Node-Based Optimal Routing in Wireless Networks

Stephan M. Günther
Supervisor: Maximilian Riemensberger

25 June 2010
Internationales Hauptseminar Signalverarbeitung (IHS)

Outline

Motivation and Network Model

Cost Function

Optimality Conditions

Routing Algorithm

Summary

Motivation

Centralized algorithms rely on a designated node which

- gathers information about network from other nodes,
- obtains full knowledge of the network's topology, and
- makes routing decisions which are re-distributed to other nodes.

Motivation

Centralized algorithms rely on a designated node which

- gathers information about network from other nodes,
- obtains full knowledge of the network's topology, and
- makes routing decisions which are re-distributed to other nodes.

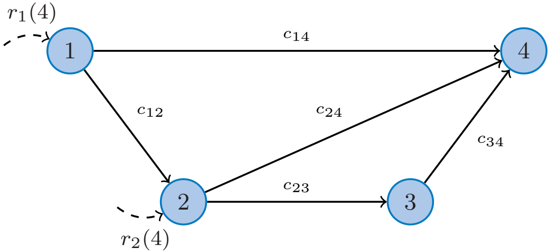
Node-Based algorithms do not employ a special node and therefore

- don't have a single point of failure,
- allow each node to make local routing decision, and
- often show better scaling behavior.

Motivation and Network Model

Network $N = (\mathcal{V}, \mathcal{A})$ consists of

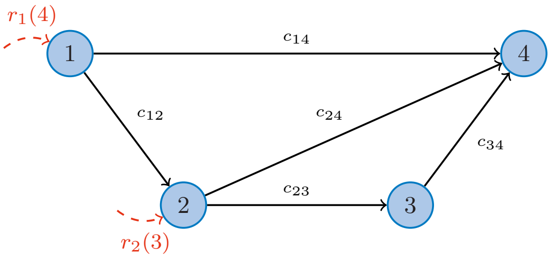
- nodes $\mathcal{V} = \{1, 2, 3, 4\}$,
- arcs $\mathcal{A} = \{(i, k) \mid i, k \in \mathcal{V} \wedge c(i, k) > 0\}$,
- link capacities $\mathcal{C} = \{c_{ik} \mid \forall (i, k) \in \mathcal{A}\}$,
- input rates $r_i(j)$ at node i destined for sink j.



Motivation and Network Model

Network $N = (\mathcal{V}, \mathcal{A})$ consists of

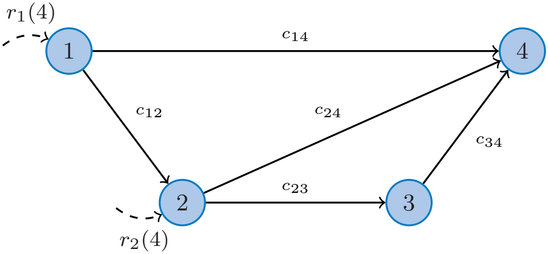
- nodes $\mathcal{V} = \{1, 2, 3, 4\}$,
- arcs $\mathcal{A} = \{(i, k) \mid i, k \in \mathcal{V} \wedge c(i, k) > 0\}$,
- link capacities $\mathcal{C} = \{c_{ik} \mid \forall (i, k) \in \mathcal{A}\}$,
- input rates $r_i(j)$ at node i destined for sink j.



Motivation and Network Model

Network $N = (\mathcal{V}, \mathcal{A})$ consists of

- nodes $\mathcal{V} = \{1, 2, 3, 4\}$,
- arcs $\mathcal{A} = \{(i,k) \mid i,k \in \mathcal{V} \wedge c(i,k) > 0\}$,
- link capacities $\mathcal{C} = \{c_{ik} \mid \forall (i,k) \in \mathcal{A}\}$,
- input rates $r_i(j)$ at node i destined for sink j.

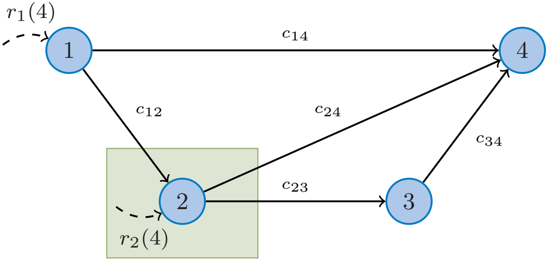


Question: How much flow destined for sink j is expected at node i?

Motivation and Network Model

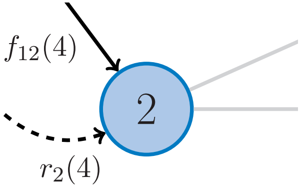
Network $N = (\mathcal{V}, \mathcal{A})$ consists of

- nodes $\mathcal{V} = \{1, 2, 3, 4\}$,
- arcs $\mathcal{A} = \{(i,k) \mid i,k \in \mathcal{V} \wedge c(i,k) > 0\}$,
- link capacities $\mathcal{C} = \{c_{ik} \mid \forall (i,k) \in \mathcal{A}\}$,
- input rates $r_i(j)$ at node i destined for sink j.



Question: How much flow destined for sink j is expected at node i?

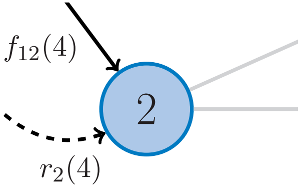
Motivation and Network Model



Total expected flow destined for sink j at node i is comprised of

- flow $f_{ki}(j)$ sent to i from k over incoming link (k, i) and
- additional flow $r_i(j)$ generated at node i.

Motivation and Network Model

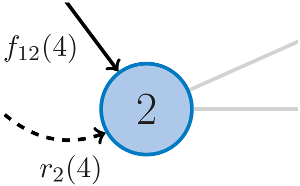


Total expected flow destined for sink j at node i is comprised of

- flow $f_{ki}(j)$ sent to i from k over incoming link (k,i) and
- additional flow $r_i(j)$ generated at node i.

$$t_i(j) = r_i(j) + \sum_{(k,i)\in\mathcal{A}} f_{ki}(j), \quad \forall j \in \mathcal{V}$$

Motivation and Network Model



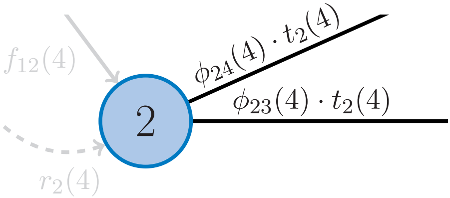
Total expected flow destined for sink j at node i is comprised of

- flow $f_{ki}(j)$ sent to i from k over incoming link (k, i) and
- additional flow $r_i(j)$ generated at node i.

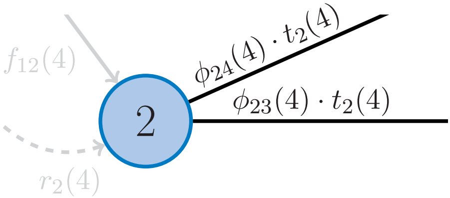
$$t_i(j) = r_i(j) + \sum_{(k,i)\in\mathcal{A}} f_{ki}(j), \quad \forall j \in \mathcal{V}$$

Next question: How is the flow $t_i(j)$ distributed to outgoing links?

Motivation and Network Model



Motivation and Network Model



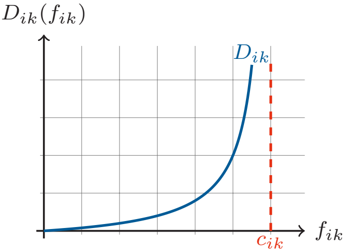
Flow $f_{ik}(j)$ towards j over link (i,k) is determined as fraction of expected traffic $t_i(j)$ according to routing variables $\phi_{ik}(j) \in \Phi$:

- $\phi_{ik}(j) = \frac{f_{ik}(j)}{t_i(j)} \geq 0, \quad \forall j \in \mathcal{V}, \ \forall (i,k) \in \mathcal{A}$,
- $\phi_{ik}(i) = 0, \quad \forall (i,k) \in \mathcal{A}$,
- $\sum_{(i,k) \in \mathcal{A}} \phi_{ik}(j) = 1, \quad \forall j \in \mathcal{V}$.

Outline

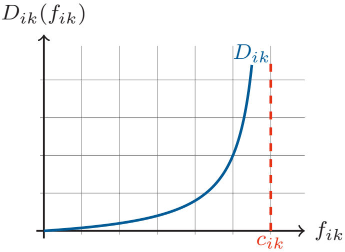
Cost Function

- Link flow $f_{ik} = \sum_{j \in \mathcal{V}} f_{ik}(j)$
- Link delay $D_{ik}(f_{ik})$,
  convex and increasing in f_{ik}
- Total delay $D_T = \sum_{(i,k) \in \mathcal{A}} D_{ik}(f_{ik})$



Cost Function

- Link flow $f_{ik} = \sum_{j \in \mathcal{V}} f_{ik}(j)$
- Link delay $D_{ik}(f_{ik})$, convex and increasing in f_{ik}
- Total delay $D_T = \sum_{(i,k) \in \mathcal{A}} D_{ik}(f_{ik})$



Optimization problem

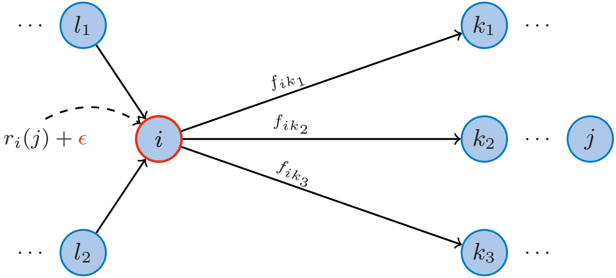
$$\min_{\phi} D_T = \min_{\phi} \sum_{(i,k) \in \mathcal{A}} D_{ik}(f_{ik})$$

s.t. $\phi \in \Phi,\ f_{ik} \leq c_{ik}\ \forall (i,k) \in \mathcal{A}$

Outline

Optimality Conditions

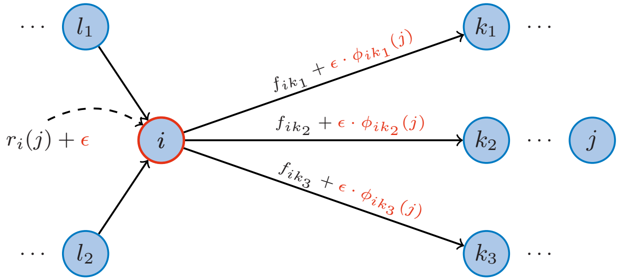
Consider a small increase ϵ of input rate $r_i(j)$:



$$\frac{\partial D_T}{\partial r_i(j)} =$$

Optimality Conditions

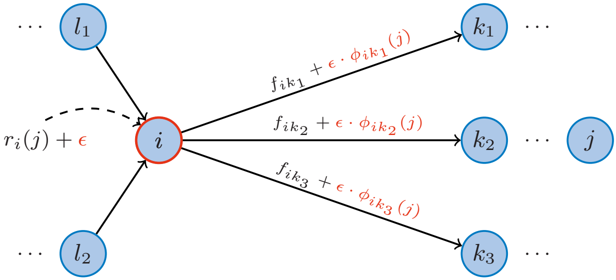
Consider a small increase ϵ of input rate $r_i(j)$:



$$\frac{\partial D_T}{\partial r_i(j)} =$$

Optimality Conditions

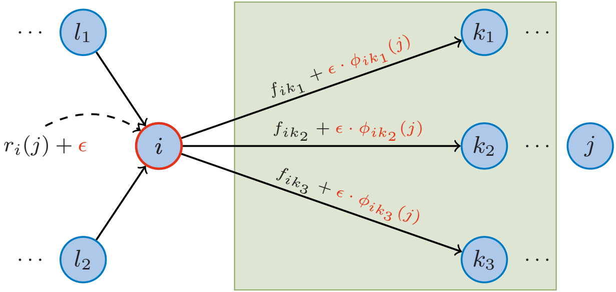
Consider a small increase ϵ of input rate $r_i(j)$:



$$\frac{\partial D_T}{\partial r_i(j)} = \sum_{(i,k)\in\mathcal{A}} \phi_{ik}(j) \frac{dD_{ik}}{df_{ik}} +$$

Optimality Conditions

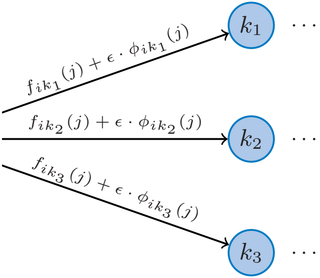
Consider a small increase ϵ of input rate $r_i(j)$:



$$\frac{\partial D_T}{\partial r_i(j)} = \sum_{(i,k)\in\mathcal{A}} \phi_{ik}(j) \frac{dD_{ik}}{df_{ik}} +$$

Optimality Conditions

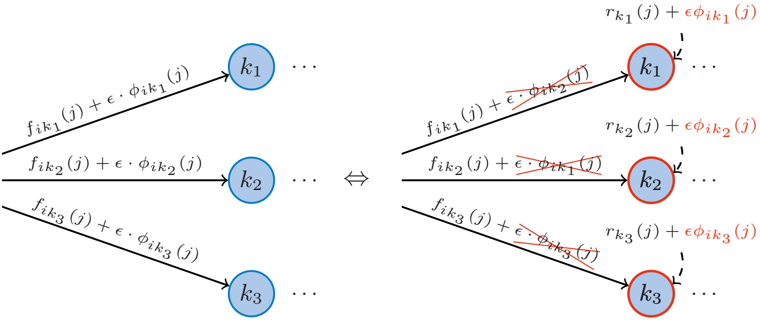
Consider a small increase ϵ of input rate $r_i(j)$:



$$\frac{\partial D_T}{\partial r_i(j)} = \sum_{(i,k)\in\mathcal{A}} \phi_{ik}(j) \frac{dD_{ik}}{df_{ik}} +$$

Optimality Conditions

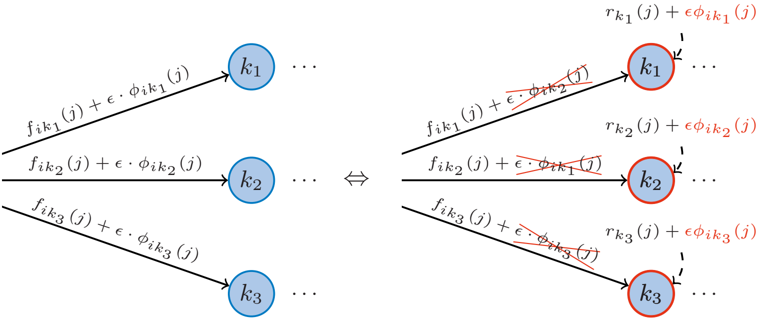
Consider a small increase ϵ of input rate $r_i(j)$:



$$\frac{\partial D_T}{\partial r_i(j)} = \sum_{(i,k)\in\mathcal{A}} \phi_{ik}(j)\frac{dD_{ik}}{df_{ik}} \; +$$

Optimality Conditions

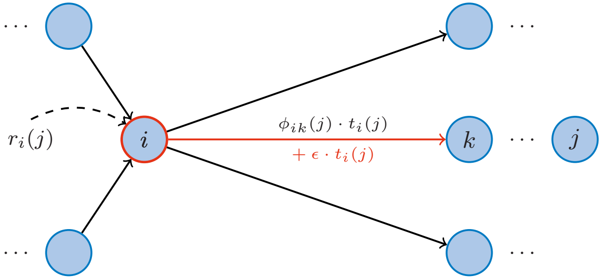
Consider a small increase ϵ of input rate $r_i(j)$:



$$\frac{\partial D_T}{\partial r_i(j)} = \sum_{(i,k)\in\mathcal{A}} \phi_{ik}(j)\frac{dD_{ik}}{df_{ik}} + \sum_{(i,k)\in\mathcal{A}} \phi_{ik}(j)\frac{\partial D_T}{\partial r_k(j)}$$

Optimality Conditions

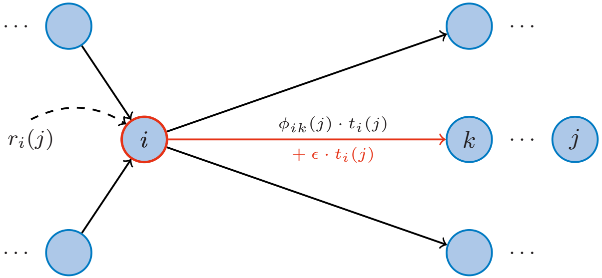
Next, consider a small increment ϵ in routing variable $\phi_{ik}(j)$:



$$\frac{\partial D_T}{\partial \phi_{ik}(j)} =$$

Optimality Conditions

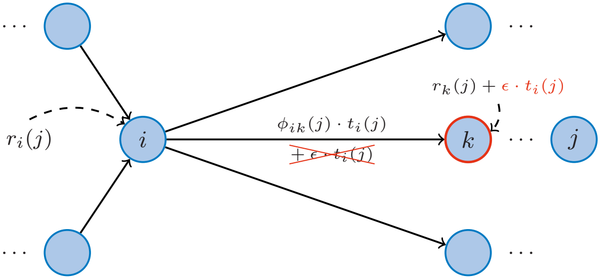
Next, consider a small increment ϵ in routing variable $\phi_{ik}(j)$:



$$\frac{\partial D_T}{\partial \phi_{ik}(j)} = t_i(j) \frac{dD_{ik}}{df_{ik}} +$$

Optimality Conditions

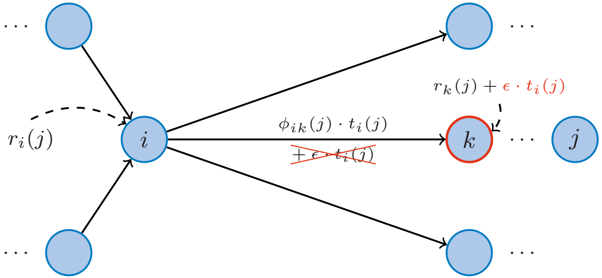
Next, consider a small increment ϵ in routing variable $\phi_{ik}(j)$:



$$\frac{\partial D_T}{\partial \phi_{ik}(j)} = t_i(j)\frac{dD_{ik}}{df_{ik}} +$$

Optimality Conditions

Next, consider a small increment ϵ in routing variable $\phi_{ik}(j)$:



$$\frac{\partial D_T}{\partial \phi_{ik}(j)} = t_i(j) \frac{dD_{ik}}{df_{ik}} + t_i(j) \frac{\partial D_T}{\partial r_k(j)}$$

Optimality Conditions

Change of delay with respect to

- change of input rates

$$\frac{\partial D_T}{\partial r_i(j)} = \sum_{(i,k)\in\mathcal{A}} \phi_{ik}(j) \left[\frac{dD_{ik}}{df_{ik}} + \frac{\partial D_T}{\partial r_k(j)}\right]$$

- change of routing variables

$$\frac{\partial D_T}{\partial \phi_{ik}(j)} = t_i(j) \cdot \left[\frac{dD_{ik}}{df_{ik}} + \frac{\partial D_T}{\partial r_k(j)}\right]$$

Optimality Conditions

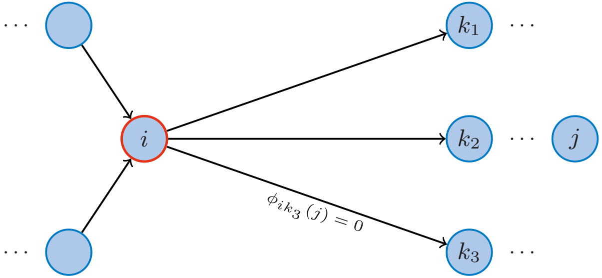
Necessary condition (derived via KKT)

$$\frac{\partial D_T}{\partial \phi_{ik}(j)} \begin{cases} = \lambda_{ij}, & \phi_{ik}(j) > 0 \\ \geq \lambda_{ij}, & \phi_{ik}(j) = 0 \end{cases}, \ \forall i \neq j, \ \forall (i,k) \in \mathcal{A}$$

Optimality Conditions

Necessary condition (derived via KKT)

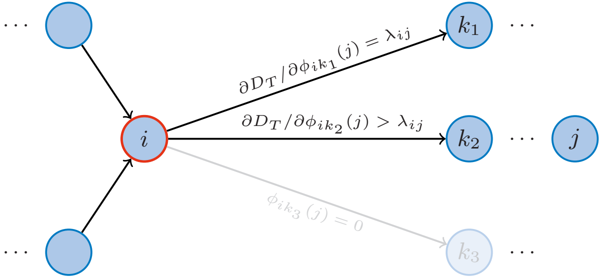
$$\frac{\partial D_T}{\partial \phi_{ik}(j)} \begin{cases} = \lambda_{ij}, & \phi_{ik}(j) > 0 \\ \geq \lambda_{ij}, & \phi_{ik}(j) = 0 \end{cases}, \ \forall i \neq j, \ \forall (i,k) \in \mathcal{A}$$



Optimality Conditions

Necessary condition (derived via KKT)

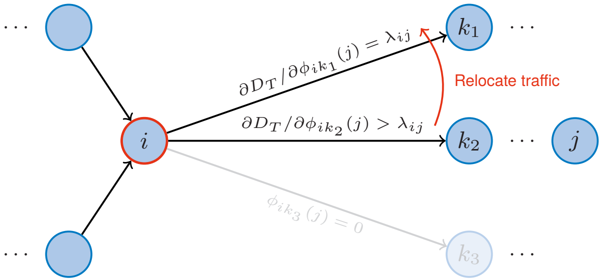
$$\frac{\partial D_T}{\partial \phi_{ik}(j)} \begin{cases} = \lambda_{ij}, & \phi_{ik}(j) > 0 \\ \geq \lambda_{ij}, & \phi_{ik}(j) = 0 \end{cases}, \ \forall i \neq j, \ \forall (i,k) \in \mathcal{A}$$



Optimality Conditions

Necessary condition (derived via KKT)

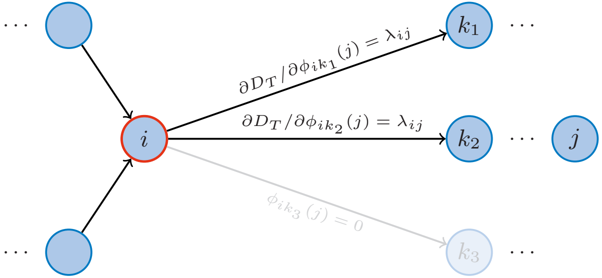
$$\frac{\partial D_T}{\partial \phi_{ik}(j)} \begin{cases} = \lambda_{ij}, & \phi_{ik}(j) > 0 \\ \geq \lambda_{ij}, & \phi_{ik}(j) = 0 \end{cases}, \ \forall i \neq j, \ \forall (i,k) \in \mathcal{A}$$



Optimality Conditions

Necessary condition (derived via KKT)

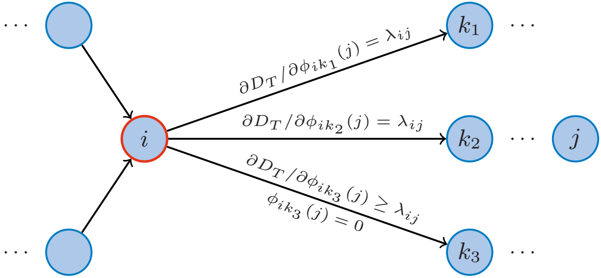
$$\frac{\partial D_T}{\partial \phi_{ik}(j)} \begin{cases} = \lambda_{ij}, & \phi_{ik}(j) > 0 \\ \geq \lambda_{ij}, & \phi_{ik}(j) = 0 \end{cases}, \; \forall i \neq j, \; \forall (i,k) \in \mathcal{A}$$



Optimality Conditions

Necessary condition (derived via KKT)

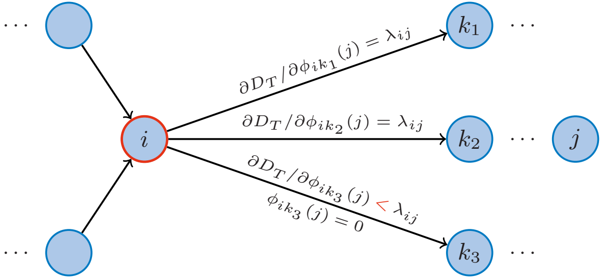
$$\frac{\partial D_T}{\partial \phi_{ik}(j)} \begin{cases} = \lambda_{ij}, & \phi_{ik}(j) > 0 \\ \geq \lambda_{ij}, & \phi_{ik}(j) = 0 \end{cases}, \ \forall i \neq j, \ \forall (i,k) \in \mathcal{A}$$



Optimality Conditions

Necessary condition (derived via KKT)

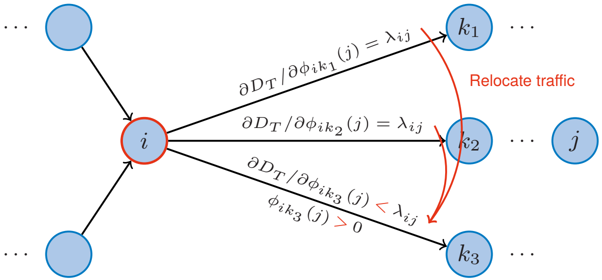
$$\frac{\partial D_T}{\partial \phi_{ik}(j)} \begin{cases} = \lambda_{ij}, & \phi_{ik}(j) > 0 \\ \geq \lambda_{ij}, & \phi_{ik}(j) = 0 \end{cases}, \ \forall i \neq j, \ \forall (i,k) \in \mathcal{A}$$



Optimality Conditions

Necessary condition (derived via KKT)

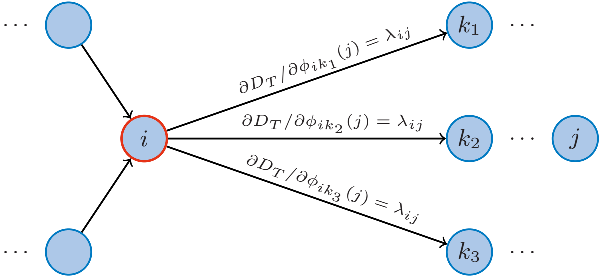
$$\frac{\partial D_T}{\partial \phi_{ik}(j)} \begin{cases} = \lambda_{ij}, & \phi_{ik}(j) > 0 \\ \geq \lambda_{ij}, & \phi_{ik}(j) = 0 \end{cases}, \ \forall i \neq j, \ \forall (i,k) \in \mathcal{A}$$



Optimality Conditions

Necessary condition (derived via KKT)

$$\frac{\partial D_T}{\partial \phi_{ik}(j)} \begin{cases} = \lambda_{ij}, & \phi_{ik}(j) > 0 \\ \geq \lambda_{ij}, & \phi_{ik}(j) = 0 \end{cases}, \ \forall i \neq j, \ \forall (i,k) \in \mathcal{A}$$

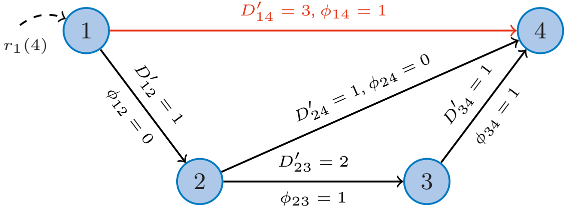


Optimality Conditions

Necessary, but not sufficient

$$\frac{\partial D_T}{\partial \phi_{ik}(j)} = t_i(j) \cdot \left[\frac{dD_{ik}}{df_{ik}} + \frac{\partial D_T}{\partial r_k(j)} \right] \begin{cases} = \lambda_{ij}, & \phi_{ik}(j) > 0 \\ \geq \lambda_{ij}, & \phi_{ik}(j) = 0 \end{cases}$$

Assume only node 1 generates traffic destined for node 4:

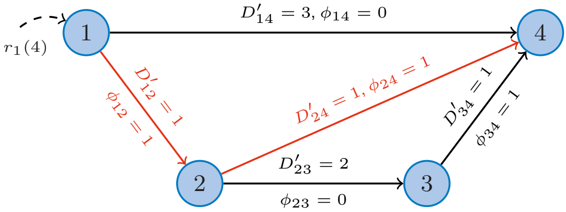


Optimality Conditions

Necessary, but not sufficient

$$\frac{\partial D_T}{\partial \phi_{ik}(j)} = t_i(j) \cdot \left[\frac{dD_{ik}}{df_{ik}} + \frac{\partial D_T}{\partial r_k(j)} \right] \begin{cases} = \lambda_{ij}, & \phi_{ik}(j) > 0 \\ \geq \lambda_{ij}, & \phi_{ik}(j) = 0 \end{cases}$$

Assume only node 1 generates traffic destined for node 4:



Optimality Conditions

Necessary, but not sufficient

$$\frac{\partial D_T}{\partial \phi_{ik}(j)} = t_i(j) \cdot \left[\frac{dD_{ik}}{df_{ik}} + \frac{\partial D_T}{\partial r_k(j)} \right] \begin{cases} = \lambda_{ij}, & \phi_{ik}(j) > 0 \\ \geq \lambda_{ij}, & \phi_{ik}(j) = 0 \end{cases}$$

Therefore, we find that for all $i \neq j$

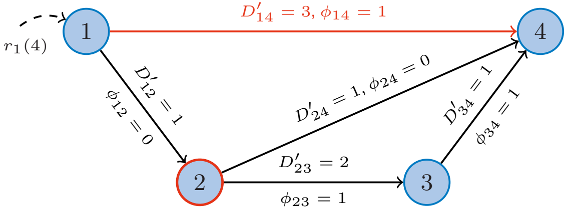
$$\frac{\partial D_T}{\partial r_i(j)} = \sum_{(i,k)\in\mathcal{A}} \phi_{ik}(j) \left[\frac{dD_{ik}}{df_{ik}} + \frac{\partial D_T}{\partial r_k(j)} \right]$$

$$\begin{cases} = \dfrac{dD_{ik}}{df_{ik}} + \dfrac{\partial D_T}{\partial r_k(j)}, & \text{if } \forall (i,k) \in \mathcal{A}:\ \phi_{ik}(j) > 0, \\ \leq \dfrac{dD_{ik}}{df_{ik}} + \dfrac{\partial D_T}{\partial r_k(j)}, & \text{if } \exists (i,k) \in \mathcal{A}:\ \phi_{ik}(j) = 0. \end{cases}$$

Optimality Conditions

Sufficient condition

$$\frac{\partial D_T}{\partial r_i(j)} \leq \frac{dD_{ik}}{df_{ik}} + \frac{\partial D_T}{\partial r_k(j)}, \ \forall i \neq j, \ \forall (i,k) \in \mathcal{A}$$



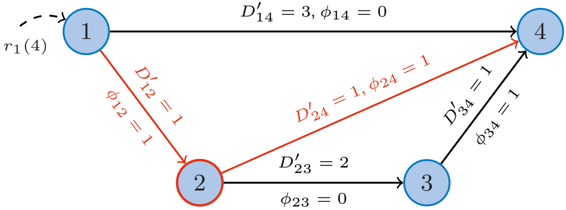
$$\frac{\partial D_T}{\partial r_2(4)} = \phi_{24}(4)\left[\frac{dD_{24}}{df_{24}} + \frac{\partial D_T}{r_4(4)}\right] + \phi_{23}(4)\left[\frac{dD_{23}}{df_{23}} + \frac{\partial D_T}{r_3(3)}\right] = 3$$

$$\frac{dD_{24}}{df_{24}} + \frac{\partial D_T}{r_4(4)} = 1$$

Optimality Conditions

Sufficient condition

$$\frac{\partial D_T}{\partial r_i(j)} \leq \frac{dD_{ik}}{df_{ik}} + \frac{\partial D_T}{\partial r_k(j)},\ \forall i \neq j,\ \forall (i,k) \in \mathcal{A}$$



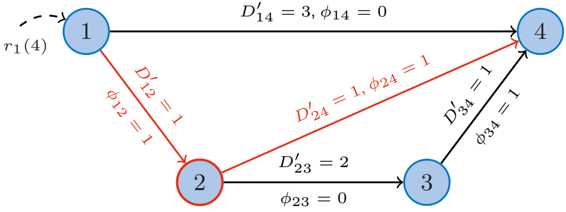
$$\frac{\partial D_T}{\partial r_2(4)} = \phi_{24}(4)\left[\frac{dD_{24}}{df_{24}} + \frac{\partial D_T}{r_4(4)}\right] + \phi_{23}(4)\left[\frac{dD_{23}}{df_{23}} + \frac{\partial D_T}{r_3(3)}\right] = 1$$

$$\frac{dD_{24}}{df_{24}} + \frac{\partial D_T}{r_4(4)} = 1$$

Optimality Conditions

Sufficient condition

$$\frac{\partial D_T}{\partial r_i(j)} \leq \frac{dD_{ik}}{df_{ik}} + \frac{\partial D_T}{\partial r_k(j)}, \; \forall i \neq j, \; \forall (i,k) \in \mathcal{A}$$



Proof: Skipped (technical), can be found in: *Robert G. Gallager, A Minimum Delay Routing Algorithm Using Distributed Computation*.

Outline

Routing Algorithm

We consider an iterative routing algorithm consisting of two phases:

1. Propagation of delay gradients to upstream neighbors
2. Update of routing variables according to information received from downstream neighbors

Routing Algorithm

We consider an iterative routing algorithm consisting of two phases:

1. Propagation of delay gradients to upstream neighbors
2. Update of routing variables according to information received from downstream neighbors

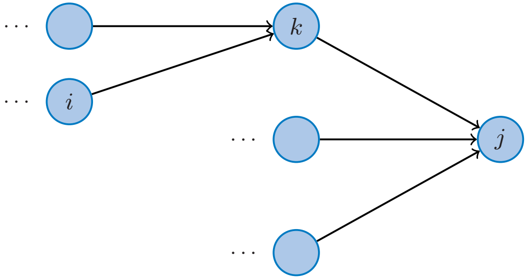
For the following example assume that

- only one flow exists (traffic destined for sink j) and
- the network is loop-free.

Both assumptions ease notation and can be relaxed
(networks with loops require some extensions to the algorithm).

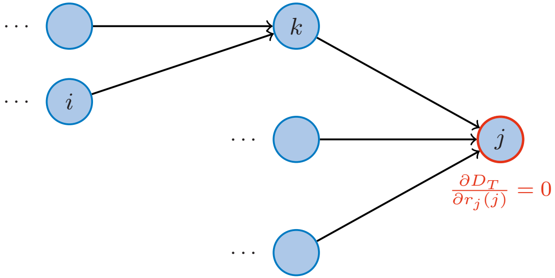
Routing Algorithm

Phase 1: Propagation of delay gradients to upstream neighbors



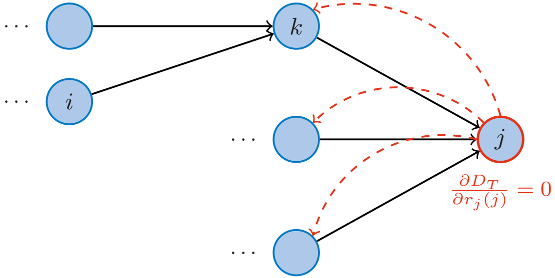
Routing Algorithm

Phase 1: Propagation of delay gradients to upstream neighbors



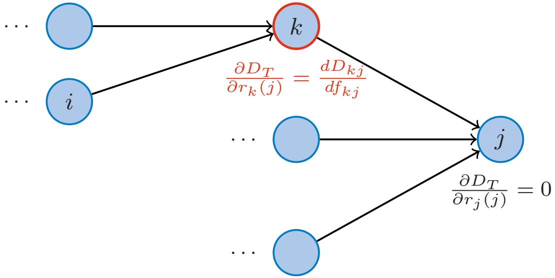
Routing Algorithm

Phase 1: Propagation of delay gradients to upstream neighbors



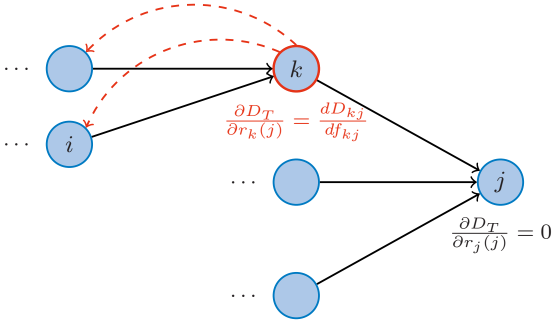
Routing Algorithm

Phase 1: Propagation of delay gradients to upstream neighbors



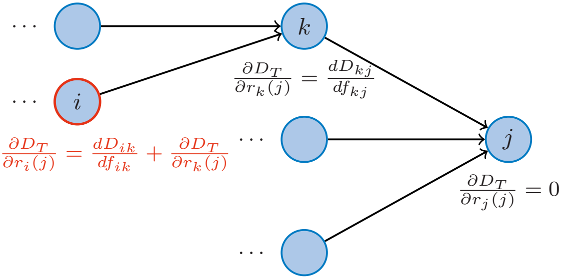
Routing Algorithm

Phase 1: Propagation of delay gradients to upstream neighbors



Routing Algorithm

Phase 1: Propagation of delay gradients to upstream neighbors



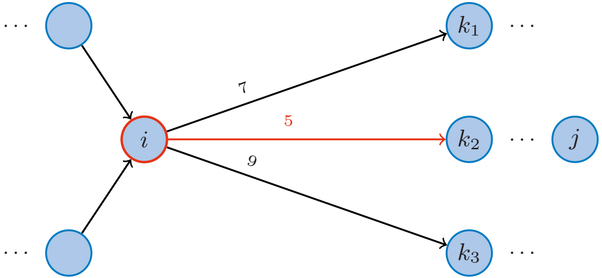
Routing Algorithm

Phase 2: Update of routing variables

Each node $i \in \mathcal{V}$ performs the following steps:

1. Choose the best link (i, m_i) with

$$m_i = \underset{\{k \mid (i,k) \in \mathcal{A}\}}{\arg\min} \left[\frac{dD_{ik}}{df_{ik}} + \frac{\partial D_T}{\partial r_k(j)} \right].$$



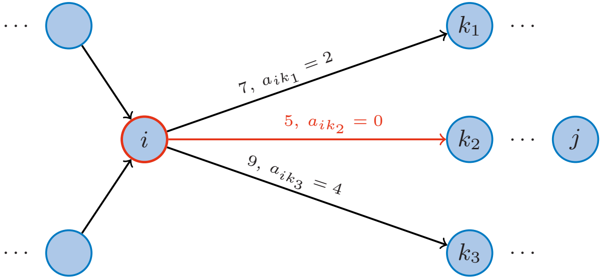
Routing Algorithm

Phase 2: Update of routing variables

Each node $i \in \mathcal{V}$ performs the following steps:

2. Between any link (i, k) and the best link calculate the difference of delay slopes

$$a_{ik}(j) := \frac{dD_{ik}}{df_{ik}} + \frac{\partial D_T}{\partial r_k(j)} - \left[\frac{dD_{im_i}}{df_{im_i}} + \frac{\partial D_T}{\partial r_{m_i}(j)}\right].$$



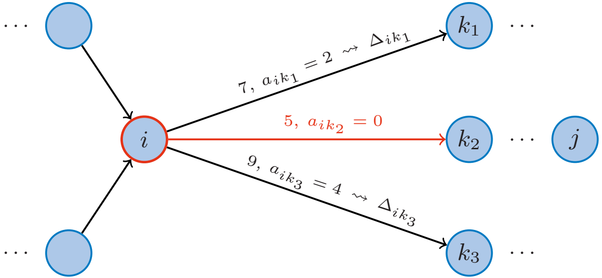
Routing Algorithm

Phase 2: Update of routing variables

Each node $i \in \mathcal{V}$ performs the following steps:

3. For each routing variable $\phi_{ik}(j)$ determine correction term

$$\Delta_{ik}(j) := \min\left\{\phi_{ik}(j), \frac{a_{ik}(j)}{t_i(j)}\right\}.$$



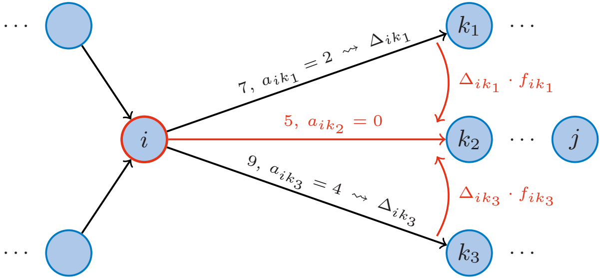
Routing Algorithm

Phase 2: Update of routing variables

Each node $i \in \mathcal{V}$ performs the following steps:

4. Update routing variables for step n:

$$\phi_{ik}(j)^{(n)} = \begin{cases} \phi_{ik}^{(n-1)}(j) - \Delta_{ik}(j), & k \neq m_i, \\ \phi_{ik}^{(n-1)}(j) + \sum\limits_{k \neq m_i} \Delta_{ik}(j), & k = m_i. \end{cases}$$



Outline

Summary

We saw an

- iterative routing algorithm
- operating in a distributed manner and
- converging to a minimum-delay solution

by usage of delay information from downstream neighbors.

Summary

We saw an

- iterative routing algorithm
- operating in a distributed manner and
- converging to a minimum-delay solution

by usage of delay information from downstream neighbors.

It can be adapted to allow for

- networks with loops ($\rightarrow$ blocking set),
  (Robert G. Gallager, A Minimum Delay Routing Algorithm Using Distributed Computation), and
- power control ($\rightarrow$ power control and allocation variables),
  (Yufang Xi and Edmund Yeh, Node-Based Optimal Power Control, Routing, and Congestion Control in Wireless Networks).