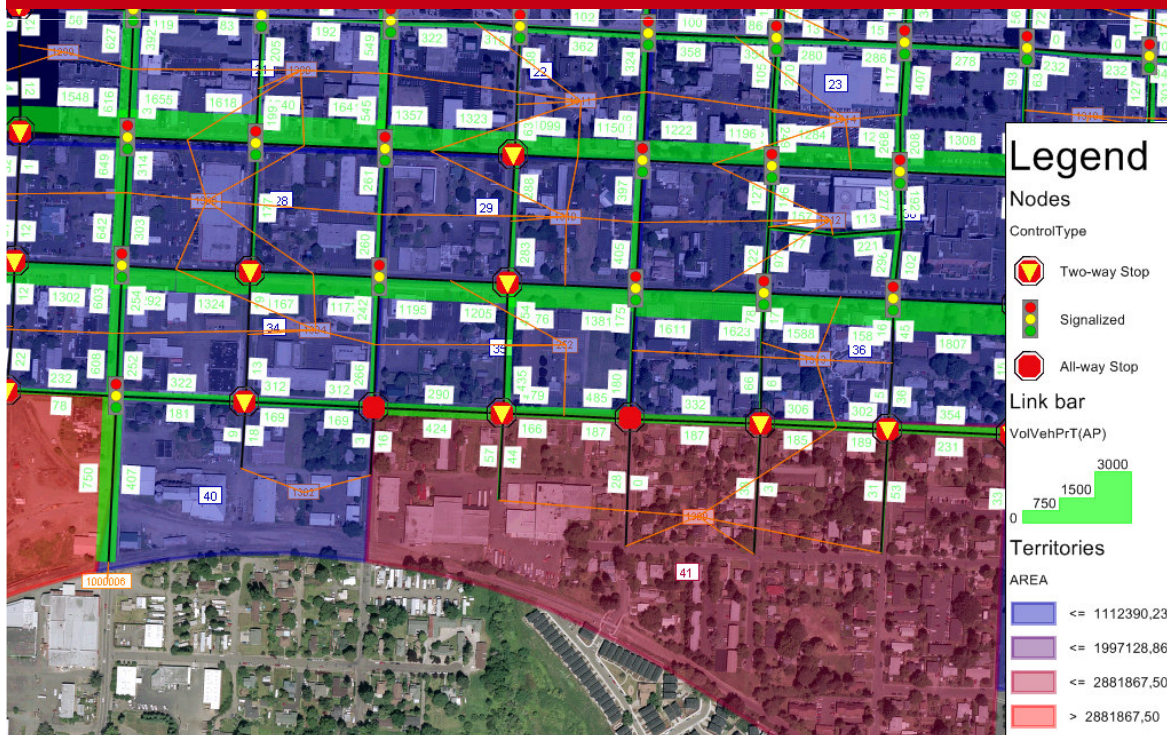


9th UK PTV Vision User Group Meeting

LUCE – new highway assignment ... and other ways to make VISUM faster

Klaus Nökel, PTV AG, Karlsruhe



Speed-up through Multi-Threading

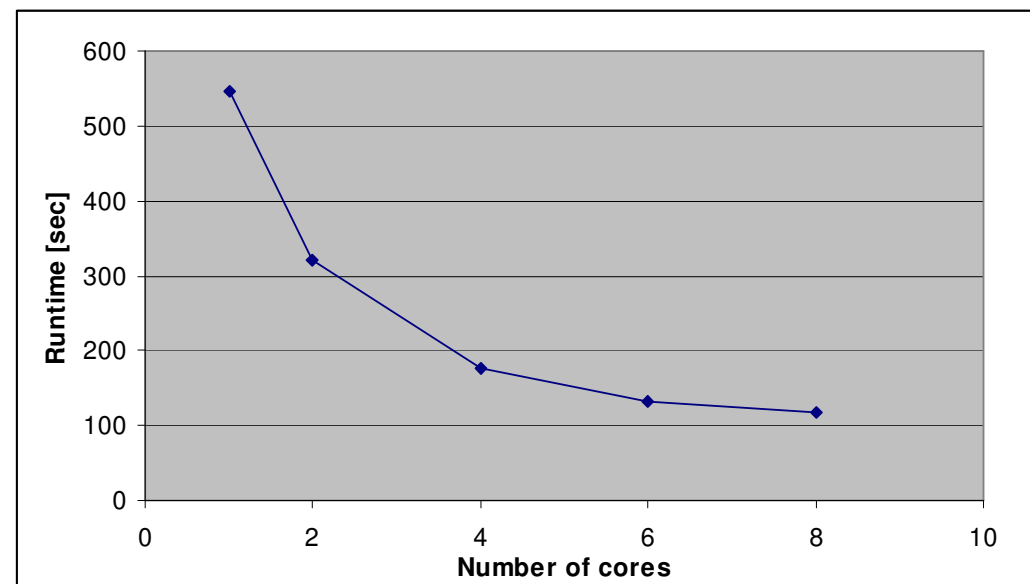
Computationally intense operations speeded up

- > **now:** Equilibrium Lohse and timetable-based PuT assignment
- > next release: most other assignments, skimming, demand model operations

Supports all modern multi-core processors

Examples

- > Lohse equilibrium assignment on two cores ~ 60% of sequential runtime
- > Timetable-based assignment on eight cores 4 – 6 times faster (see right for KA.ver performance)



Equilibrium Auto Assignment

Speed improvements in VISUM 10

- > Results stored as double precision numbers
→ better gaps can be reached at all
- > faster shortest-path search in very large networks
- > more intelligent control over inner iterations

Speed improvements in VISUM 11

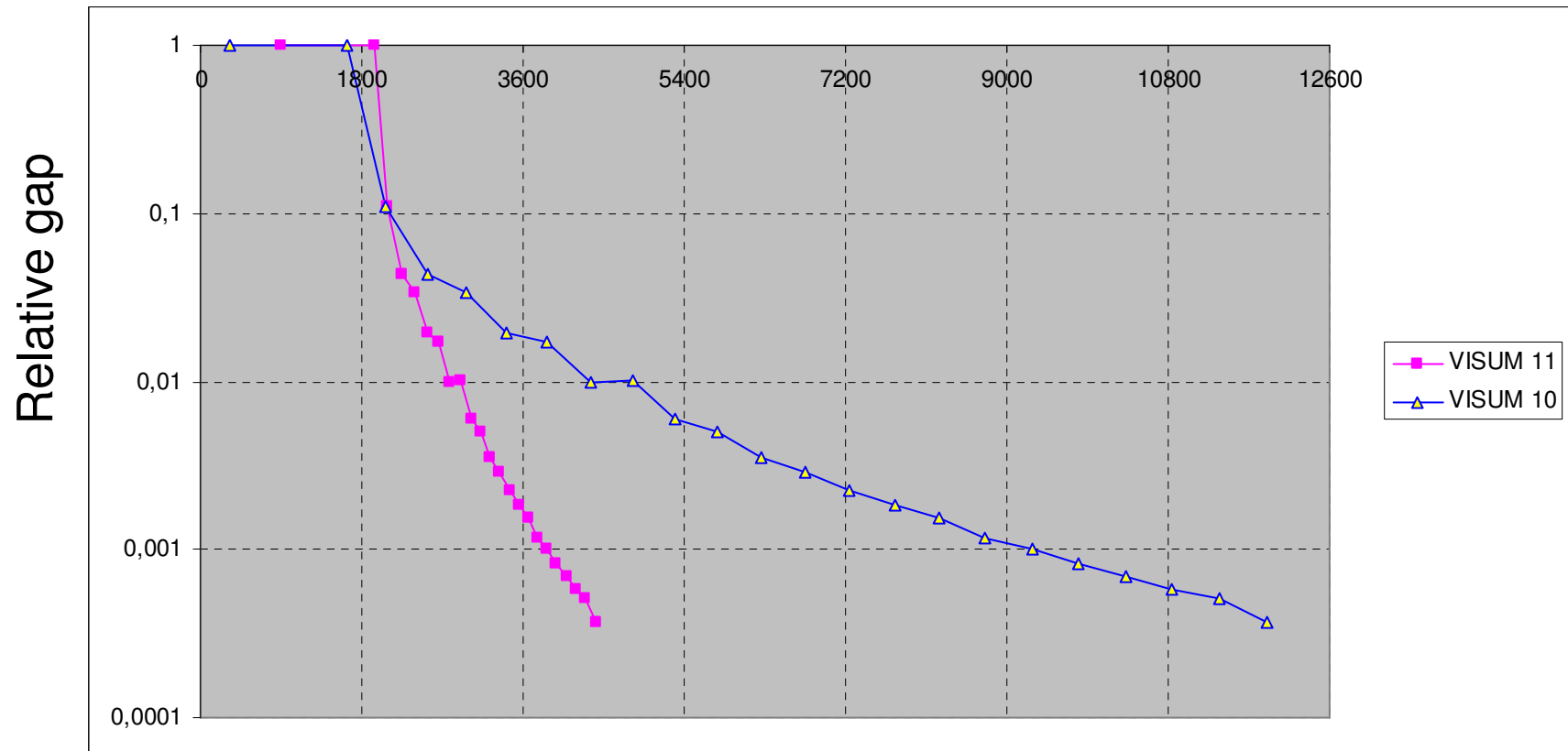
- > Focus on multi-threading, but unfortunately not applicable to eq assignment
- > Instead tuning of very basic operations, e.g. for path storage and retrieval
- > Comes at a price: more memory (typically + 20-30%)
- > Also: Work on completely new algorithm LUCE

Speed improvements in VISUM 11.5 - planned

- > More tuning of basic operations, e.g. impedance calculation
- > Finish LUCE

Current Equilibrium Auto Assignment - Runtime

Runtime in seconds on 2.66 GHz computer



Equilibrium Auto Assignment

Speed improvements in VISUM 10

- > Results stored as double precision numbers
→ better gaps can be reached at all
- > faster shortest-path search in very large networks
- > more intelligent control over inner iterations

Speed improvements in VISUM 11

- > Focus on multi-threading, but unfortunately not applicable to eq assignment
- > Instead tuning of very basic operations, e.g. for path storage and retrieval
- > Comes at a price: more memory (typically + 5-10%)
- > Also: Work on completely new algorithm LUCE

Speed improvements in VISUM 11.5 - planned

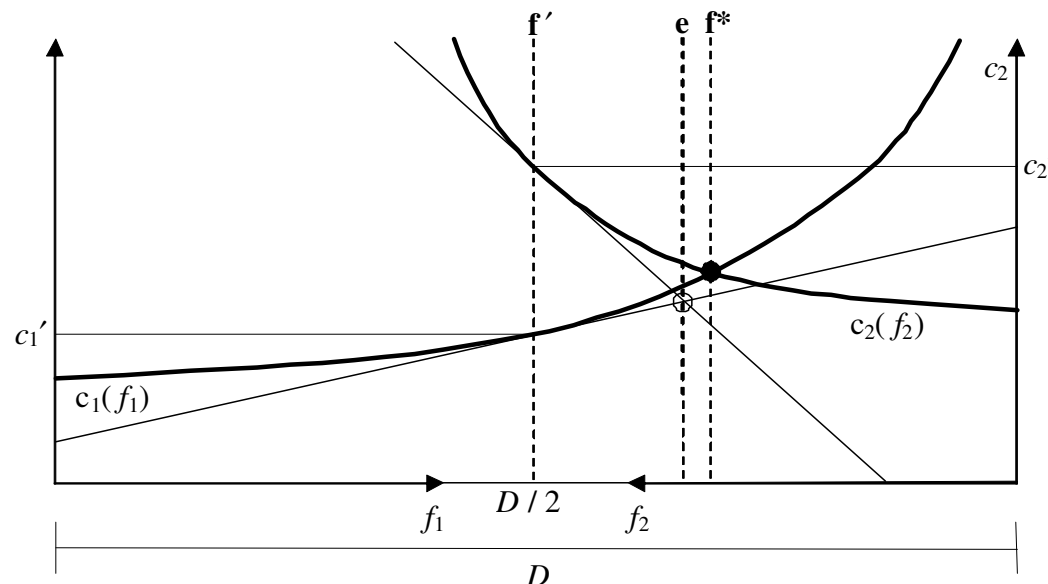
- > More tuning of basic operations, e.g. impedance calculation
- > Finish LUCE

LUCE (Guido Gentile 2008)

Origin-based Assignment

- > Treats all OD pairs for one O simultaneously
- > Achieves perfect proportionality of path volumes within one O
- > Loads richer path set (in congested networks)
- > Implicit path representation as bushes instead of explicit paths

Gradient Method

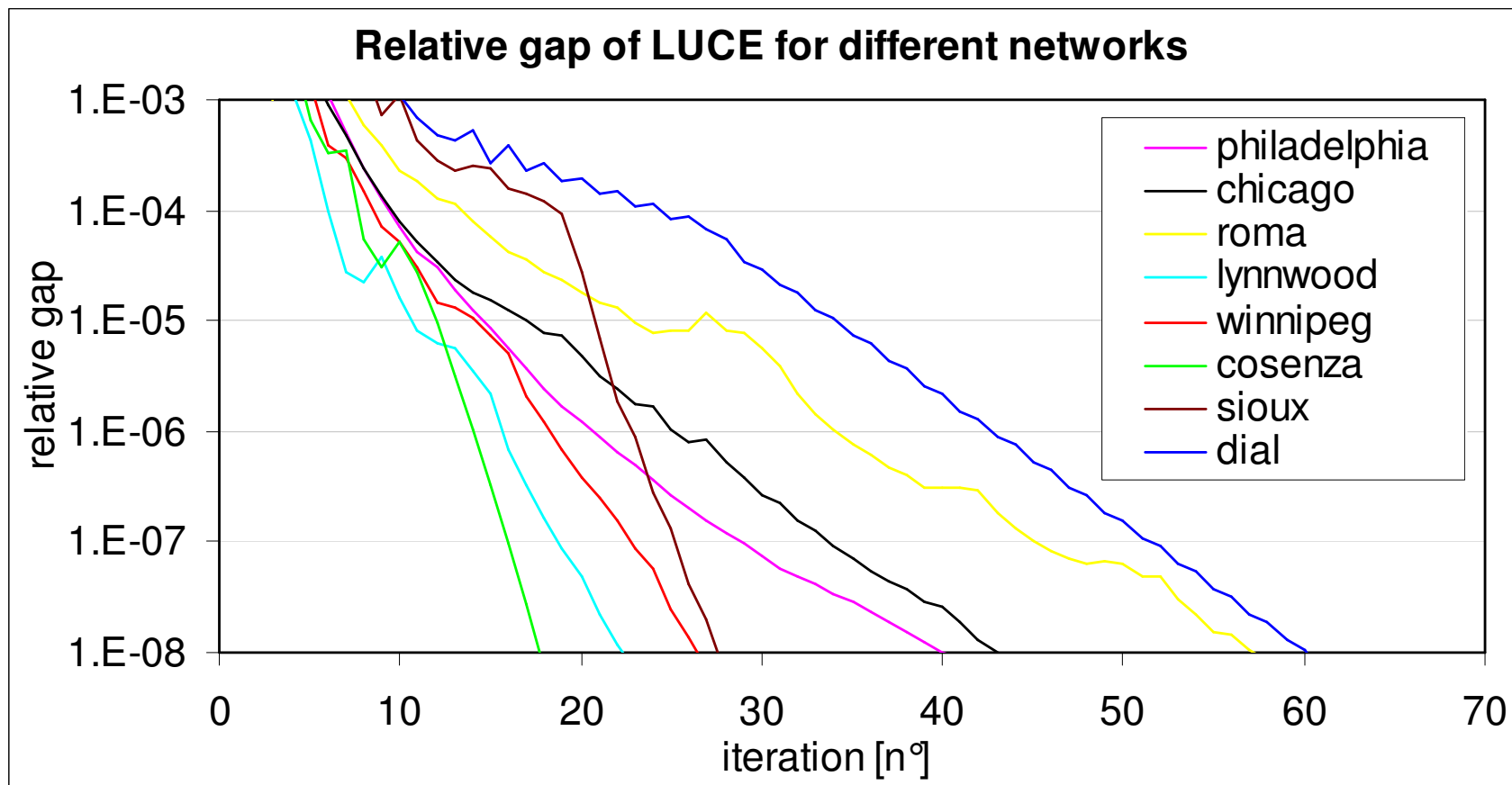


Gradient corresponds to
 Linearizing cost functions
 w.r.t. current solution
 → Linearized User Cost Equilibrium
 → LUCE

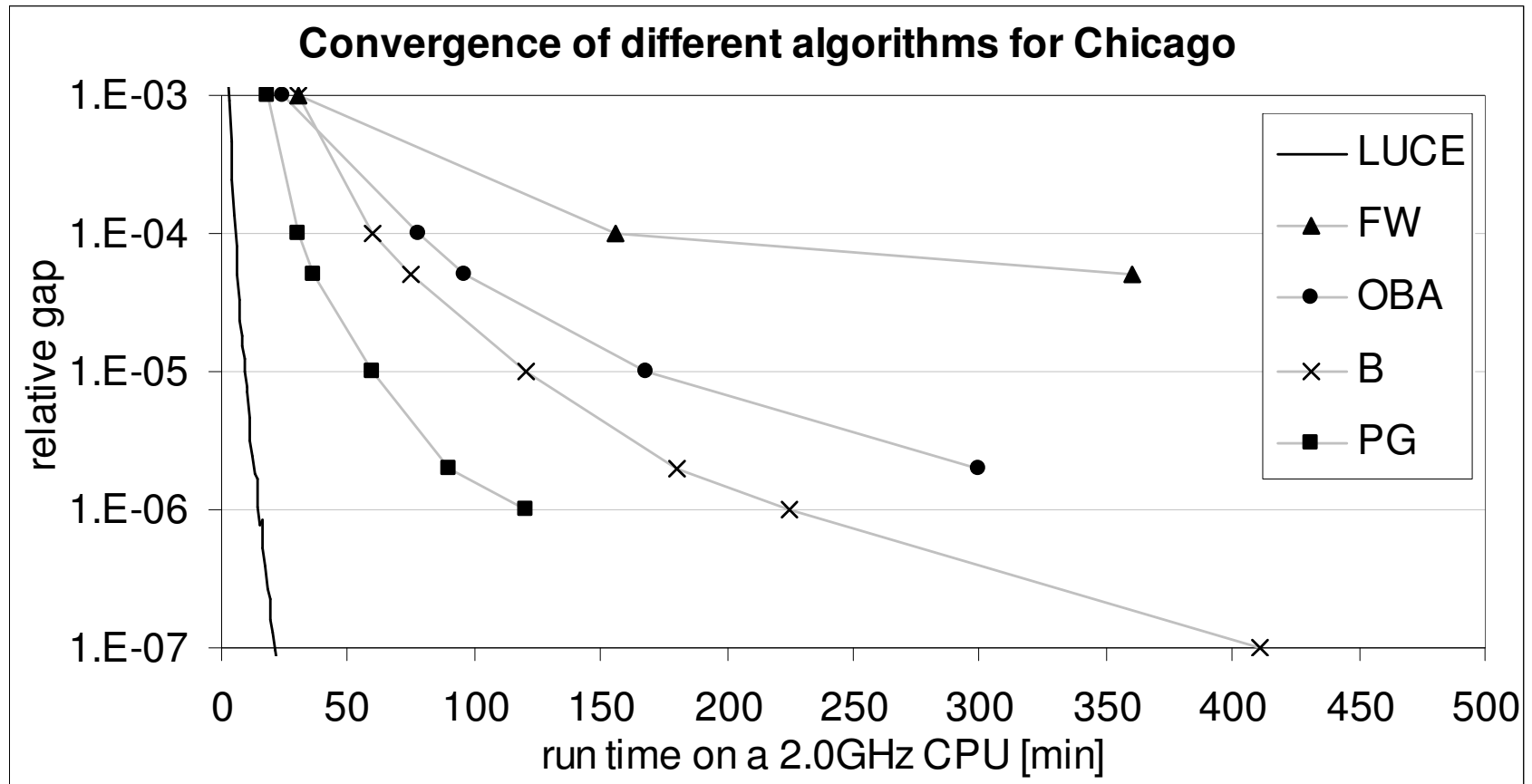
LUCE - Algorithm

```
initialize the solution flows to zero
for iteration  $k = 1$  to  $n$ :
  for each destination  $d \in Z$ :
    compute arc costs and their derivatives
    initialize or modify the current bush
    for each node  $i \neq d$  in the bush in topological order :
      compute the node average cost to  $d$  and its derivative
    reset the link and node flows to  $d$ 
    load on the origins the demand to  $d$ 
    for each node  $i \neq d$  in the bush in reverse topological order:
      recalc the "attractive" forward links towards  $d$ 
      and their shares
      propagate the link flows to the ToNode
    compute optimal step size for combining new and old flow pattern
    update link flows according to step size
```

LUCE - Results



LUCE - Results

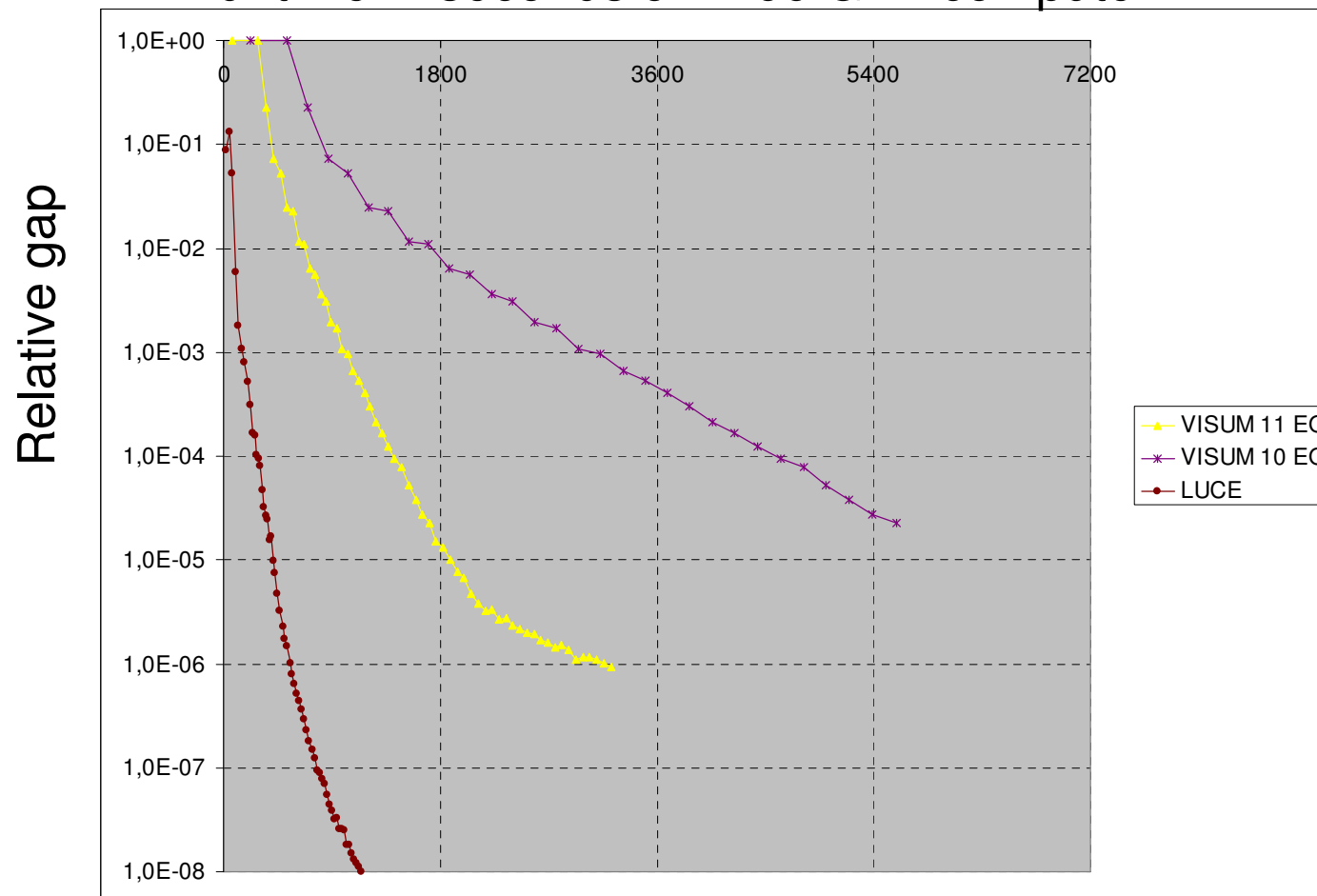


modified after Michael Florian

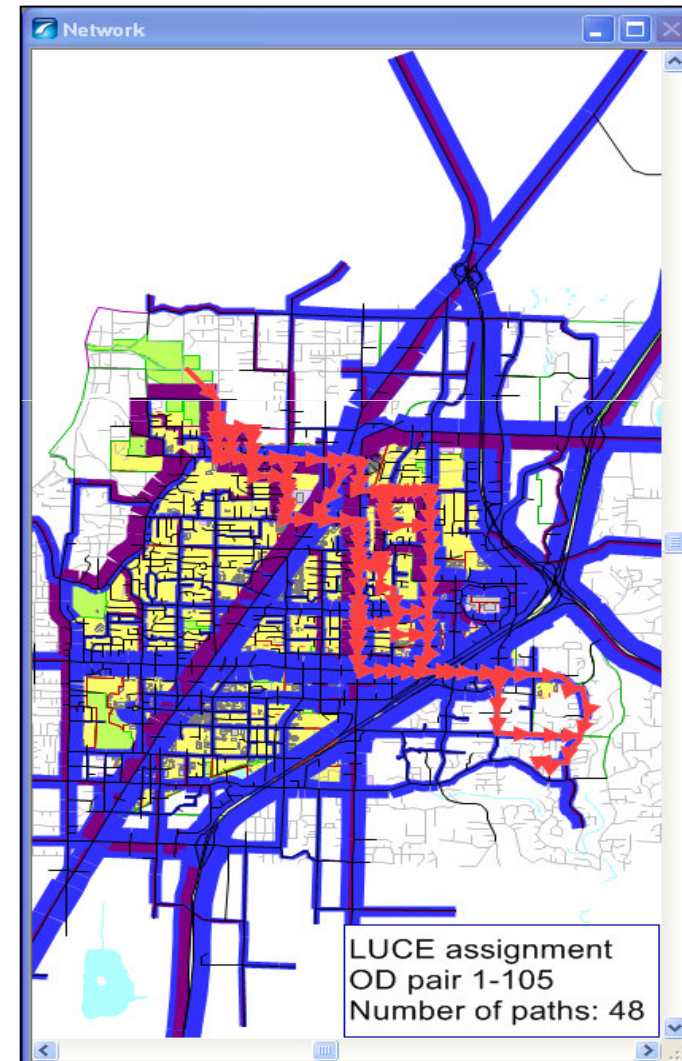
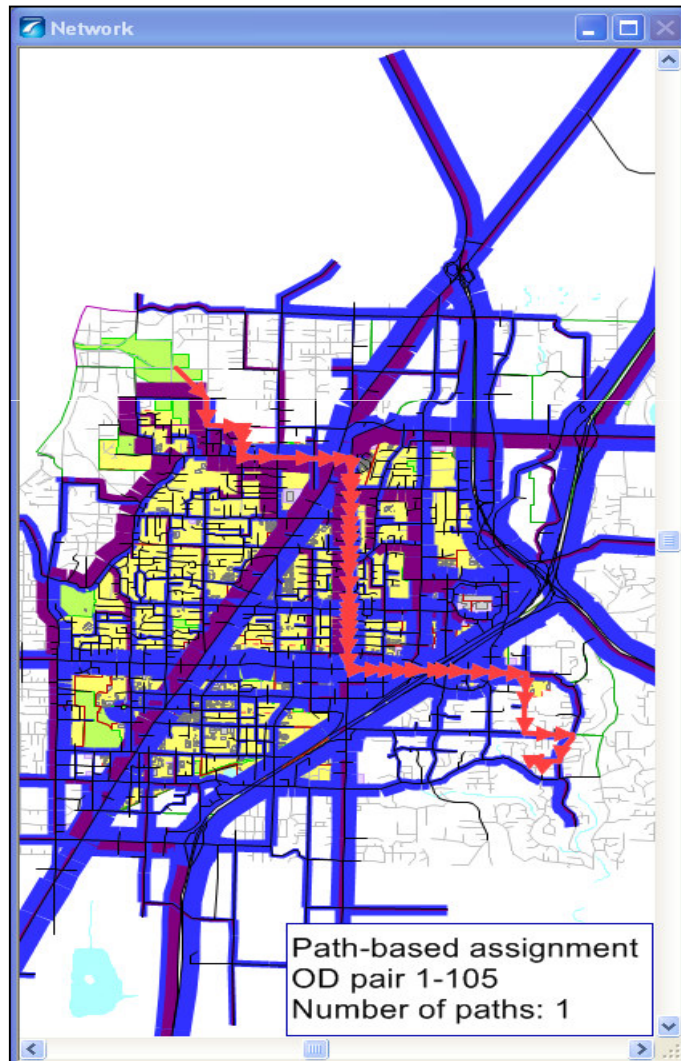
LUCE - Results

Compared to VISUM assignment on Chicago network

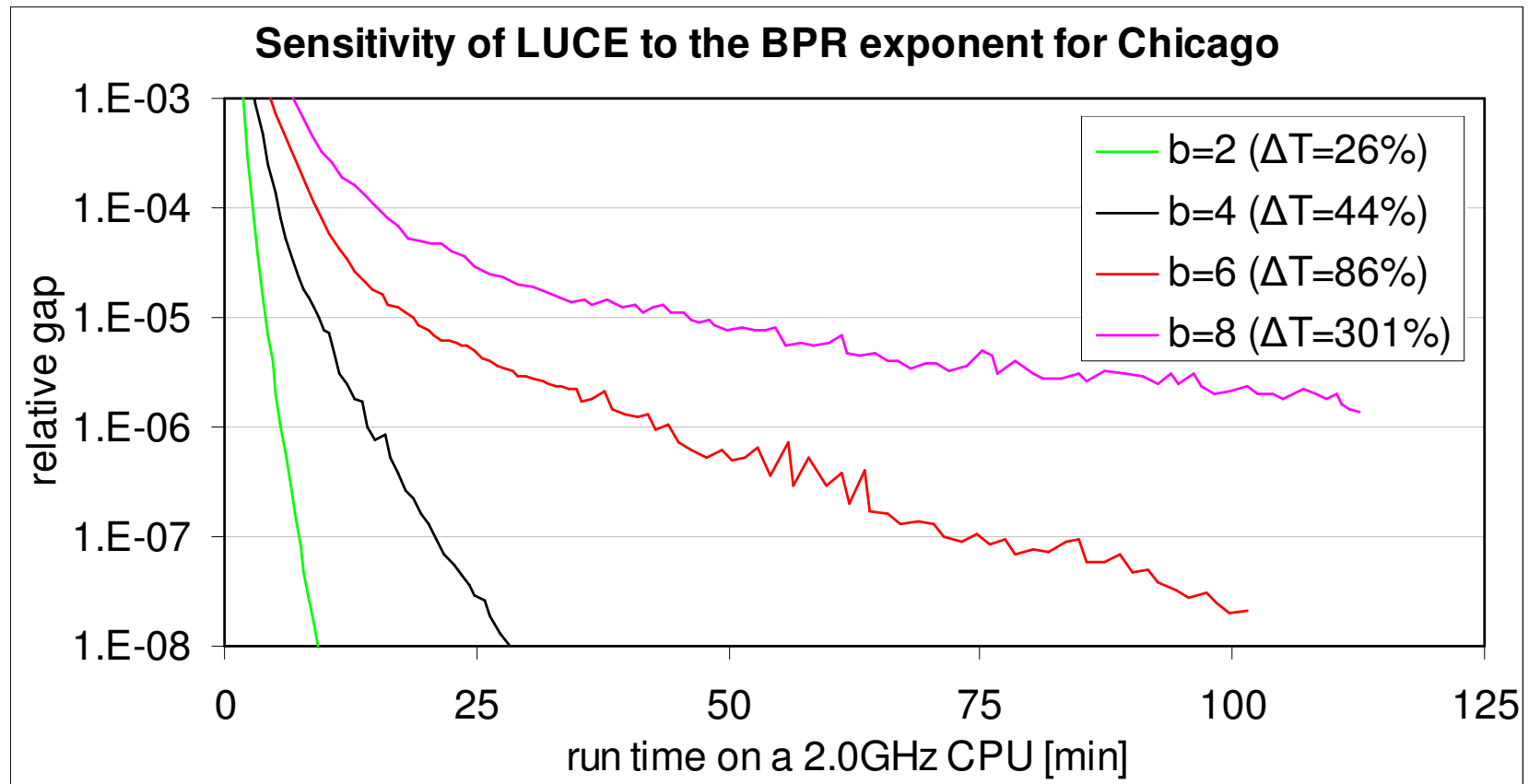
Runtime in seconds on 2.66 GHz computer



LUCE – richer path set loaded



LUCE - Results

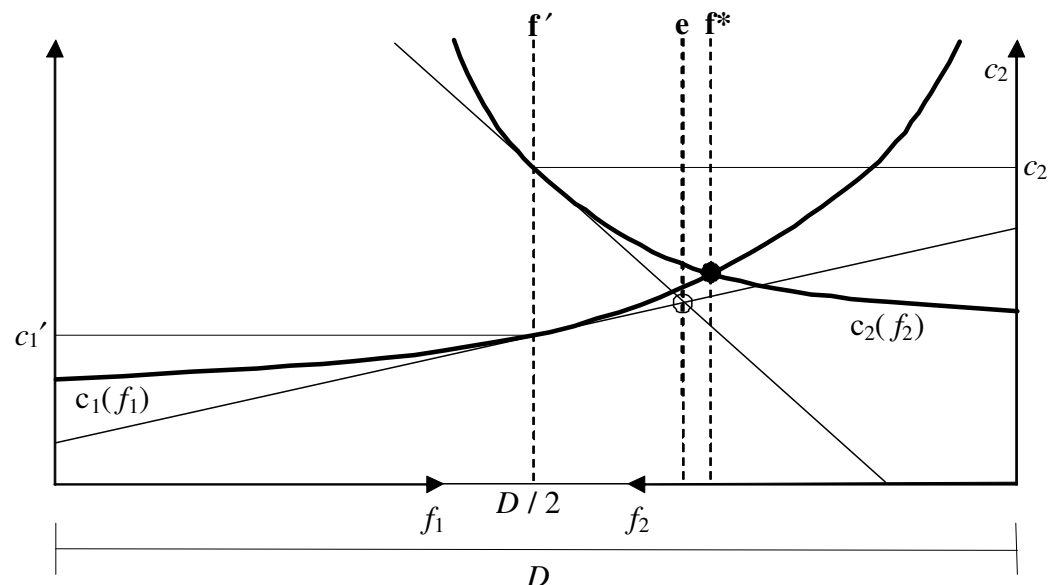


LUCE (Guido Gentile 2008)

Origin-based Assignment

- > Treats all OD pairs for one O simultaneously
- > Achieves perfect proportionality of path volumes within one O
- > Loads richer path set (in congested networks)
- > Implicit path representation as bushes instead of explicit paths

Gradient Method



Gradient corresponds to
 Linearizing cost functions
 w.r.t. current solution
 → Linearized User Cost Equilibrium
 → LUCE

LUCE – still work to do

Currently a prototype: What does that mean?

- > Explicit route storage is VISUM's „common exchange format“ between different assignment methods and different analysis methods
- > LUCE does not use explicit paths
- > Need to adapt analysis methods (skimming, select-link, matrix estimation) to implicit bush representation and bypass explicit routes
- > Save bushes after assignment and store in VER file → basis for warm start
- > Implementations expected within 2009
- > **CAVEAT:** results may change



Thank you ! Questions ?

www.ptv.de