CIE4801 Transportation and spatial modelling
Congested assignment

Rob van Nes, Transport & Planning

31-08-18

Content

- Comments/questions uncongested assignment
  - Plus remaining part
- Modelling component 5: Assignment: congested conditions
  - Congested case
  - Your comments/questions on Chapter 10 and 11
  - Use of the assignment map
  - Practical issues

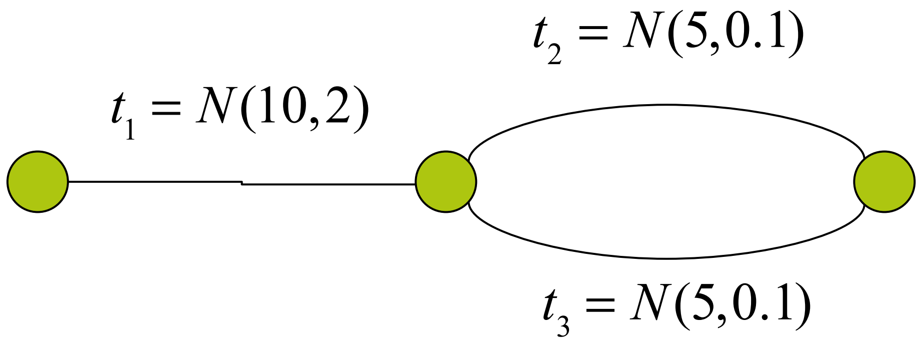
1.

Comments/questions last lecture

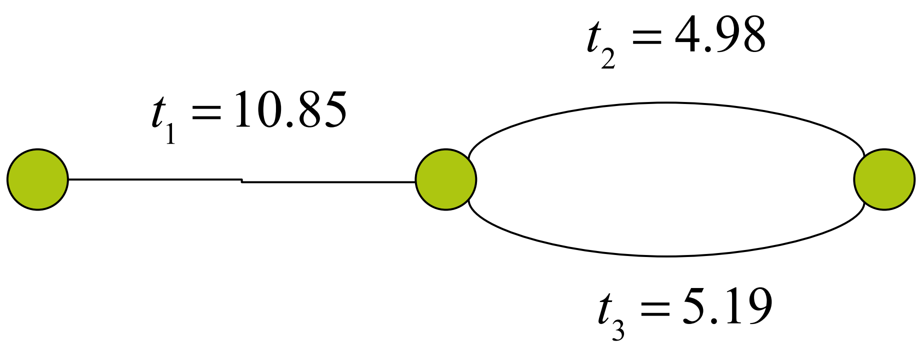
Comments/questions

- Uncongested assignment
  - Shortest path algorithm
  - All-or-nothing
  - Assignment map
  - Probit and logit
- Remaining part
  - Link level versus route level

Example link level

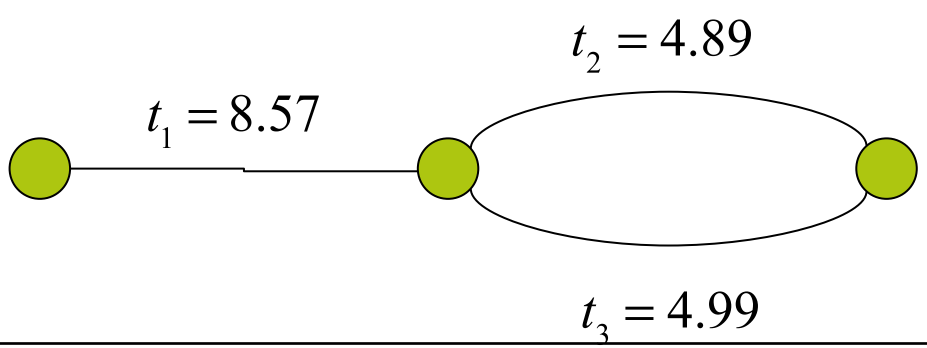


State 1



Route 1=15.83
Route 2=16.04

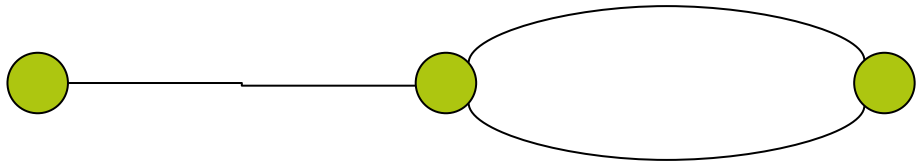
State 2



Route 1=13.46
Route 2=13.56

Switch from link level to route level

$t_{r1} = N(15,2)$

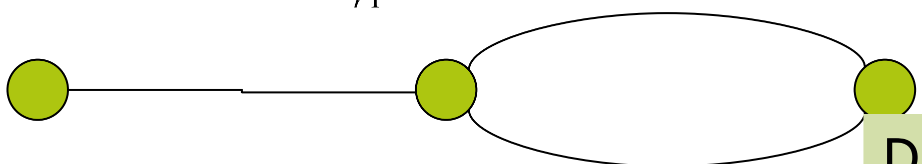


$t_{r2} = N(15,2)$

Thus sampling route travel times instead of link travel times

$t_{r1} = 15.85$

State 1

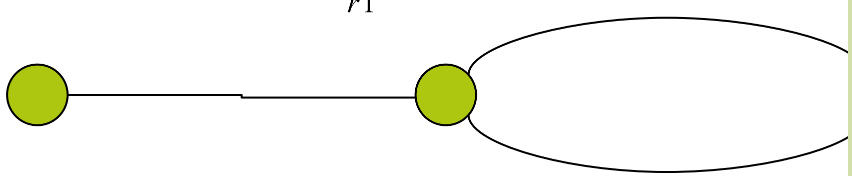


Route 1=15.83

$t_{r2} = 14.24$

$t_{r1} = 16.78$

State 2



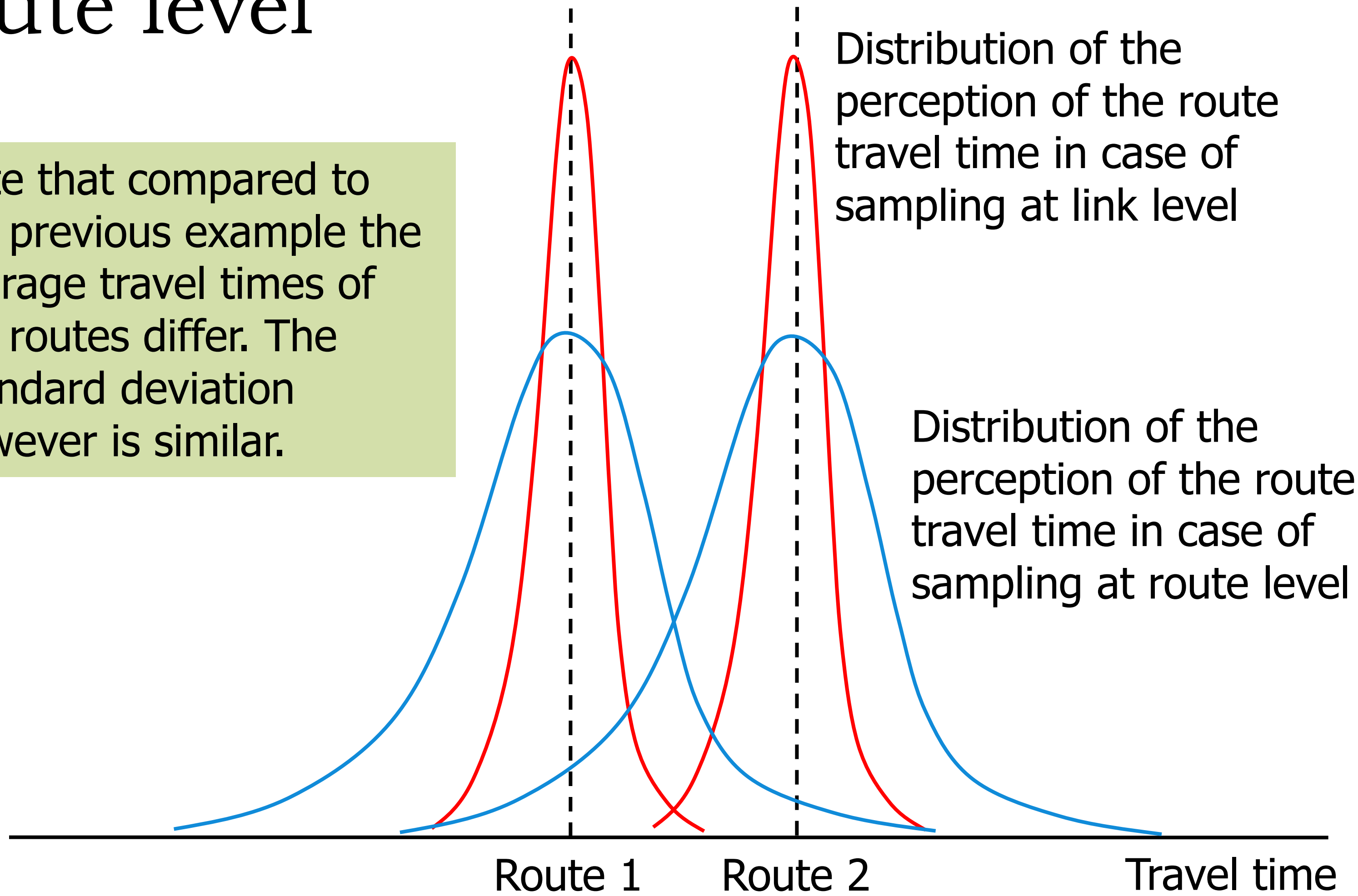
$t_{r2} = 13.57$

Differences between routes is much larger due to (implicit) different assumption for time of link 1 within a given state

This approach is therefore not correct

Effect of sampling at link level versus route level

Note that compared to the previous example the average travel times of the routes differ. The Standard deviation however is similar.



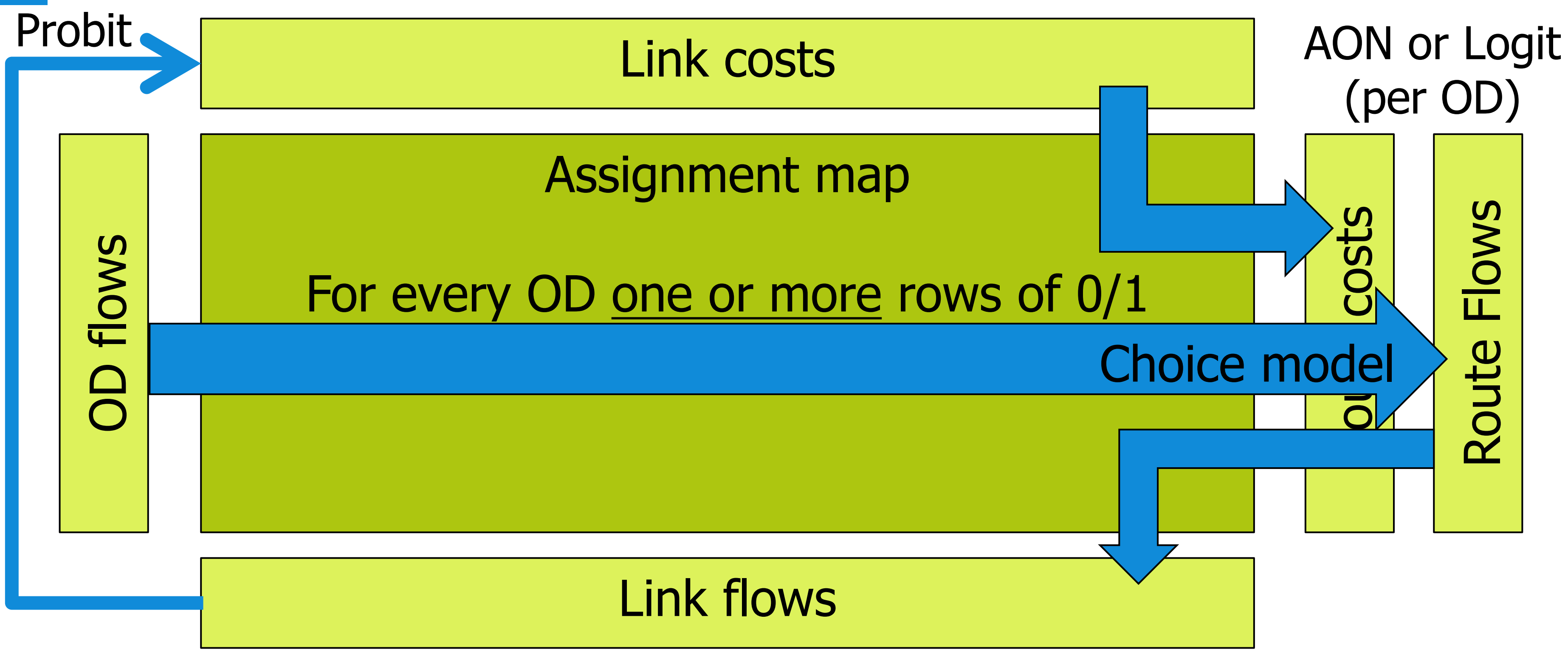
Stochastic assignment and path representation

- Iterative scheme as in Probit is tailor-made for tree search algorithms (or vice versa)

- Assignment map approach can be used as well, especially for Logit
  - Generate routes first,
    e.g. manually or using repetitive shortest path searches while systematically eliminating or penalizing links (k-shortest path algorithm)
  - Check whether routes are realistic (e.g. large or very short detours)

- Advantage of assignment map is that you can check for overlap!

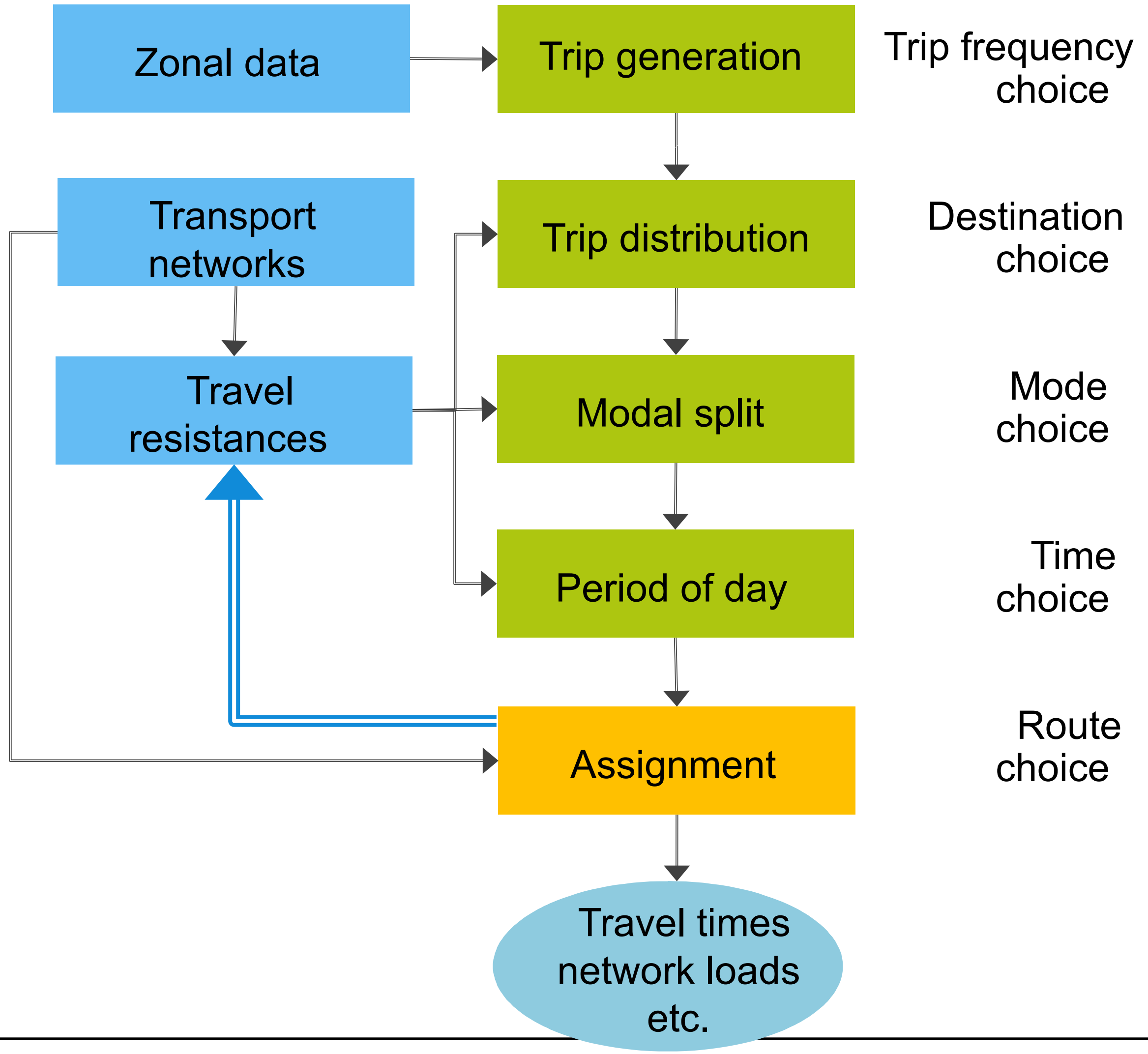
Assignment map and stochastic assignment



2.

Congested assignment

Introduction congested assignment



Topics for discussion

- What does this modelling component do? What's its output and what's its input? How does it fit in the framework?
- The main concepts
  - Wardrop's equilibrium, speed-flow curves
- The modelling methods
  - Iterative scheme: Method successive averages (MSA)
  - Convergence criteria: Relative or Duality Gap
  - Mathematical programming approach
  - Frank-Wolfe algorithm versus MSA
  - Route based assignment
  - Stochastic equilibrium assignment (SUE)
    - NB Probit or Logit makes quite a difference!
  - Braess's paradox
- Practical issues
- Are these models appropriate?

3.0

Generic formulation of the network assignment problem

General network assignment problem: Main elements

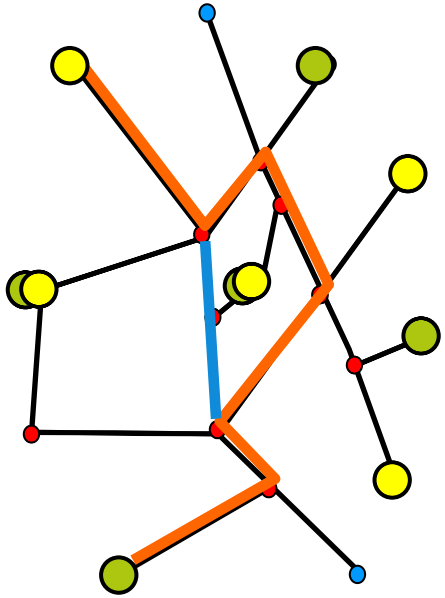
<u>Indices:</u>

origin *i*

destination *j*

route *r*

link *a*



Network assignment variables (comment on changes)?

INPUT		OUTPUT	
T_{ij}	OD travel demand	T_{ijr}	route flow
α_{ijr}^{a}	link-route incidence matrix (assignment map)	ϕ_{ijr}	route choice proportion
$t_a(q_a)$	link travel time function	t_{ijr}	route travel time
Θ	flow dispersion parameter of the stochastic component	q_a	link flow
		t_a	link travel time

Relationships between variables

1. Link travel times t_a

$$t_a = t_a(q_a)$$

2. Route choice proportions ϕ_{ijr}

$$\phi_{ijr} = \phi_{ijr}(\Theta, t_{ijr})$$

3. Route travel times t_{ijr}

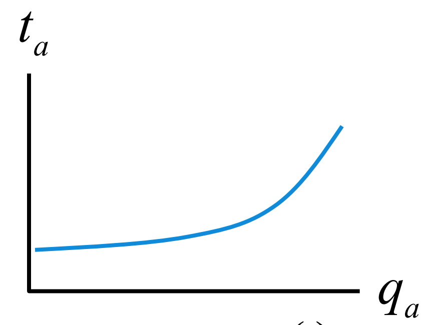
$$t_{ijr} = \sum_a \alpha_{ijr}^a t_a$$

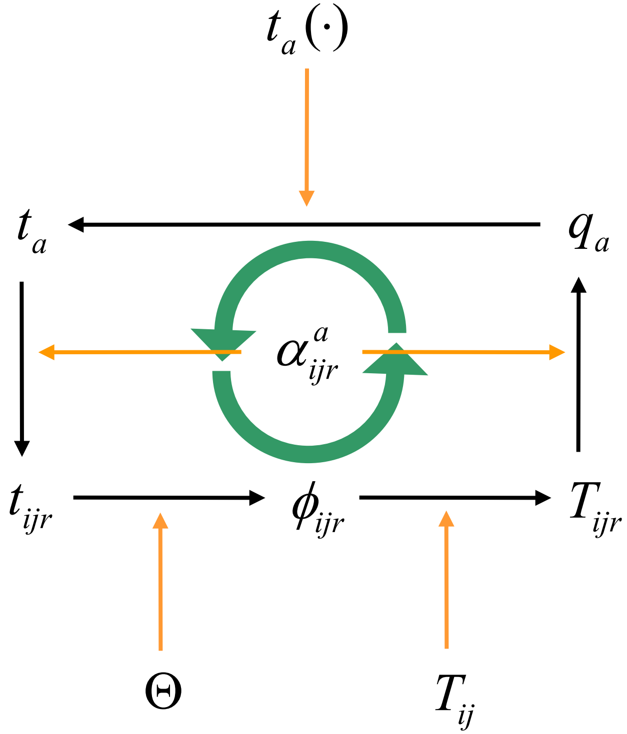
4. Route flows T_{ijr}

$$T_{ijr} = \phi_{ijr} T_{ij}$$

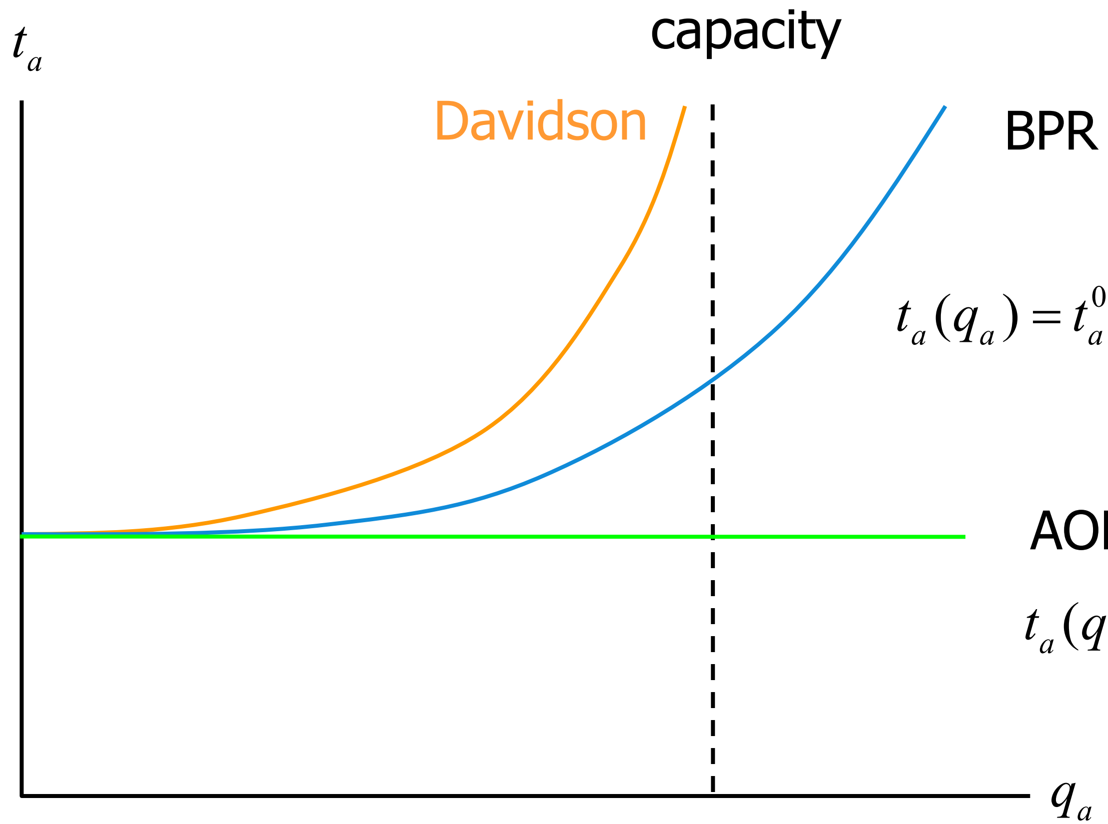
5. Link flows q_a

$$q_a = \sum_i \sum_j \sum_r \alpha_{ijr}^a T_{ijr}$$





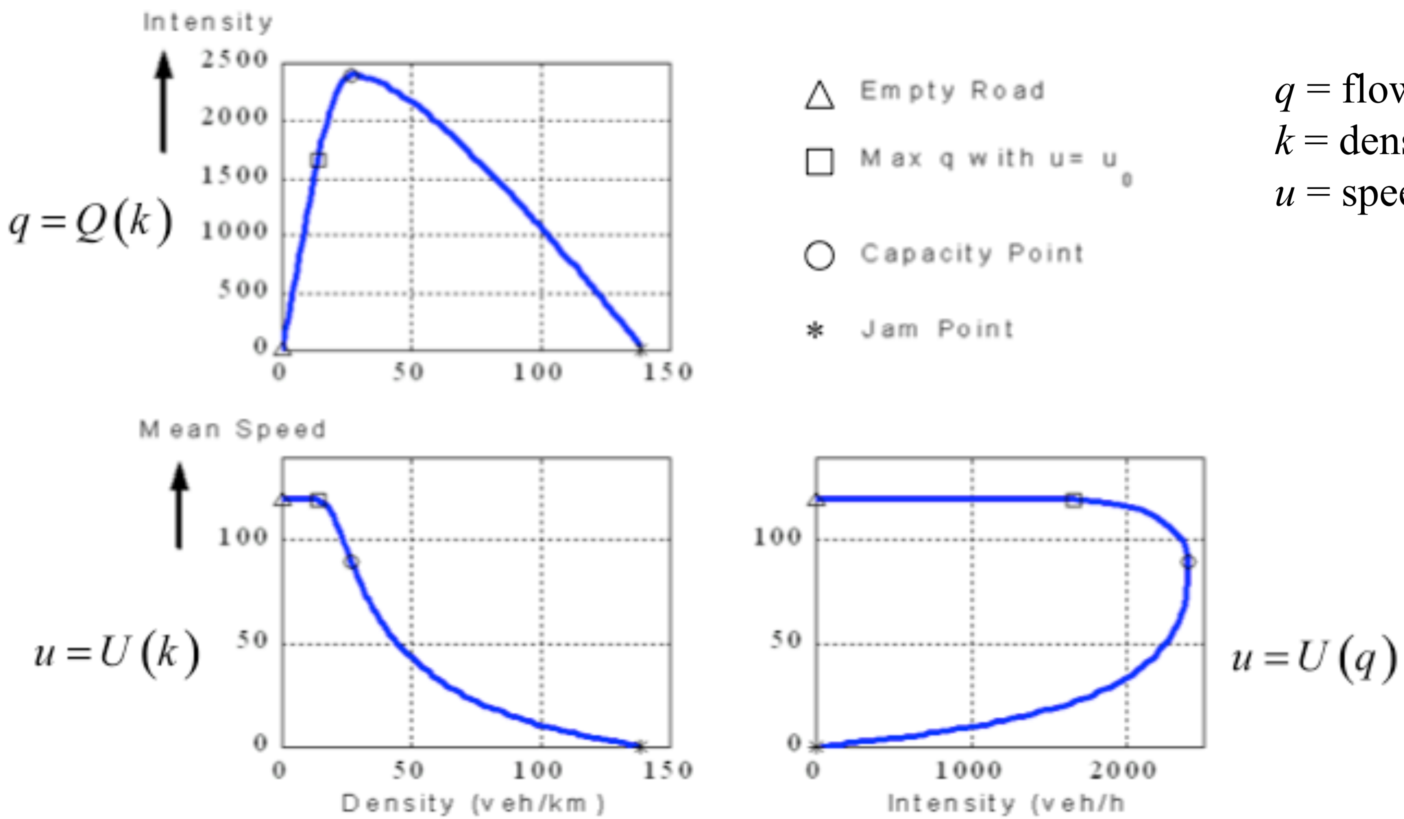
Link performance functions



$$t_a(q_a) = t_a^0 \left(1 + \alpha \left(\frac{q_a}{C_a} \right)^{\beta} \right)$$

$$t_a(q_a) = t_a^0$$

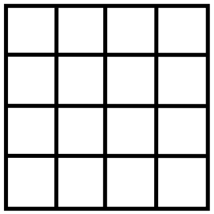
Traffic flow theory: Fundamental diagram



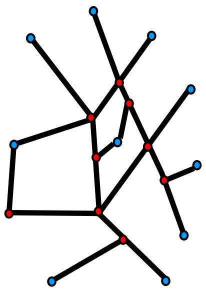
q = flow (veh/hr)
k = density (veh/km)
u = speed (km/hr)

Assignment types

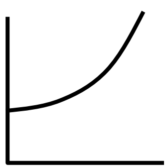
inputs



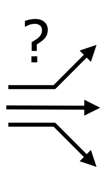
T_{ij}



α_{ijr}^{a}

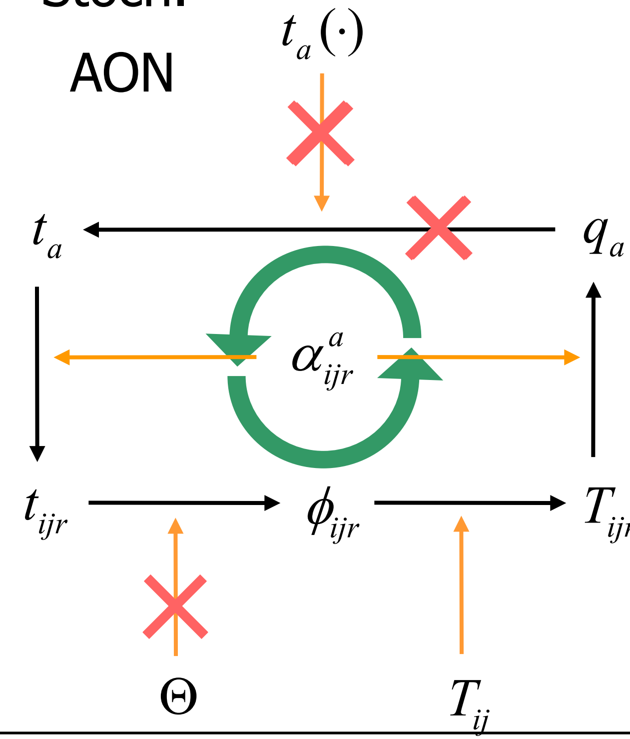


$t_a(q_a)$



Θ

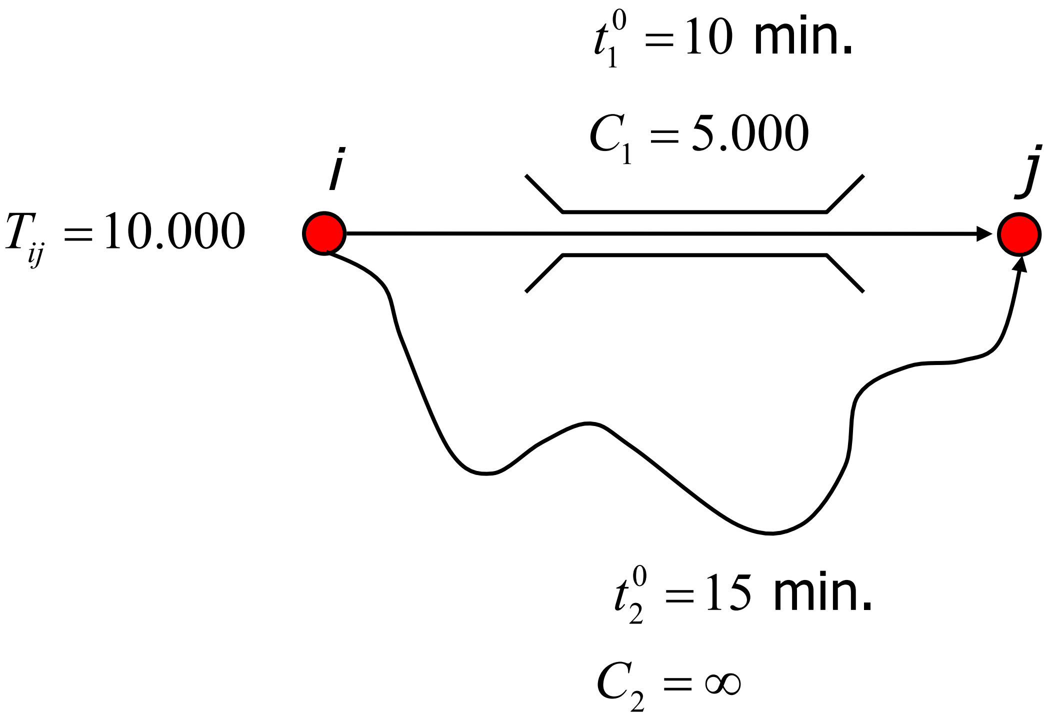
$t_a(q_a)$	Θ	
$t_a(q_a)$	>0	SUE
$t_a(q_a)$	$=0$	DUE
t_a	>0	Stoch.
t_a	$=0$	AON



3.1

DUE: Deterministic user equilibrium assignment: main concept

Route choice with congestion effects



When congestion is taken into account,
how long will the trip from *i* to *j* take along route 1? 15 min.!

→ DUE

DUE assignment

1. Link travel times t_a

$$t_a = t_a(q_a)$$

2. Route choice proportions ϕ_{ijr}

$$\phi_{ijr} = \phi_{ijr}(t_{ijr})$$

3. Route travel times t_{ijr}

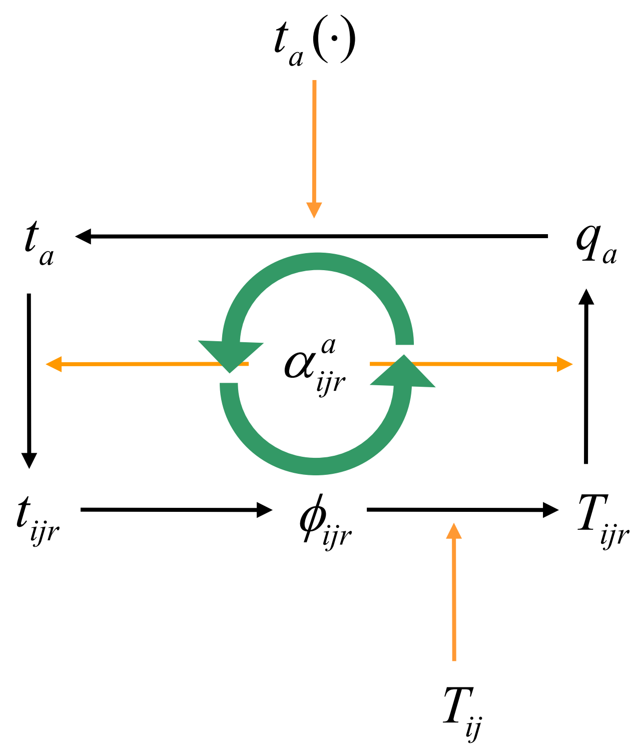
$$t_{ijr} = \sum_a \alpha_{ijr}^a t_a$$

4. Route flows T_{ijr}

$$T_{ijr} = \phi_{ijr} T_{ij}$$

5. Link flows q_a

$$q_a = \sum_i \sum_j \sum_r \alpha_{ijr}^a T_{ijr}$$



Main principle

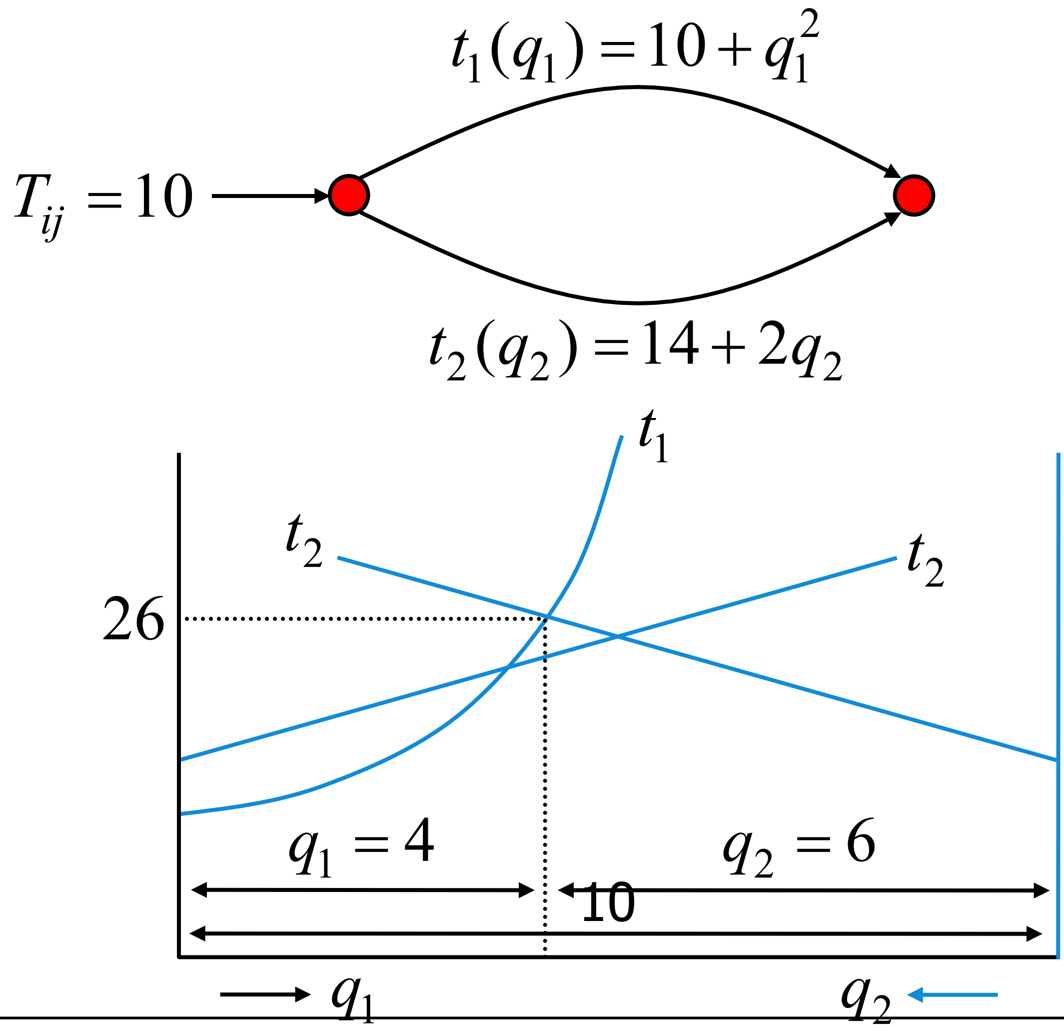
Deterministic user-equilibrium (DUE) takes congestion effects into account and is defined as:

Wardrop's equilibrium law
All travellers choose their optimal route, such that no traveller can improve his/her travel time by unilaterally changing routes.

This equilibrium is reached if the following condition holds:

Wardrop's first principle
All used routes have the same travel time which is not greater than the travel time on any unused route.

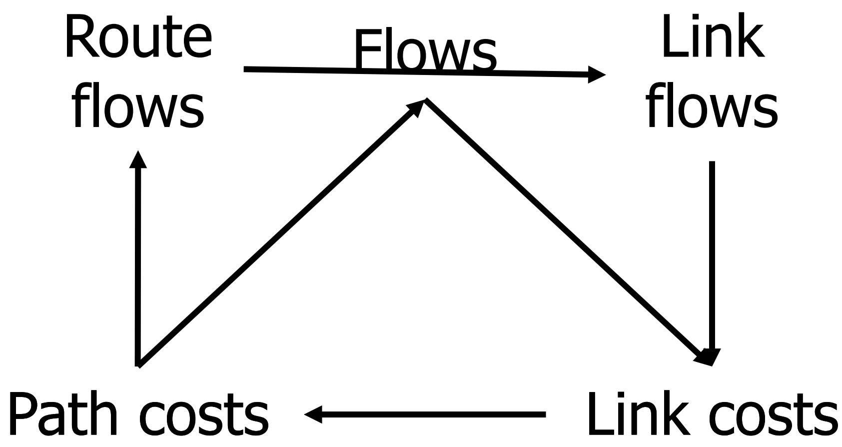
Example DUE assignment



3.2

DUE: Algorithms: MSA

Key point assignment problem



General solution scheme:
- Start with an assumption on e.g. link costs
- Follow the arrows until convergence is achieved

Solution principle

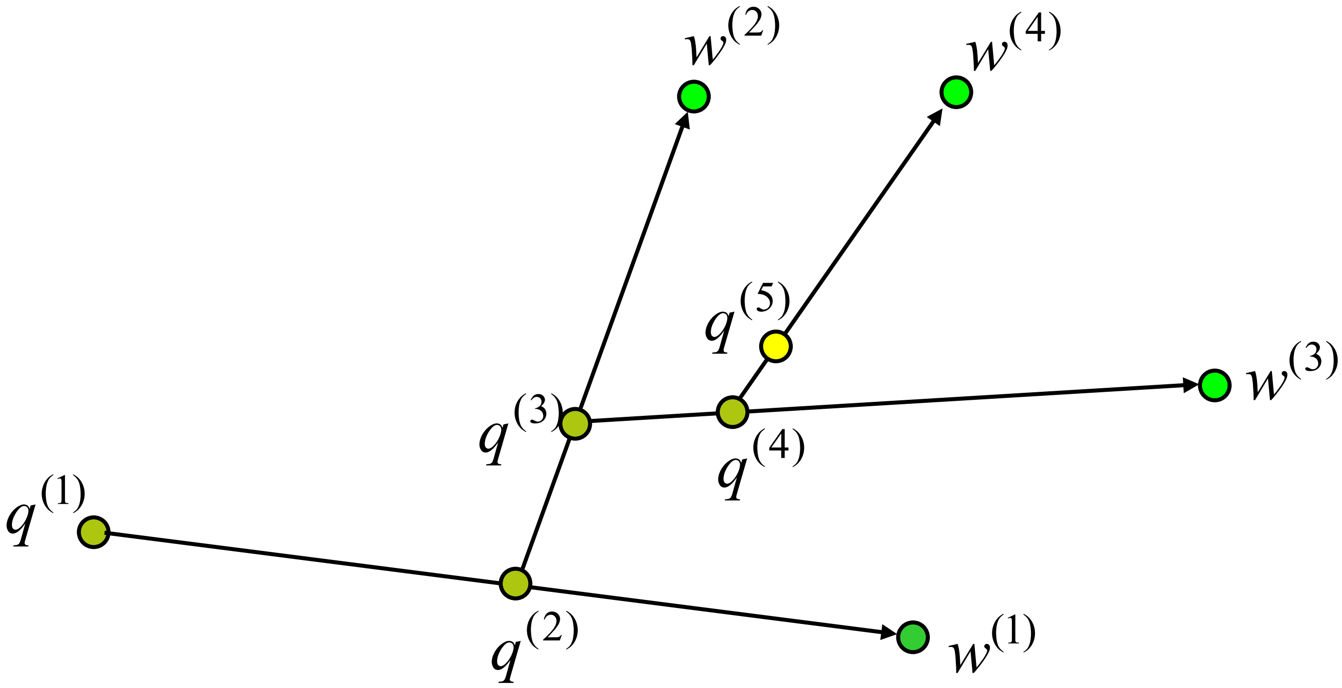
1. Set flows of all links equal to 0
2. Determine link costs based on the link flows
   - In first iteration use free flow travel times
3. Perform an assignment (AON)
4. Determine link flows
5. Return to step 2

Different approaches for step 4

- Naive: Repeat again and again
- Simple: Average flows with previous results (*1/n*)
- Smart: Average with optimal step sizes

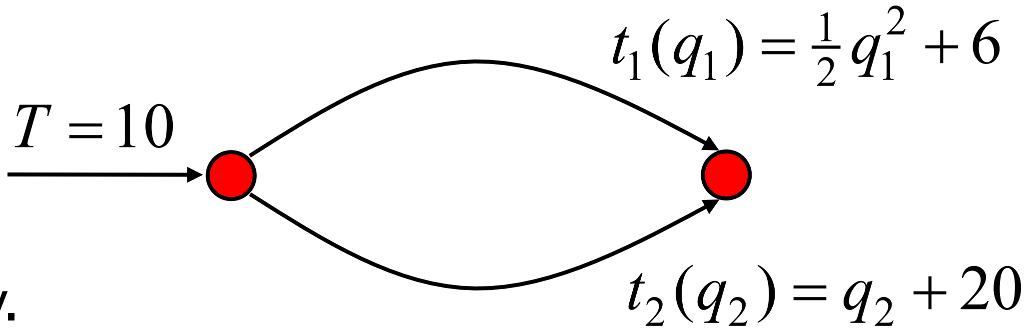
=> Don't
=> MSA
=> Frank-Wolfe

Illustration MSA algorithm



MSA: Example

Network



Find DUE iteratively.

q_1	q_2	t_1	t_2	w_1	w_2	α
0	0	6	20	10	0	1
10	0	56	20	0	10	1/2
5	5	18.5	25	10	0	1/3
6.7	3.3	28.2	23.3	0	10	1/4
5	5	18.5	25	10	0	1/5
6	4	24	24	-	-	-

$$q_a^{(i+1)} = q_a^{(i)} + \tfrac{1}{i}(w_a^{(i)} - q_a^{(i)})$$

$$\begin{pmatrix} q_1 \\ q_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} + \frac{1}{1} \cdot \left[\begin{pmatrix} 10 \\ 0 \end{pmatrix} - \begin{pmatrix} 0 \\ 0 \end{pmatrix} \right] = \begin{pmatrix} 10 \\ 0 \end{pmatrix}$$

$$\begin{pmatrix} q_1 \\ q_2 \end{pmatrix} = \begin{pmatrix} 10 \\ 0 \end{pmatrix} + \frac{1}{2} \cdot \left[\begin{pmatrix} 0 \\ 10 \end{pmatrix} - \begin{pmatrix} 10 \\ 0 \end{pmatrix} \right] = \begin{pmatrix} 5 \\ 5 \end{pmatrix}$$

$$\begin{pmatrix} q_1 \\ q_2 \end{pmatrix} = \begin{pmatrix} 5 \\ 5 \end{pmatrix} + \frac{1}{3} \cdot \left[\begin{pmatrix} 10 \\ 0 \end{pmatrix} - \begin{pmatrix} 5 \\ 5 \end{pmatrix} \right] = \begin{pmatrix} 6\frac{2}{3} \\ 3\frac{1}{3} \end{pmatrix}$$

$$\vdots$$

$$\vdots$$

3.3

DUE: MSA and convergence

Convergence criteria

- Number of iterations
- Equality of path costs
- Successive link flows (FA): $\left(q^{i} - q^{i-1}\right) / q^{i-1} < d$
- Duality gap $\sum_{a} t_a^i \cdot q_a^i - \sum_{o} \sum_{d} T_{od} \cdot \tau_{od}^i$

  i.e. total travel time based on links minus total travel time based on (latest) shortest paths

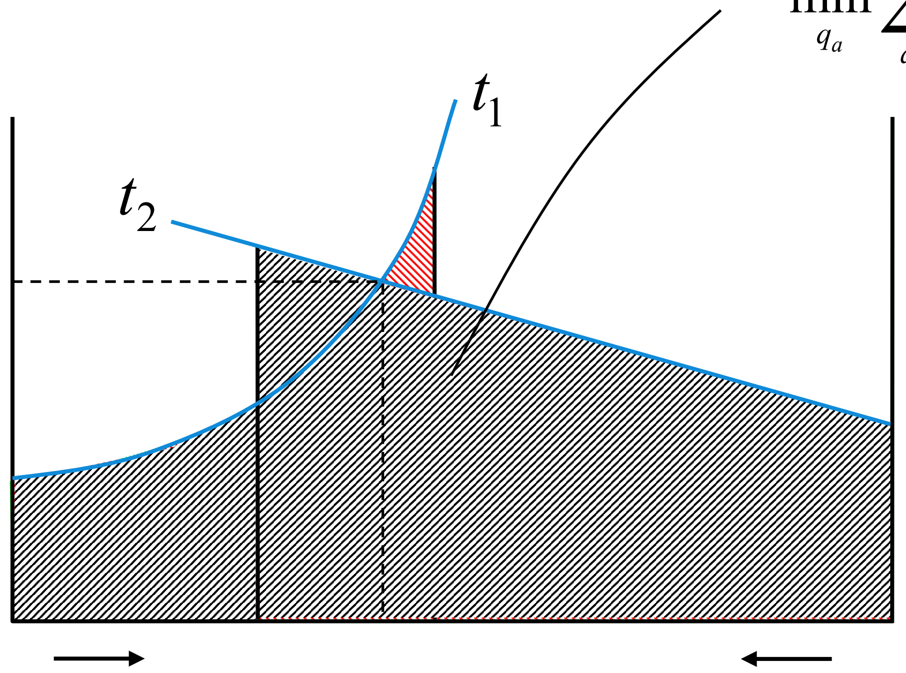
3.4

DUE: Mathematical formulation

DUE assignment: Mathematical formulation

What is the objective function?

$$\min_{q_a} \sum_a \int_{x=0}^{q_a} t_a(x)dx$$



Mathematical programming formulation

$$\min_{q_a} \sum_a \int_{x=0}^{q_a} t_a(x)dx$$

subject to:

$$\sum_r T_{ijr} = T_{ij} \quad \forall i,j$$ (flow conservation)

$$q_a = \sum_i \sum_j \sum_r \alpha_{ijr}^a T_{ijr} \quad \forall a$$ (definition)

$$T_{ijr} \geq 0 \quad \forall i,j,r$$ (non negativity)

→ non-linear programming problem

Everything in time?

People do not only decide on travel times,
but may also take other factors into account:

- fuel costs
- toll costs
- ... etc

How to take these factors into account?

Replace $t_a(q_a)$ with <u>generalized costs</u> $c_a(q_a)$

For example, $c_a(q_a) = \eta \cdot t_a(q_a) + \kappa_a$

value-of-time (VoT) (η), travel time ($t_a(q_a)$), toll (κ_a)

$$\Rightarrow \min_{q_a} \sum_a \int_{x=0}^{q_a} c_a(x)dx$$

3.5

DUE: Algorithm: Frank-Wolfe

Frank-Wolfe algorithm

A nonlinear programming problem can be solved using a *steepest descent algorithm*.

The *convex combinations algorithm*, also known as the *Frank-Wolfe algorithm* (1956), is a common (formal!) algorithm to solve the DUE assignment problem.

Main principle:

- Find an initial feasible solution
- Linearise the objective function
- Find a new intermediate solution
- Determine a new solution in the direction of the intermediate solution
- Iterate until the new solution does not change anymore

Frank-Wolfe algorithm 1/4

$$\min_{q_a} Z = \sum_a \int_{x=0}^{q_a} t_a(x)dx \qquad (1)$$

subject to:

$$\sum_r T_{ijr} = T_{ij} \quad \forall i,j \qquad (2)$$

$$q_a = \sum_i \sum_j \sum_r \alpha_{ijr}^a T_{ijr} \quad \forall a \qquad (3)$$

$$T_{ijr} \geq 0 \quad \forall i,j,r \qquad (4)$$

Step 1: Find an initial feasible solution (iteration 1)

We have to find a solution that satisfies constraints (2), (3), and (4).

An AON assignment yields a feasible solution $q^{(1)}$

Frank-Wolfe algorithm 2/4

$$\min_{q_a} Z = \sum_a \int_{x=0}^{q_a} t_a(x)dx \qquad (1)$$

subject to:

$$\sum_r T_{ijr} = T_{ij} \quad \forall i,j \qquad (2)$$

$$q_a = \sum_i \sum_j \sum_r \alpha_{ijr}^a T_{ijr} \quad \forall a \qquad (3)$$

$$T_{ijr} \geq 0 \quad \forall i,j,r \qquad (4)$$

Step 2: Linearise the objective function (iteration i)

$$\tilde{Z}(w^{(i)}) = Z(q^{(i)}) + \sum_a \frac{\partial Z(q^{(i)})}{\partial q_a}(w_a - q_a^{(i)})$$

$$= Z(q^{(i)}) + \sum_a t_a(q_a^{(i)})(w_a - q_a^{(i)})$$

(first-order expansion Taylor polynomial)

Frank-Wolfe algorithm 3/4

$$\min_{q_a} Z = \sum_a \int_{x=0}^{q_a} t_a(x)dx \qquad (1)$$

subject to:

$$\sum_r T_{ijr} = T_{ij} \quad \forall i,j \qquad (2)$$

$$q_a = \sum_i \sum_j \sum_r \alpha_{ijr}^a T_{ijr} \quad \forall a \qquad (3)$$

$$T_{ijr} \geq 0 \quad \forall i,j,r \qquad (4)$$

Step 3: Solve the linearised problem (iteration i)

$$\min_{w_a^{(i)}} \tilde{Z}(w^{(i)}) = \min_{w_a^{(i)}} \left\{ Z(q^{(i)}) + \sum_a t_a(q_a^{(i)})(w_a - q_a^{(i)}) \right\}$$

An AON assignment using $t_a(q_a^{(i)})$ yields a feasible solution w_a

$$\min_{w_a^{(i)}} \sum_a t_a(q_a^{(i)}) w_a^{(i)}$$

subject to (2), (3), and (4)

Frank-Wolfe algorithm 4/4

$$\min_{q_a} Z = \sum_a \int_{x=0}^{q_a} t_a(x)dx \qquad (1)$$

subject to:

$$\sum_r T_{ijr} = T_{ij} \quad \forall i,j \qquad (2)$$

$$q_a = \sum_i \sum_j \sum_r \alpha_{ijr}^a T_{ijr} \quad \forall a \qquad (3)$$

$$T_{ijr} \geq 0 \quad \forall i,j,r \qquad (4)$$

Step 4: Find the new solution (iteration i)

$$\tilde{\alpha}^{(i)} = \arg\min_{0 \leq \alpha \leq 1} Z(q^{(i)} + \alpha(w^{(i)} - q^{(i)}))$$

α = optimal stepsize

$$\Rightarrow \quad q^{(i+1)} = q^{(i)} + \tilde{\alpha}^{(i)}(w^{(i)} - q^{(i)})$$

Frank-Wolfe algorithm

Step 1: $i := 1$. Set $q_a^{(i)} = 0$ (assume empty network)

Step 2: Perform a shortest-path AON assignment based on $t_a^{(i)} = t_a(q_a^{(i)})$ yielding link flows $w_a^{(i)}$

Step 3: Compute new solution $q_a^{(i+1)} = q_a^{(i)} + \alpha^{(i)}(w_a^{(i)} - q_a^{(i)})$

with $\tilde{\alpha}^{(i)} = \arg\min_{0 \le \alpha \le 1} Z(q^{(i)} + \alpha(w^{(i)} - q^{(i)}))$

Step 4: If no convergence yet, set $i := i+1$ and return to Step 2.

N.B. Using $\alpha^{(i)} = 1/i$ is called:
Method of Successive Averages (MSA)
(see earlier slides)

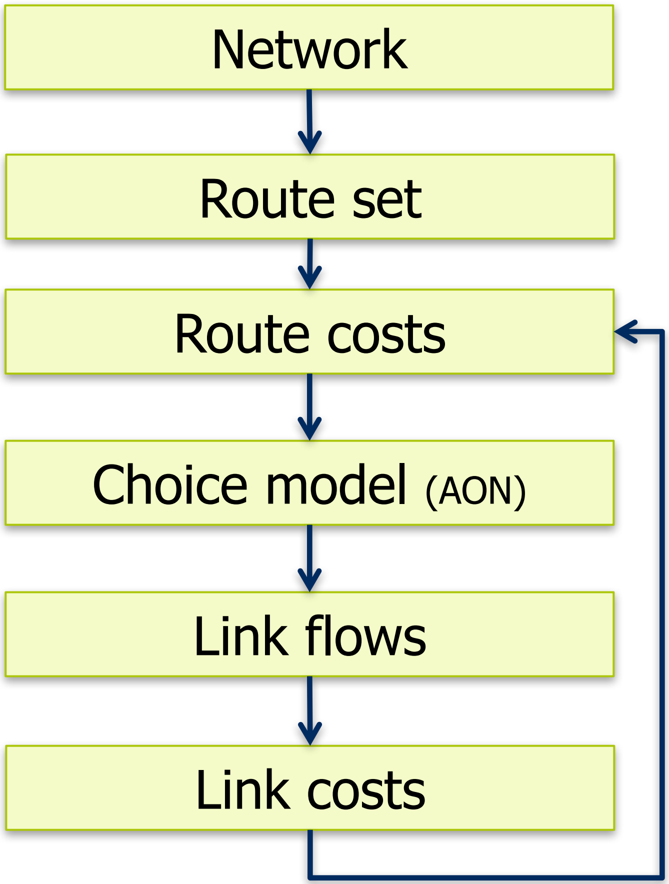
3.6

DUE: Route based approach

Link based versus route based

- Previous slides all referred to a link based formulation
- i.e. repetitive tree searches
- What if you have a set of possible routes?
- Then there's no need to search for routes at each MSA-step!

Route based approach: Generic procedure



Generic solution scheme
Route flow averaging

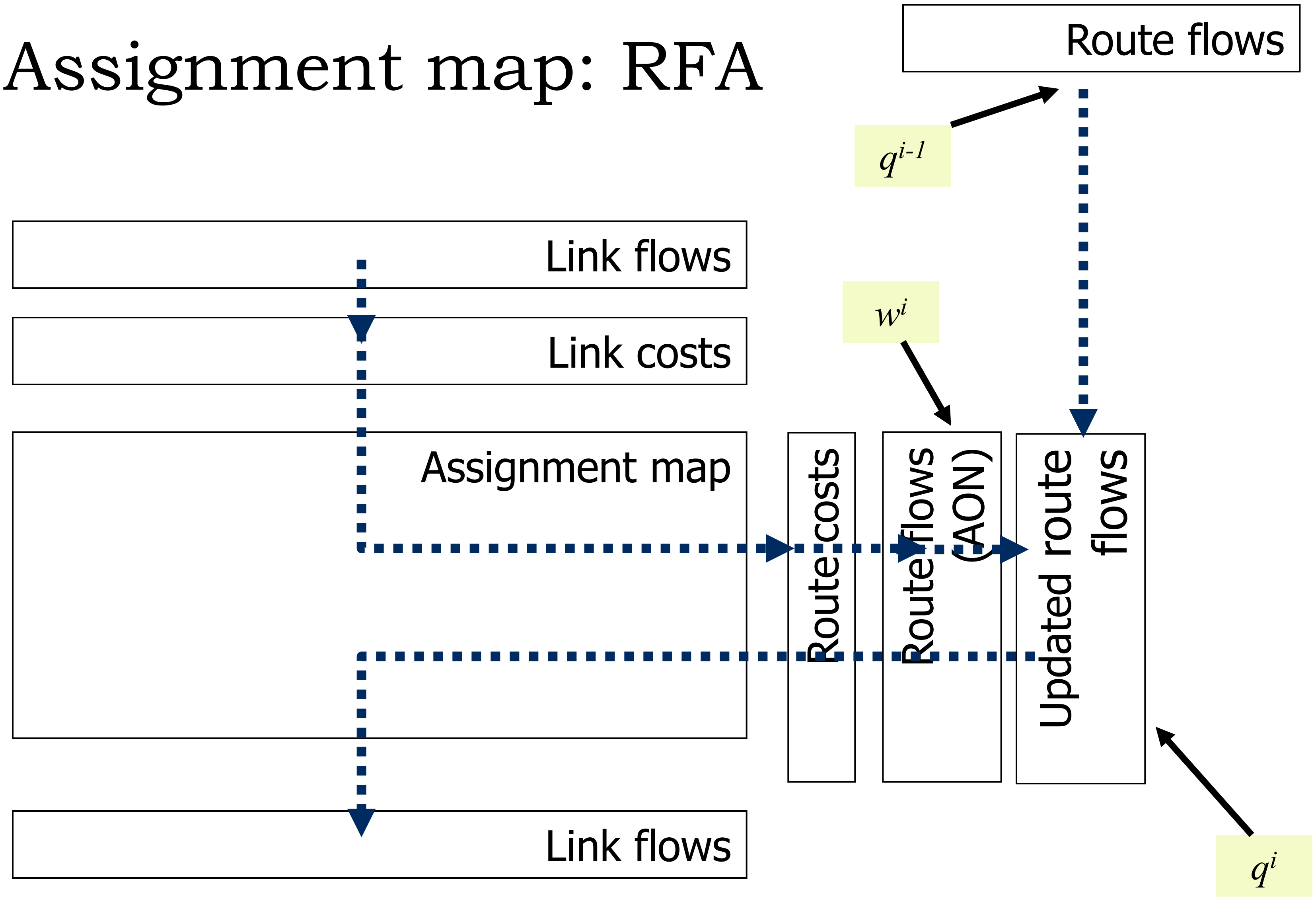
1. Specify routes/choice sets
2. Calculate path costs
3. Assign OD (AON)=> route flows
4. Recalculate route flows (MSA)

$$q_r^{\,i} = q_r^{\,i-1} + \left(w_r^{\,i} - q_r^{\,i-1} \right) / i$$

where $w_r^{\,i}$ = route flow of the intermediate solution

5. Calculate link flows
6. Check convergence
7. Go to step 2 or stop

Assignment map: RFA



Benefits of a route-based approach

- Full control of routes that are used
- Freedom in route choice modelling (see SUE)
- Overlapping routes can explicitly be dealt with
- Reduced computational efforts in equilibrium assignment
- Suitable for all kind of network concepts e.g. multimodal networks

- So, why isn't route-based assignment used in practice?

3.7

SUE: Stochastic user equilibrium assignment

Stochastic user equilibrium

- Combination of two concepts
- Congestion: travel time is function of flow
  => equilibrium modelling
- Perception: travellers consider a set of routes
  => route choice
- Adjustment Wardrop
  All travellers choose their optimal route, such that no traveller can improve his/her **perceived** travel time by unilaterally changing routes.

Stochastic equilibrium assignment

1. Link travel times t_a

$$t_a = t_a(q_a)$$

2. Route choice proportions ϕ_{ijr}

$$\phi_{ijr} = \phi_{ijr}(\Theta, t_{ijr})$$

3. Route travel times t_{ijr}

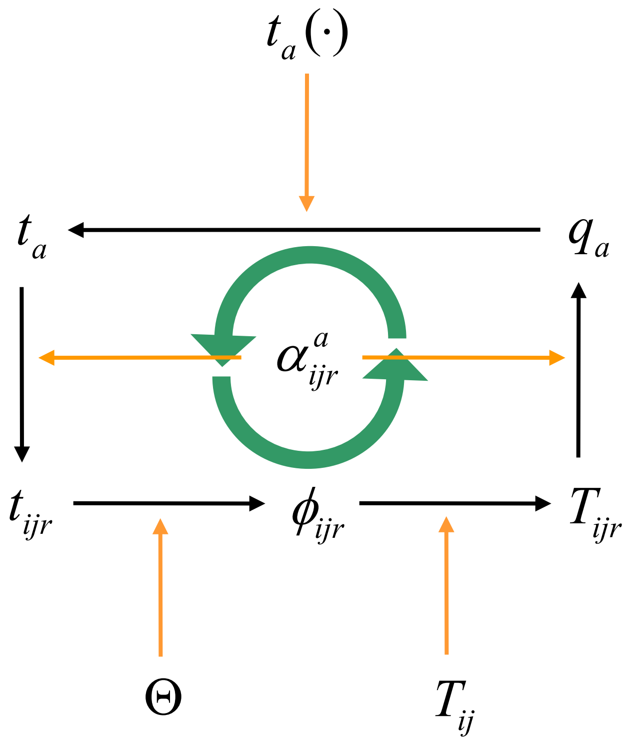
$$t_{ijr} = \sum_a \alpha_{ijr}^a t_a$$

4. Route flows T_{ijr}

$$T_{ijr} = \phi_{ijr} T_{ij}$$

5. Link flows q_a

$$q_a = \sum_i \sum_j \sum_r \alpha_{ijr}^a T_{ijr}$$



Generic solution scheme
Route flow averaging

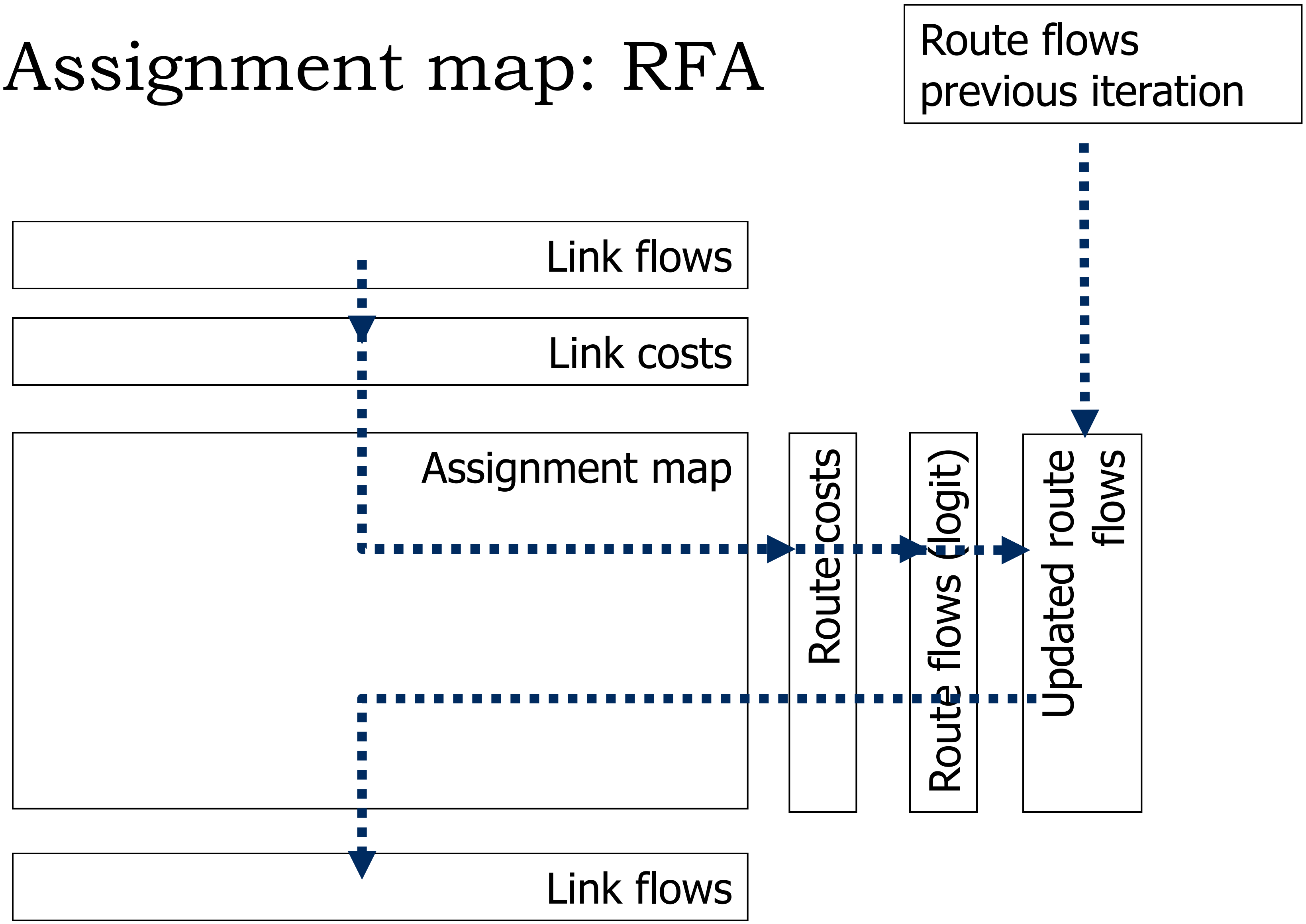
1. Specify paths/choice sets
2. Calculate path costs
3. Assign OD (Logit, Probit)=> route flows
4. Recalculate route flows (MSA)

$$q_r^{\,i} = q_r^{\,i-1} + \left(w_r^{\,i} - q_r^{\,i-1}\right)/\,i$$

where $w_r^{\,i}$ = route flow of the intermediate solution

5. Calculate link flows
6. Check convergence
7. Go to step 2 or stop

Assignment map: RFA



Alternative solution scheme Link flow averaging

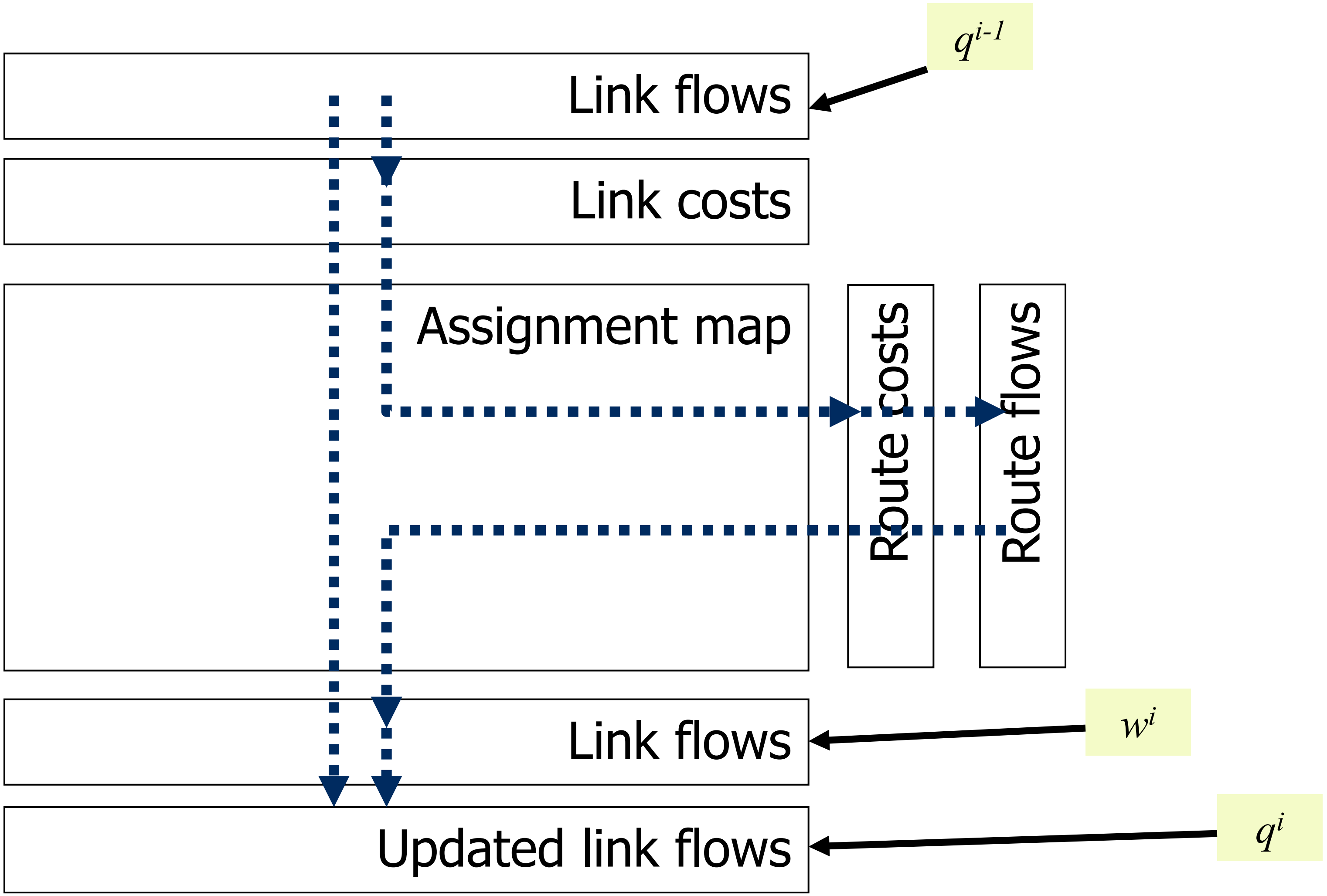
1. Specify paths/choice sets
2. Calculate path costs
3. Assign OD (Probit, Logit)=> route flows
4. Calculate link flows
5. Recalculate link flows (MSA)

   $$q_a^{\,i} = q_a^{\,i-1} + \left(w_a^{\,i} - q_a^{\,i-1}\right)/\,i$$

   where $w_a^{\,i}$ = link flow of the intermediate solution

6. Check convergence
7. Go to step 2 or stop

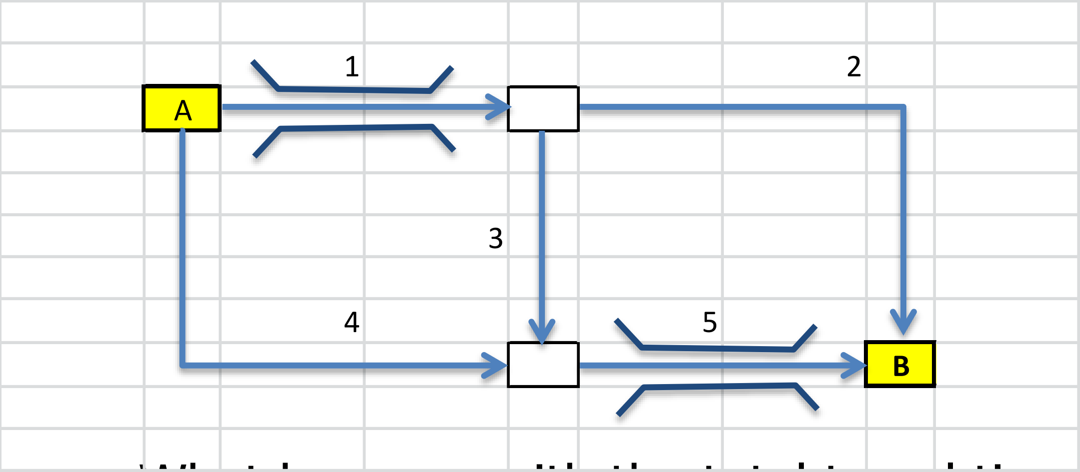
Assignment map: LFA



3.8

Braess's paradox

Braess's paradox



- What happens with the total travel time if link 3 is open for traffic?
- Explore spreadsheet Braess (on Blackboard)
- Check internet, e.g. YouTube!

4.

Practical issues

Practical topics

- Trucks
- Convergence speed DUE and SUE
- Duality gap SUE
- Frank-Wolfe or MSA?
- Values for parameters in BPR
- Where's the congestion?

Trucks

Three options

- Sum the OD-matrices of car and truck into OD-matrix vehicles or using a PCU-value
- Assign trucks before performing equilibrium assignment, e.g. using multiple routing, and use the flows as a preload (PCU!)
- Assign trucks and cars simultaneously (again using a PCU-value), i.e. multi-user class assignment

Convergence speed DUE and SUE

- In an equilibrium assignment you distribute traffic over a set of routes

- In a DUE you have a single route per MSA-step, in a SUE you have multiple routes per MSA-step
  => SUE needs less MSA-steps

- However, if you use a Probit for SUE, you need iterations for a single MSA-step
  => SUE with Probit takes more computation time
  => SUE with Logit is faster than DUE

Duality gap SUE

- Duality gap in words:
  total travel time based on links minus total travel time based on (latest) shortest paths

- For DUE the duality gap should become zero (Wardrop principle)

- In a SUE travellers opt routes that are longer but they perceive to be shortest
  => Total travel time for SUE is higher than for DUE
  => Duality gap > 0

Frank-Wolfe or MSA

- Frank-Wolfe algorithm is a generic mathematical tool
- Theoretically it is only justified if the travel time on a link is a function of the flow on the link
  => so what about intersections?
- MSA is a pragmatic approach, which proves to be rather robust

Values for parameters in BPR

- BPR-function: $t_a(q_a) = t_a^0\left(1+\alpha\left(\dfrac{q_a}{C_a}\right)^{\beta}\right)$
- Commonly mentioned values: $\alpha = 0.15$ and $\beta = 4$
- However, function differs per road type:
  e.g. 0.15 is used for freeways, for regional and urban roads higher values are more suitable

Where's the congestion?

- Net result of assignment: network with flows
- Common unit for analysis: flow-capacity (q/c) ratio
- For which q/c-ratio there is congestion?
- Practice:
  q/c-ratio > 0.85: congestion
  (N.B. q represents average flow, thus q/c=1 implies 50% congestion)
- Where's the queue?