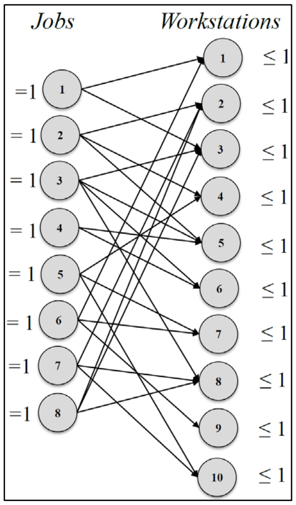
$$\sum_{i \in I_j} x_{ij} = d_j, \text{ for all } j \in J$$ demand is met

$$0 \le x_{ij} \le u_{ij} \text{ for all arcs } (i, j)$$ upper bounds for transportation

14

14

Assignment Models



I = set of jobs
J = set of workstations
A= set of allowed assignments (i, j)

$$\min \sum_{i\in I}\ \sum_{j\in J|(i,j)\in A} c_{ij}x_{ij} \quad \text{minimize total assignment cost}$$

$$\text{s.t.} \sum_{j\in J|(i,j)\in A} x_{ij} = 1, \text{ for all } i \in I \quad \text{every job } i \text{ is assigned to exactly one workstation } j$$

$$\sum_{i\in I|(i,j)\in A} x_{ij} \le 1, \text{ for all } j \in J \quad \text{every workstation } j \text{ is assigned to at most one job } i$$

$$x_{ij} = 0, \text{ or } 1 \text{ for each assignment } (i, j) \quad \text{assignment is integer valued}$$

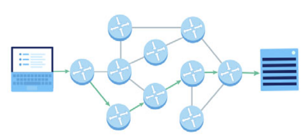
Because assignments are network flow problems, the binary requirement for the x_{ij} variables is unnecessary , i.e., x_{ij} will naturally be 0 or 1, if the binary requirement for each assignment (i,j) is replaced by $0 \le x_{ij} \le 1$.

15

15

Shortest Path

- Finding a shortest path from a node to another node is a problems faced in (i) vehicle routing, (ii) internet routing, (iii) designing facility layouts



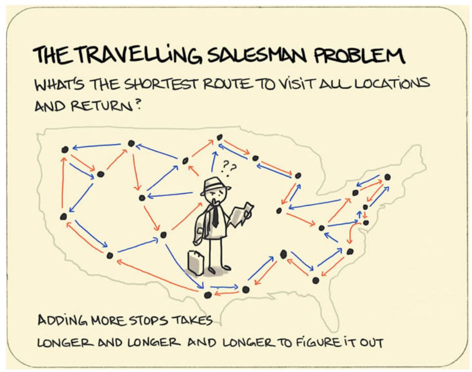
- This problem can be solved:
  - formulating it as a minimum cost flow network with a supply of 1 unit at the origin node and a demand of 1 unit at the destination mode
  - using Dijkstra's algorithm that iteratively finds the shortest path from the origin to the destination https://www.youtube.com/watch?v=pVfj6mxhdMw

16

16

Travelling Salesman Problem

- Problem: A salesman has to visit each one of N cities exactly once, and return back to the starting point. The goal is to minimize the total length (or cost) of the trip.



17

17

Formulating Travelling Salesman Problem

- Problem: A salesman has to visit each one of N cities exactly once, and return back to the starting point. The goal is to minimize the total length (or cost) of the trip. We assume

  c_{ij} = the cost to travel from node i to node j

  x_{ij} = 1 if the route goes from node i to node j, and 0 otherwise

$$min \sum_{i \in N} \sum_{\substack{j \in N \\ j \neq i}} c_{ij} x_{ij}$$ minimize the trip length

s.t.

$$\sum_{i \in N, i \neq j} x_{ij} = 1, \qquad j \in N$$

$$\sum_{j \in N, j \neq i} x_{ij} = 1, \qquad i \in N$$

enter and exit each city once

$$\sum_{i \in N} \sum_{j \in N} x_{ij} \leq |S| - 1, \qquad S \subset V, 2 \leq |S| \leq N-2$$

subtour elimination constraints (S is the set of all tours and V is the set of all vertices)

$$x_{ij} \in \{0,1\}, \qquad \forall i, j \in N$$

18

18

Travelling Salesman Problem Computational Times

PROBLEM SIZE (STOPS)	APPROXIMATE SOLUTION TIME
10	3 mili-seconds
17	4 days
20	77 years
25	490 million years
30	8.4 x 10^15 years
50	9.6 x 10^47 years

19

19

Travelling Salesman Problem: Miller-Tucker-Zemlin Formulation (https://co-enzyme.fr/blog/traveling-salesman-problem-tsp-in-cplex-opl-with-miller-tucker-zemlin-mtz-formulation/)

- Problem: A salesman has to visit each one of N cities exactly once, and return back to the starting point. The goal is to minimize the total length (or cost) of the trip. We assume

  c_{ij} = the cost to travel from node i to node j, $i, j = 1, \ldots, n$

  x_{ij} = 1 if the route goes from node i to node j, and 0 otherwise

  u_i = is a helper variable used in subtour elimination constraint

$$min \sum_{i \in N} \sum_{\substack{j \in N \\ j \neq i}} c_{ij} x_{ij}$$

minimize the cost of the trip

s.t.

$$\sum_{i \in N, i \neq j} x_{ij} = 1, \qquad j \in N$$

$$\sum_{j \in N, j \neq i} x_{ij} = 1, \qquad i \in N$$

enter and exit each city once

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

$$u_1 = 0$$

enforces that there is only a single connected tour rather than several disjointed tours that collectively cover the cities (u_i denotes the order in which a city is visited, we start at i=1)

$$x_{ij} \in \{0,1\}, \qquad \forall i, j \in N$$

$$u_i \in \mathbb{R}^+, \qquad i \in N$$

"tsm.ipynb"

20

20

Travelling Salesman Problem: Miller-Tucker-Zemlin Formulation

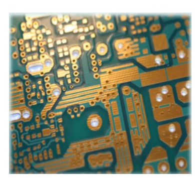
- Problem: A salesman has to visit each one of N cities exactly once, and return back to the starting point. The goal is to minimize the total length (or cost) of the trip

PROBLEM SIZE (STOPS)	APPROXIMATE SOLUTION TIME
10	3 mili-seconds
17	4 days
20	77 years
25	490 million years
30	8.4 x 10^15 years
50	9.6 x 10^47 years

```
Explored 1662 nodes (12183 simplex iterations) in 0.64 seconds
Thread count was 8 (of 8 available processors)

Solution count 6: 39 42 44 ... 80

Optimal solution found (tolerance 1.00e-04)
Best objective 3.900000000000e+01, best bound 3.900000000000e+01, gap 0.0000%

Value of objective function: 39

Decision variables:
route[1,12] = 1
route[2,14] = 1
route[3,1] = 1
route[4,6] = 1
route[5,4] = 1
route[6,7] = 1
route[7,16] = 1
route[8,17] = 1
route[9,5] = 1
route[10,11] = 1
route[11,13] = 1
route[12,8] = 1
route[13,2] = 1
route[14,3] = 1
route[15,10] = 1
route[16,15] = 1
route[17,9] = 1
```


Solving Travelling Salesman Problem

- Can be extremely hard
- It may not be solvable for larger problems, in which case heuristics are used
- Is present in several contexts, such as:
  - finding the optimal sequence to drill holes for micro chips
  - vehicle routing
  - job shop scheduling



Summary

- Many problems can be formulated as network problems:
  - Transportation models
  - Assignment models
  - Shortest path
  - Travelling salesman problem

- Often the network formulation is the most efficient way to solve the problem

23

23

Optimization in Networks Using Python and Gurobi

Janne Kettunen

24

24

Peloton's Multicommodity Flow Problem

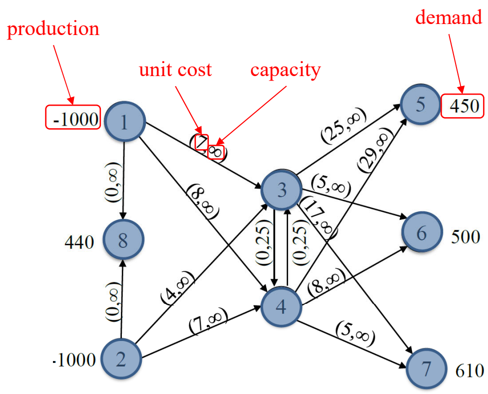
Peloton is producing both its (i) accessories and (ii) main sport equipment in Detroit and Denver for its customers in Boston, Washington D.C., and Seattle. In Detroit, Peloton produces 50 units of accessories and 60 units of equipment. In Denver, Peloton produces 60 units of accessories and 40 units of equipment. The demand for accessories is 50, 50, and 10 units in Boston, Washington D.C., and Seattle, respectively. The demand for equipment is 40, 30, and 30 units in Boston, Washington D.C., and Seattle, respectively. The transportation capacities and transportation unit costs between the cities are as follows:

		Boston		Washington D.C.		Seattle	
		unit cost	capacity	unit cost	capacity	unit cost	capacity
Detroit	accessories	10	100	20	80	60	120
	equipment	20		20		80	
Denver	accessories	40	120	40	120	30	120
	equipment	60		70		30	

25

Reminder: A Network

Network Representation $\leftrightarrow$ ***LP Formulation***



$$\min 7x_{13} + 8x_{14} + 4x_{23} + 7x_{24} + 25x_{35}$$
$$+5x_{36} + 17x_{37} + 29x_{45} + 8x_{46} + 5x_{47}$$
$$-x_{13} - x_{14} - x_{18} = -1000 \quad \text{(Node 1)}$$
$$-x_{23} - x_{24} - x_{28} = -1000 \quad \text{(Node 2)}$$
$$+x_{13} + x_{23} + x_{43} - x_{34} - x_{35} - x_{36} - x_{37} = 0 \quad \text{(Node 3)}$$
$$+x_{14} + x_{24} + x_{34} - x_{43} - x_{45} - x_{46} - x_{47} = 0 \quad \text{(Node 4)}$$
$$+x_{35} + x_{45} = 450 \quad \text{(Node 5)}$$
$$+x_{36} + x_{46} = 500 \quad \text{(Node 6)}$$
$$+x_{37} + x_{47} = 610 \quad \text{(Node 7)}$$
$$+x_{18} + x_{28} = 440 \quad \text{(Node 8)}$$
$$x_{34} \le 25, x_{43} \le 25$$
$$x_{ij} \ge 0, \forall (i,j) \in A$$

26

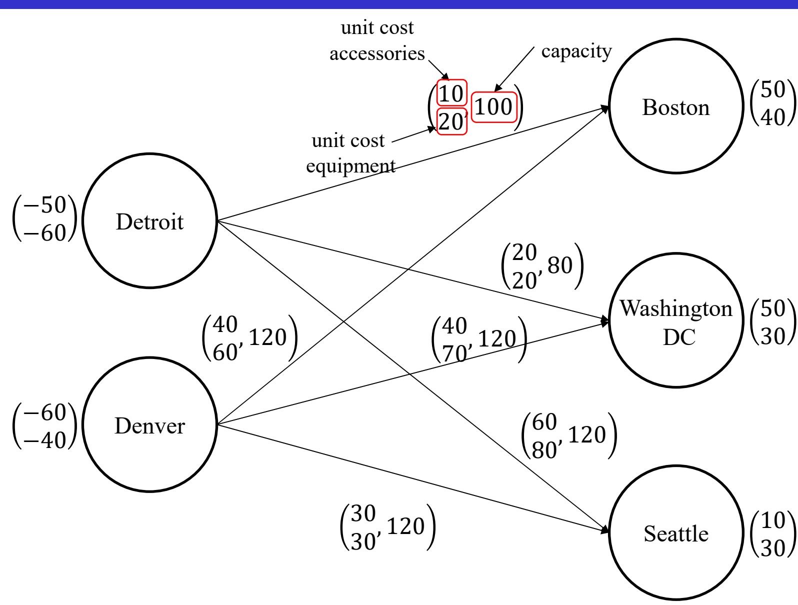
Task

- Draw a network for the Peloton’s problem. Note that there are now two categories of items to ship, namely equipment and accessories. Think how to represent these in the network figure.

27

27

Network



28

28

Peloton’s Network Flow Optimization

Decision variables:
x_{kij}= flow of commodity k from node i to node j

Sets and constants:
N = set of nodes in the network
A= set of arcs in the network
C= set of commodities
(i, j)= arc from node i to node j
c_{kij}= cost to transport commodity k from node i to node j
b_{kj}= total net demand of commodity k for node j
u_{ij}= flow capacity for arc (i, j)

$$min \sum_{k \in C} \sum_{(i,j) \in A} c_{kij} x_{kij}$$ — minimize transportation costs

$$\text{s.t.} \quad \sum_{(i,j) \in A} x_{kij} - \sum_{(j,i) \in A} x_{kji} = b_{kj} \quad \forall j \in N, k \in C$$ — flow balance constraints

$$0 \leq \sum_{k \in C} x_{kij} \leq u_{ij} \quad \forall (i,j) \in A$$ — bounds on flow

29

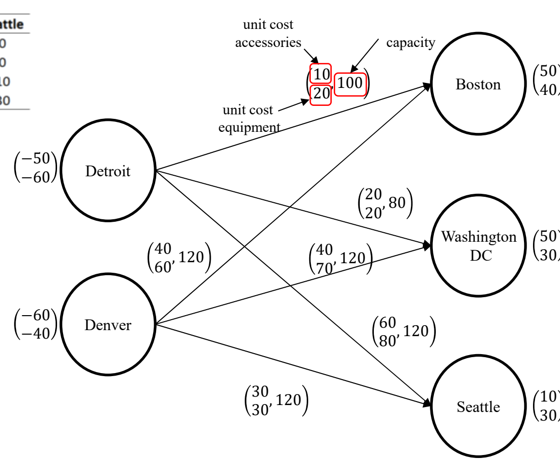
Solution

units transported

		Boston	Washington D.C.	Seattle
Detroit	accessories	50	0	0
	equipment	30	30	0
Denver	accessories	0	50	10
	equipment	10	0	30

Total transportation cost: 5,500

“multicom.ipynb”



- Does the solution make sense intuitively?
- What happens to the solution if all transportation capacities are reduces by 20%?

30

Final Exam

- On 21st Oct, Thursday 4:30pm – 7:00pm at Duques 258
- About 20 multiple choice questions and 5 short answer / calculation questions
- You can have a one A4 size notes page with your study notes. You are not permitted to use calculators, any other notes, books, computers, mobile phones during the exam.
- Preparation:
  - Go carefully over lecture notes so that you know the content very well.
  - Prepare the one A4 size note page carefully.
  - If you did not attend a certain session or did not have a chance to watch the recording yet, please do so.
  - Make sure you understand all the covered concepts thoroughly and also how to interpret figures and graphs that are outputs from software we used.
  - When taking the exam manage your time well, make sure to answer for all questions.

31

Interactive Review for Final Exam

- You will have about 3 min in small groups of 3-4 to identify 1-2 areas that you would like to be reviewed (list these areas based on session number and topic)
- After that we will list the topics to be reviewed and go over a few of these areas

32

32

Course Topics to Review

Session Number	Topic / Concept	Counts

33

Finally: Thank You!

- Thank you everyone for taking this class and working hard on the assignments
- All the best in your future endeavors!

34

34