Communications Network Design

lecture 11

Matthew Roughan
<matthew.roughan@adelaide.edu.au>

Discipline of Applied Mathematics
School of Mathematical Sciences
University of Adelaide

March 23, 2009

Multicommodity flow problems

In this section we consider a special case of the network design with linear separable costs, but note that this is still NP-hard, so we need a heursitic solution. The first we try is Minoux's greedy method.

Notation recap

Mostly as before

- A **network** is a graph $G(N,E)$, with **nodes** $N = \{1,2,\ldots n\}$ and **links** $E \subseteq N \times N$
- Offered traffic between O-D pair (p,q) is t_{pq}
- The set of all **paths** in $G(N,E)$ is $P = \cup_{[p,q]\in K} P_{pq}$
- Each link $e \in E$ has
  - a **capacity**, denoted by $r_e (\geq 0)$
  - a **distance** $d_e (\geq 0)$
  - a **load** $f_e (\geq 0)$
- The vector $\mathbf{x} = (x_\mu : \mu \in P)$ is called the **routing**

$$f_e = \sum_{\mu \in P : e \in \mu} x_\mu$$

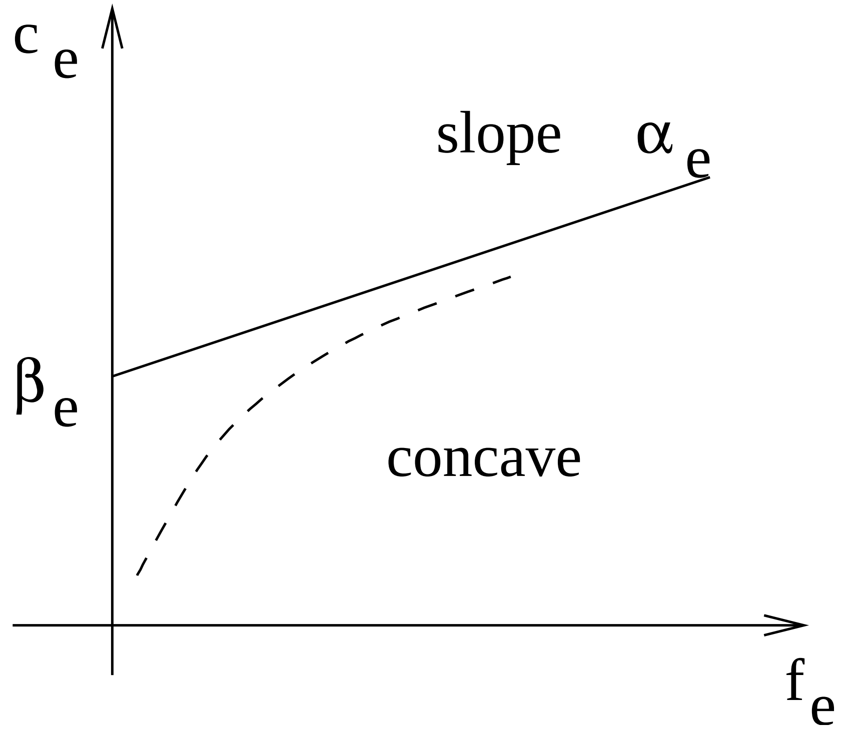
A simplified problem

- There are some interesting special cases of the minimum cost, multicommodity flow problem, which we now consider.
  - lets us start a little simpler
    - similar to earlier presentation
- choose capacities to carry required loads with overhead
  - $r_e = \gamma f_e$ for some $\gamma > 1$
- separable linear cost model (with two components)
  - a fixed cost for provision of the link β_e
  - a cost proportional to the capacity r_e (i.e. $\alpha_e f_e$)
  - distances come in through β_e and α_e

Separable linear cost model

$$c_e(f_e) = \begin{cases} 0 & \text{if } f_e = 0 \\ \beta_e + \alpha_e f_e & \text{if } f_e > 0 \end{cases}$$

Note that $C(\mathbf{f}) = \sum_{e: f_e > 0} (\beta_e + \alpha_e f_e)$ is concave:



Complete topology

For a given node set N, the completely connected topology has

$$|E| = \frac{|N|(|N|-1)}{2}$$

possible links and $2^{|E|}$ possible networks.

Only those links with $f_e > 0$ will be included in the final design, so put

$$L(\mathbf{f}) = \{e \in E : f_e > 0\}$$

$L(\mathbf{f})$ is the set of links used in the network design.

Problem formulation

Formal optimization problem

$$
\begin{aligned}
\text{(P)} \qquad \text{min.} \qquad C(\mathbf{f}) &= \sum_{e \in L(\mathbf{f})} (\beta_e + \alpha_e f_e) \\
\text{s.t.} \qquad f_e &= \sum_{\mu \in P: e \in \mu} x_\mu & \forall e \in E. \\
x_\mu &\geq 0 & \forall \mu \in P \\
\sum_{\mu \in P_k} x_\mu &= t_k & \forall k \in K
\end{aligned}
$$

where $\beta_e, \alpha_e, t_k, N$ are all givens, and the link capacities will be $r_e = \gamma f_e$.

An aside

Recall (from SPF routing) that

$$\begin{aligned}\sum_e \alpha_e f_e &= \sum_e \alpha_e \left(\sum_{\mu \in P : e \in \mu} x_\mu \right) \\ &= \sum_{\mu \in P : e \in \mu} \left(\sum_{e \in \mu} \alpha_e \right) x_\mu \\ &= \sum_{\mu \in P} l_\mu x_\mu\end{aligned}$$

where $l_\mu = \sum_{e \in \mu} \alpha_e$ is the length of path μ, so

$$C(\mathbf{f}) = \sum_{e \in L(\mathbf{f})} (\beta_e + \alpha_e f_e) = \sum_{e \in L(\mathbf{f})} \beta_e + \sum_{\mu \in P} l_\mu(L(\mathbf{f})) x_\mu$$

Simplification

For a given set of links L, we can solve this problem by routing the traffic t_{pq} on a shortest path in the network which has link set L, for all O-D pairs, $k \in K$. So

$$C(\mathbf{f}) = \sum_{k \in K} \hat{l}_k(L) t_k + \sum_{e \in L} \beta_e = v(L)$$

where $\hat{l}_k(L)$ represents the length of the shortest path for O-D pair k, in the network with link set L.

- cost of the network only depends on the choice of L
- becomes integer programming problem: choose which links to include or exclude
- always using SPF routing (linear cost is also convex)

Heuristic Methods

Problem we wish to solve is minimise $\{v(L) : L \subseteq E\}$
Decision variables

$$z_e = \begin{cases} 1 & \text{if link } e \in L \text{ (i.e. we use } e\text{)} \\ 0 & \text{if link } e \notin L \text{ (i.e. we don't use } e\text{)} \end{cases}$$

- difficult problem
  - each link can be in one of two states
  - there are $2^{|E|}$ possible choices for L
  - NP-hard (see travelling salesman problem)
- NP-hard $\Rightarrow$ heuristic methods
  - Minoux's greedy method [1]
  - branch and bound (next lectures)

Greedy Methods

heuristic = a rule of thumb (unprovable, but reasonable)

Greedy heuristic

- at each step we make the best choice
  - don't ever go back
- e.g. Dijkstra, Minoux's greedy method
- advantage
  - generally pretty simple
- disadvantage
  - doesn't reach true optimum in many cases
    - results are still sometimes quite good
  - Dijkstra does find an optimum

Minoux's Greedy Method

(a) **Initialise:** $k = 0$, $L^{(0)} = E$, and $\mathbf{f}^{(0)}$ is the initial load

(b) **For** each link $e = (i, j) \in L^{(k)}$ such that $f_e^{(k)} > 0$,

- determine $\hat{l}_{\mu_{ij}}(L - e)$, the length of the shortest path μ_{ij} from i to j, in the network with link e removed from L
- compute $\Delta_e = \hat{l}_{\mu_{ij}}(L - e)\, f_e^{(k)} - (\alpha_e f_e^{(k)} + \beta_e)$
  - Δ_e is the increase in cost of rerouting load on link e to the shortest path μ_{ij}, when link e is removed.
  - By convention, $\Delta_e = \infty$ if there is no path from p to q, for $e = (p, q)$.

Minoux's Greedy Method (cont)

(c) **If** there exists e such that $\Delta_e < 0$
we can improve the network. Let

$$\Delta_e = \min\{\Delta_g : \Delta_g < 0, g \in L^{(k)}\}, \quad L^{(k+1)} = L^{(k)} - \{e\}$$

For all $g \in L^{(k)}$,

$$f_g^{(k+1)} = \begin{cases} f_g^{(k)} & \text{if} \quad g \notin \mu_{ij}, g \neq e \\ f_g^{(k)} + f_e^{(k)} & \text{if} \quad g \in \mu_{ij} \\ 0 & \text{if} \quad g = e \end{cases}$$

$k \leftarrow k+1$. **Goto (b)**

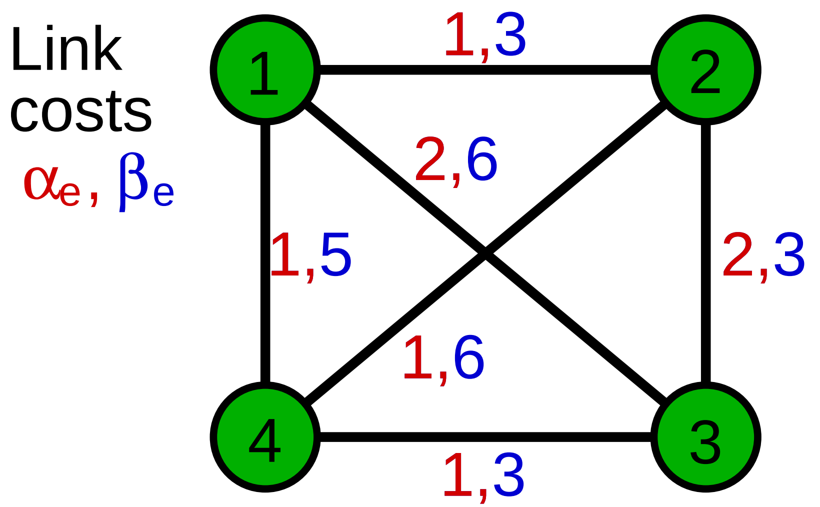
Else ($\Delta_e \geq 0$ for all $e \in L^{(k)}$) **STOP**

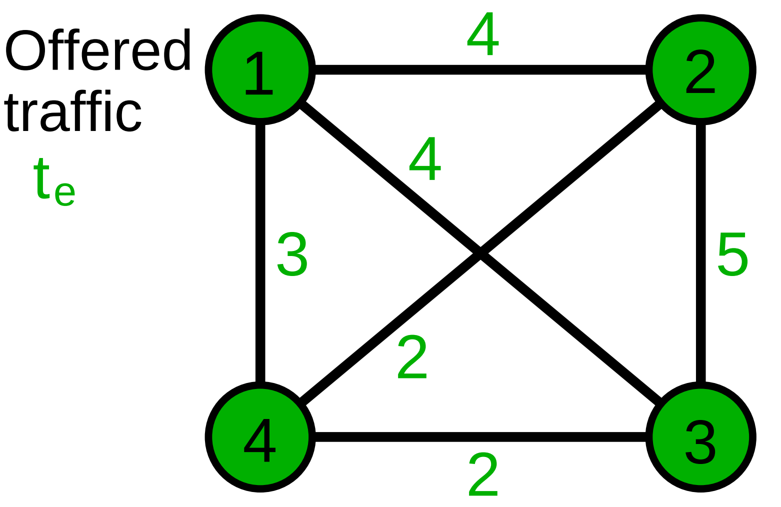
Minoux's Greedy Method

- When it finishes, the greedy solution has been found
  - cannot be bettered by this method.
  - might not be optimal
- Recall the proposition: Use only ONE path at (c), because costs are concave.
- Costs linear, so also convex, so shortest path routing is minimal (for a given network).

Minoux's Method: Example 1

The network $G(N,E)$ and data for the fixed charge model (α_e, β_e) and offered traffic, t_{pq}





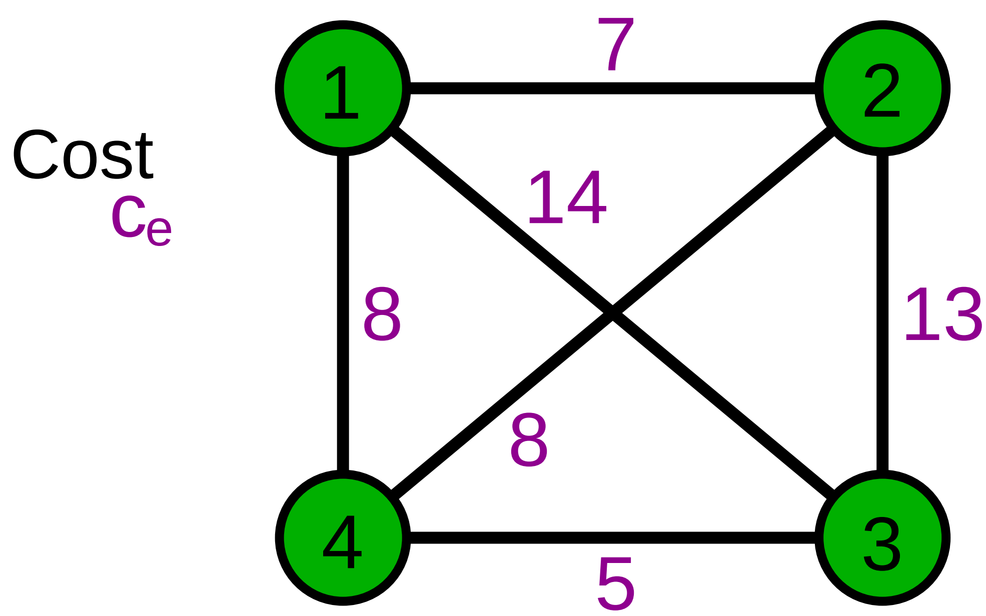
$$C(\mathbf{f}) = \sum_{e \in L} c_e(f_e)$$

$$c_e(f_e) = \alpha_e f_e + \beta_e.$$

Minoux's Method: Example 1

$C(\mathbf{f}) = \sum_{e \in L} c_e(f_e) = \sum_{e \in L} \alpha_e f_e + \beta_e$, where $L \subseteq E$

Assume initially direct routing i.e. $f_e = t_{pq}$ for all $e = (p,q)$, and $L^{(0)} = E$.



Total cost initially is 55 units.

Minoux's Method: Example 1

Iteration 1: Calculate all Δ_e

$$\begin{aligned} \Delta_e &= l_{\hat{\mu}}(\mathbf{f}) - (\alpha_e f_e + \beta_e) \\ &= \sum_{e' \in \hat{\mu}} \alpha_{e'} f_{e'} - \alpha_e f_e - \beta_e \end{aligned}$$

For example Δ_{12} is the change in cost, if link $(1,2)$ is removed, and f_{12} is rerouted onto the remaining shortest path, here 1-4-2.

$$\begin{aligned} \Delta_{12} &= (\alpha_{14} + \alpha_{42} - \alpha_{12}) f_{12} - \beta_{12} \\ &= (1 + 1 - 1) \times 4 - 3 \\ &= 1 \end{aligned}$$

Minoux's Method: Example 1

Iteration 1: Calculate all Δ_e

$$
\begin{aligned}
\Delta_{12} &= (\alpha_{14} + \alpha_{42} - \alpha_{12}) f_{12} - \beta_{12} = (1 + 1 - 1) \times 4 - 3 = 1 \\
\Delta_{13} &= (\alpha_{14} + \alpha_{34} - \alpha_{13}) f_{13} - \beta_{13} = (1 + 1 - 2) \times 4 - 6 = -6 \\
\Delta_{14} &= (\alpha_{12} + \alpha_{42} - \alpha_{14}) f_{14} - \beta_{14} = (1 + 1 - 1) \times 3 - 5 = -2 \\
\Delta_{23} &= (\alpha_{24} + \alpha_{34} - \alpha_{23}) f_{23} - \beta_{23} = (1 + 1 - 2) \times 5 - 3 = -3 \\
\Delta_{24} &= (\alpha_{12} + \alpha_{14} - \alpha_{24}) f_{24} - \beta_{24} = (1 + 1 - 1) \times 2 - 6 = -4 \\
\Delta_{34} &= (\alpha_{23} + \alpha_{24} - \alpha_{34}) f_{34} - \beta_{34} = (1 + 2 - 1) \times 2 - 3 = 1
\end{aligned}
$$

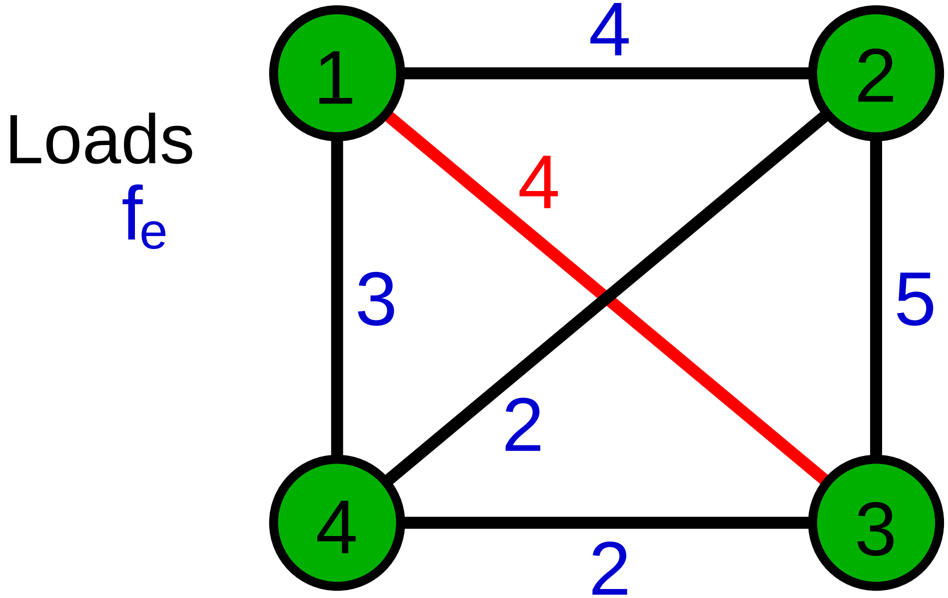
Therefore minΔ_e=-6, for $e = (1,3)$.

Minoux's Method: Example 1

Iteration 1: Remove link $(1,3)$ from the network, e.g. put $L^{(1)} = L^{(0)} \setminus \{(1,3)\}$

Reroute f_{13} onto the path 1-4-3.

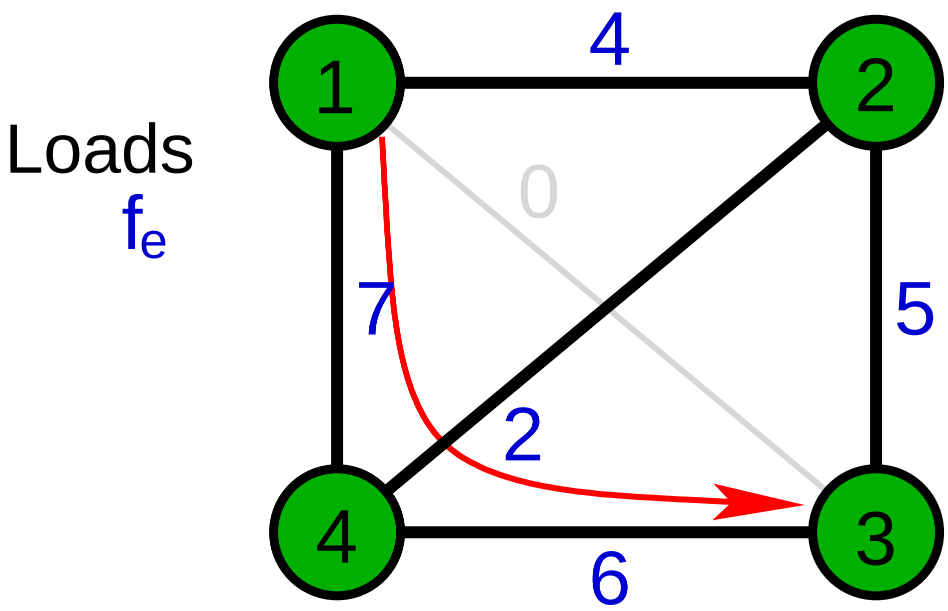
The new network and loads are:

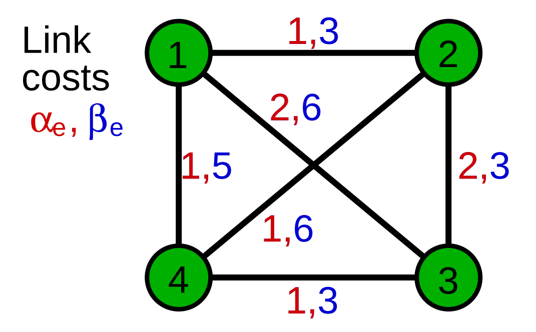


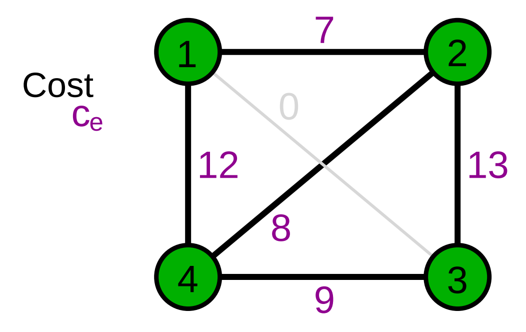
The new cost is old cost $+\Delta_{13}$=55-6=49 units.

Minoux's Method: Example 1

Iteration 1: Remove link $(1,3)$ from the network,
e.g. put $L^{(1)} = L^{(0)} \setminus \{(1,3)\}$
Reroute f_{13} onto the path 1-4-3.
The new network and loads are:







The new cost is old cost $+\Delta_{13}$=55-6=49 units.

Minoux's Method: Example 1

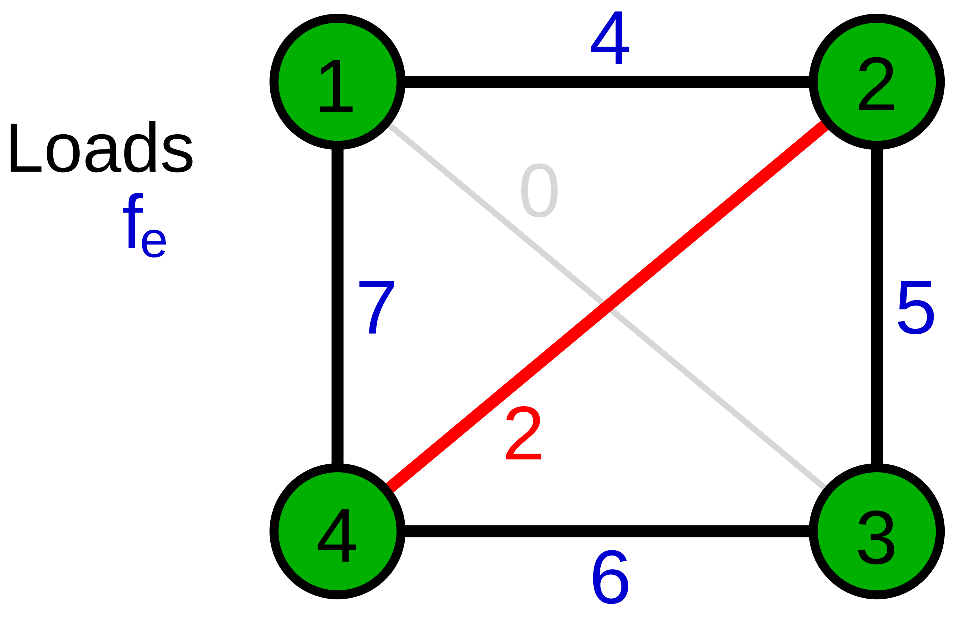
Iteration 2: Working with this latest network $L^{(1)}$, re-calculate all Δ_e

$$\begin{aligned}
\Delta_{12} &= (\alpha_{14}+\alpha_{42}-\alpha_{12})\,f_{12}-\beta_{12} = (1+1-1)\times 4-3 = 1 \\
\Delta_{14} &= (\alpha_{12}+\alpha_{42}-\alpha_{14})\,f_{14}-\beta_{13} = (1+1-1)\times 7-5 = 2 \\
\Delta_{23} &= (\alpha_{24}+\alpha_{34}-\alpha_{23})\,f_{23}-\beta_{23} = (1+1-2)\times 5-3 = -3 \\
\Delta_{24} &= (\alpha_{12}+\alpha_{14}-\alpha_{24})\,f_{24}-\beta_{24} = (1+1-1)\times 2-6 = -4 \\
\Delta_{34} &= (\alpha_{23}+\alpha_{24}-\alpha_{34})\,f_{34}-\beta_{34} = (1+2-1)\times 6-3 = 9
\end{aligned}$$

Therefore $\min \Delta_e = -4$, for $e = (2,4)$.

Minoux's Method: Example 1

Iteration 2: Put $L^{(2)} = L^{(1)} \setminus \{(2,4)\}$; reroute f_{24} onto the path 2-1-4.
The new network and loads are:

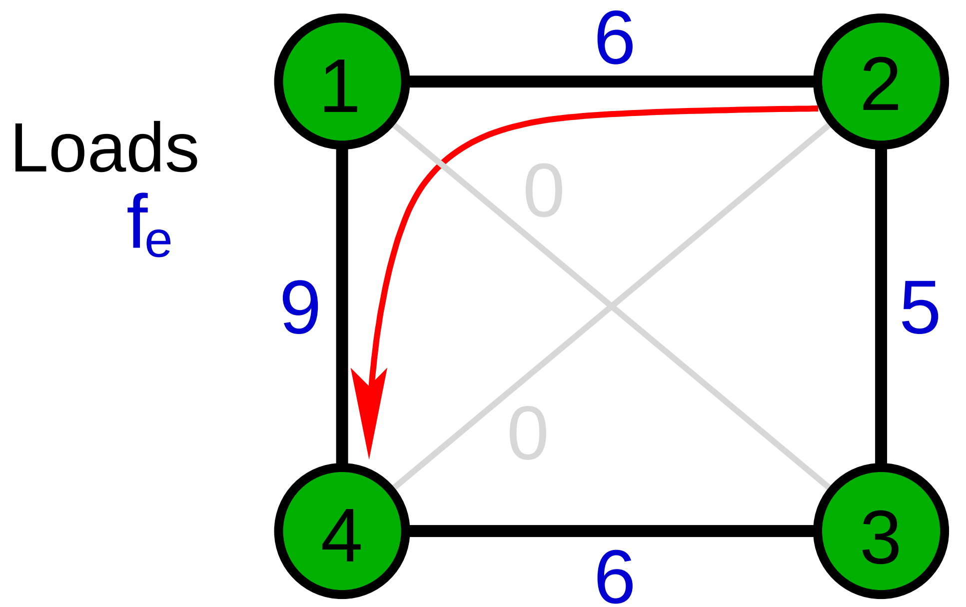


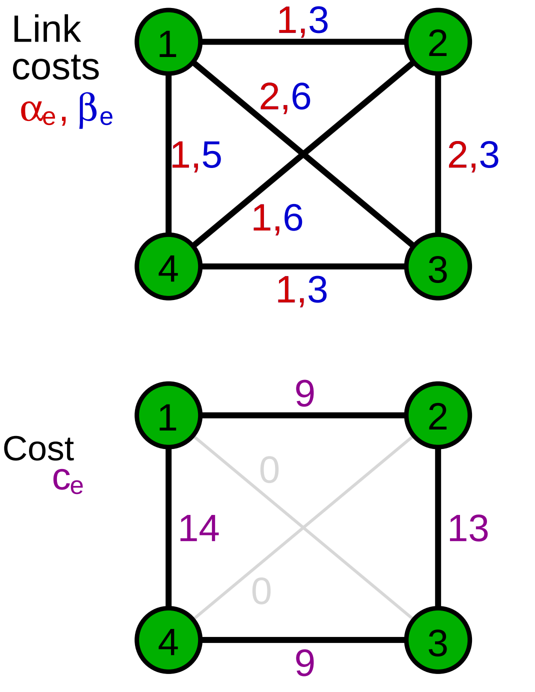
The new cost is $49 - 4 = 45$ units.

Minoux's Method: Example 1

Iteration 2: Put $L^{(2)} = L^{(1)} \setminus \{(2,4)\}$; reroute f_{24} onto the path 2-1-4.

The new network and loads are:





The new cost is $49 - 4 = 45$ units.

Minoux's Method: Example 1

Iteration 3: Working with this latest network $L^{(2)}$, re-calculate all Δ_e

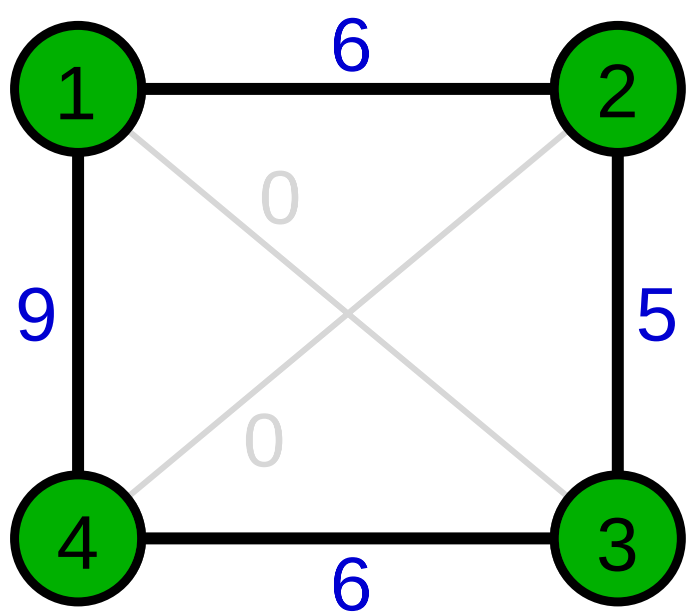
$$\begin{aligned}
\Delta_{12} &= (\alpha_{14}+\alpha_{34}+\alpha_{24}-\alpha_{12})f_{12}-\beta_{12} = (1+1+2-1)\times 6-3>0 \\
\Delta_{14} &= (\alpha_{12}+\alpha_{23}+\alpha_{34}-\alpha_{14})f_{14}-\beta_{13} = (1+2+1-1)\times 9-5>0 \\
\Delta_{23} &= (\alpha_{21}+\alpha_{14}+\alpha_{34}-\alpha_{23})f_{23}-\beta_{23} = (1+1+1-2)\times 5-3>0 \\
\Delta_{34} &= (\alpha_{14}+\alpha_{12}+\alpha_{23}-\alpha_{34})f_{34}-\beta_{34} = (1+1+2-1)\times 6-3>0
\end{aligned}$$

Therefore $\Delta_e > 0, \forall e \in L^{(2)}$ so STOP.

Minoux's Method: Example 1

So the final network design and loads are (as in interation 2):

Loads f_e



O-D	t_{pq}	routing
1-2	4	1-2
1-3	4	1-4-3
1-4	3	1-4
2-3	5	2-3
2-4	2	2-1-4
3-4	2	3-4

The cost is still 45 units.

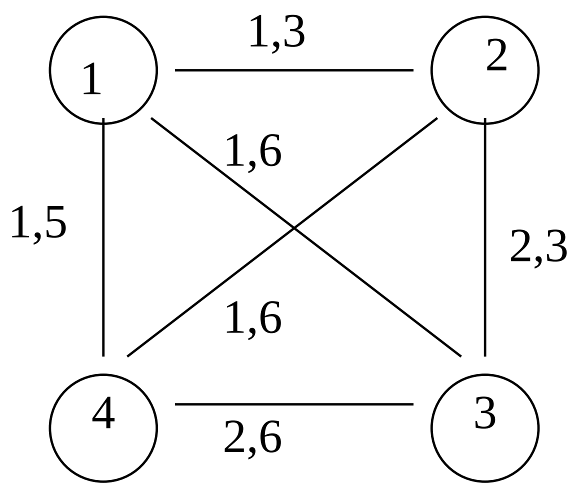
Minoux's Method: Example 1

This is actually the optimal design for the network with the given data, but obviously the method itself has a flaw in that once a link is deleted, it is deleted for good: there is never a chance for it to be reinstated.

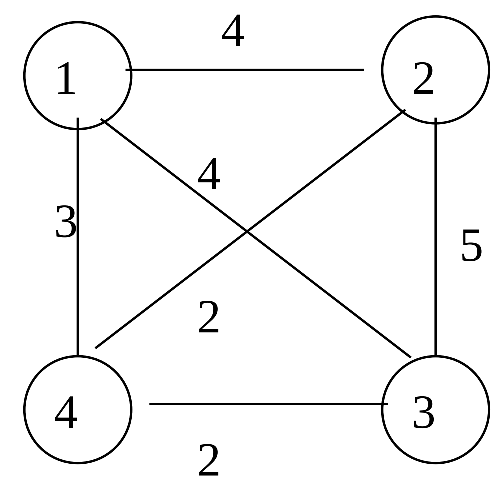
Minoux's Method: Example 2 (i)

The network $G(N,E)$ and relevant data for the fixed charge model (α_e, β_e) and offered traffic, t_{pq}, are as given in the figure below.

α_e, β_e:



t_{pq} :



$$c_e(f_e) = \alpha_e f_e + \beta_e.$$

Initially, assume direct routing i.e. $f_e = t_{pq}$ for all $e = (p,q)$, and $L = E$.

Minoux's Method: Example 2 (ii)

$\Delta_e = l_{\hat{\mu}}(\mathbf{f}) - (\alpha_e f_e + \beta_e) = \sum_{e' \in \hat{\mu}} \alpha_{e'} f_e - \alpha_e f_e - \beta_e$.

Iteration 1 Calculate all Δ_es:

e	l	$(l-\alpha)f-\beta$	$>0?$
$(1,2)$	2	$(2-1)4-3$	>0
$(1,3)$	3	$(3-1)4-6$	>0
$(1,4)$	2	$(2-1)3-5$	-2
$(2,3)$	2	$(2-2)5-3$	-3
$(2,4)$	2	$(2-1)2-6$	-4
$(3,4)$	2	$(2-2)2-6$	-6

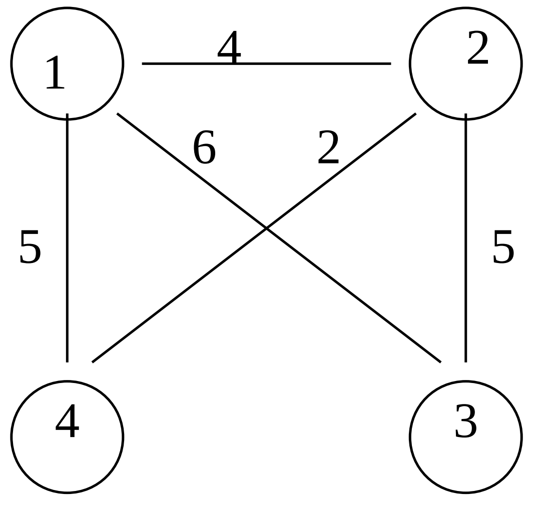
Therefore minΔ_e=-6, for $e = (3,4)$.

So delete link (3,4) and reroute its load onto the shortest path, 3-1-4.

Minoux's Method: Example 2 (iii)

Iteration 2: New loads are and Δ_e are

f_e:



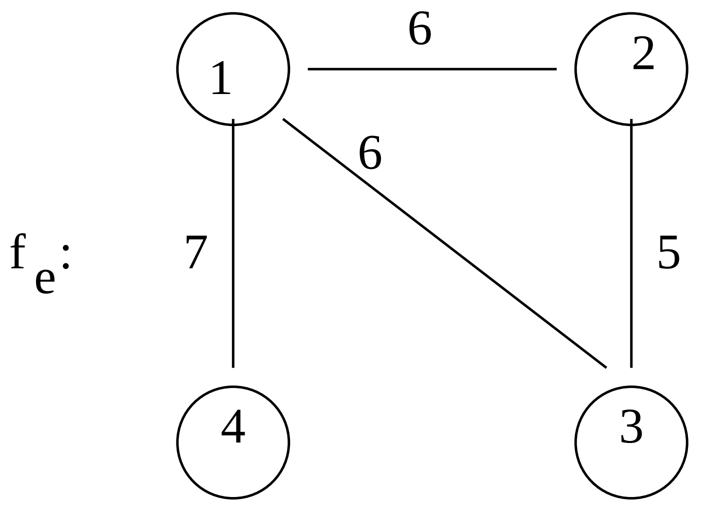
e	l	$(l-\alpha)f-\beta$	$>0?$
$(1,2)$	2	$(2-1)4-3$	>0
$(1,3)$	3	$(3-1)6-6$	>0
$(1,4)$	2	$(2-1)5-5$	$=0$
$(2,3)$	2	$(2-2)5-3$	-3
$(2,4)$	2	$(2-1)2-6$	-4

Therefore minΔ_e=-4, for $e=(2,4)$.

So delete link (2,4) and reroute its load onto the shortest path, 2-1-4.

Minoux's Method: Example 2 (iv)

Iteration 3: New loads are and Δ_e are



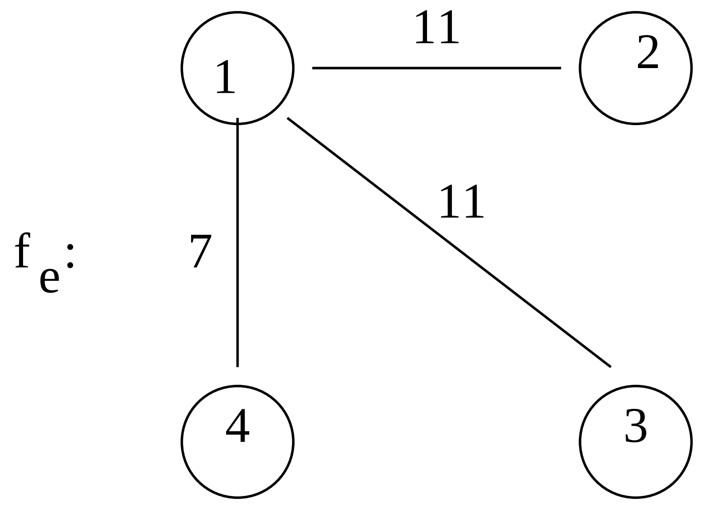
e	l	$(l-\alpha)f-\beta$	$>0?$
$(1,2)$	3	$(3-1)6-3$	>0
$(1,3)$	3	$(3-1)6-6$	>0
$(1,4)$	∞		
$(2,3)$	2	$(2-2)5-3$	-3

Therefore minΔ_e=-3, for $e=(2,3)$.

So delete link (2,3) and reroute its load onto the shortest path, 2-1-3.

Minoux's Method: Example 2 (v)

Iteration 4: New loads are



No further links can be deleted without disconnecting the network. Cost is 22+9+12=43.

Question: Is this optimal?

References

[1] M.Minoux, “Network synthesis and optimum network design problems: Models, solution methods and applications,” in Networks, vol. 19, pp. 313–360, 1989.