Polynomial-time Construction of Contraction Hierarchies for Multi-criteria Objectives (ALENEX 2013)

University of Stuttgart
Germany

Stefan Funke
Sabine Storandt

Many Roads lead to Rome

Often, there are many alternative routes to reach a target ...

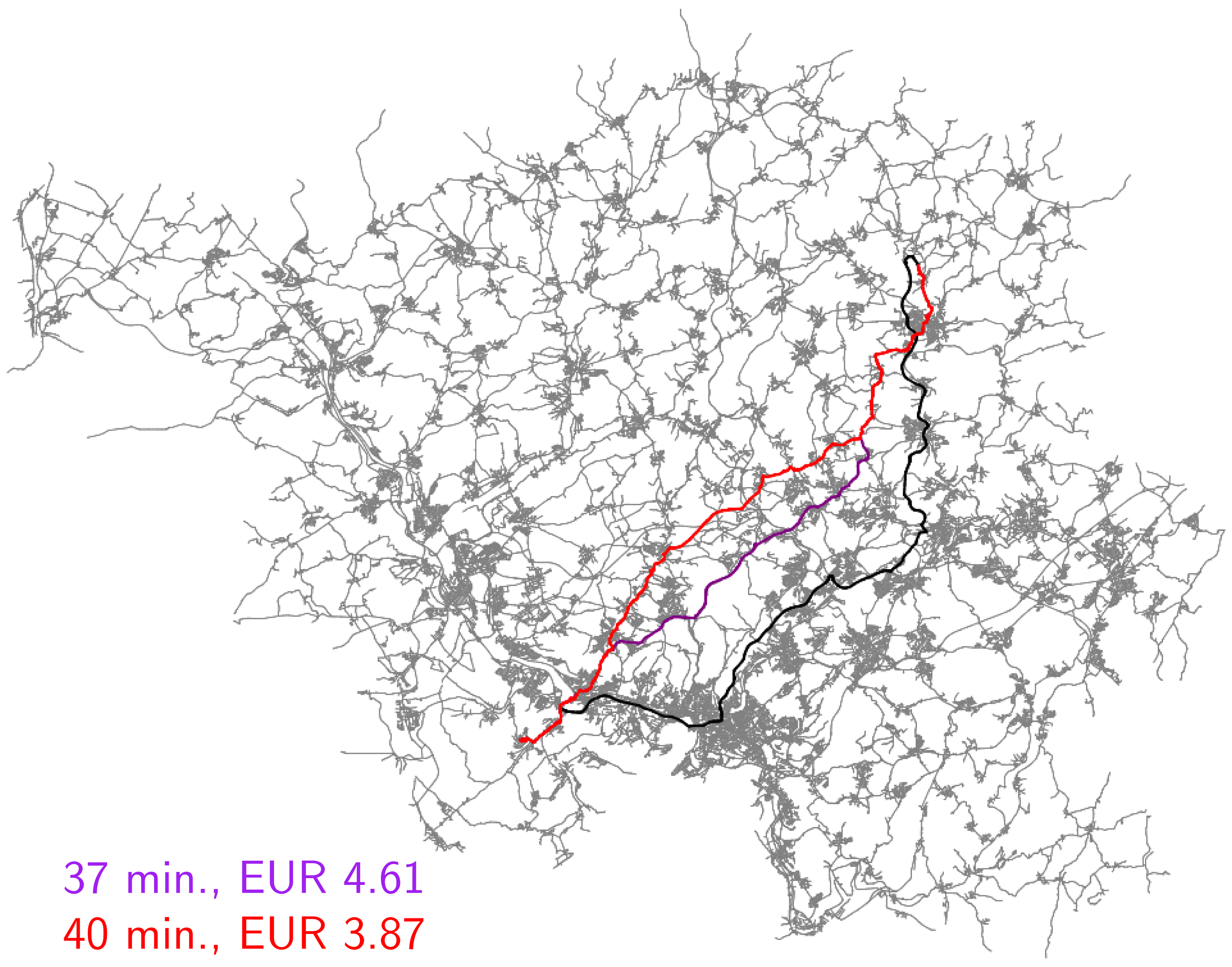
- one is the fastest
- one is the shortest
- one is the most energy-efficient
- one is the most scenic
- ...

But we might prefer a trade-off between the different criteria.

One possible way (amongst others!) is to conically combine the k different metrics and aim for the path π which minimizes this single, combined metric ($\alpha_i \geq 0$, $\sum \alpha_i = 1$):

$$\sum_{e \in \pi} \alpha_1 \cdot c_1(e) + \cdots + \alpha_k \cdot c_k(e)$$

If the weighting (the α_i) is fixed, all known speed-up techniques for shortest path queries (Transit Nodes, Reach, Hub Labels, ...) can be applied and allow for query times even below microseconds.
We are interested in a speed-up scheme where the α_i can be chosen **at query time**!



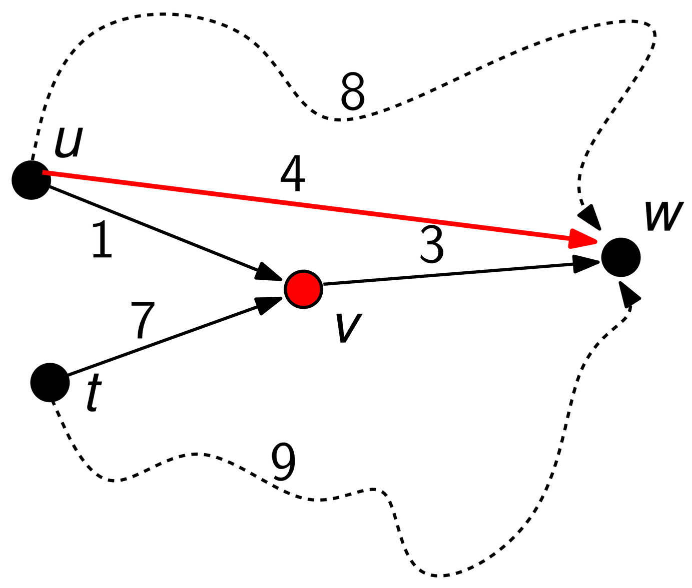
37 min., EUR 4.61
40 min., EUR 3.87
44 min., EUR 3.81

Contraction Hierarchies (CH)

by Geisberger et al., 2008 are a speed-up scheme for shortest path queries:

- originally for single scalar values on the edges
- requires a preprocessing phase (few minutes)
- creates an augmentation of the original graph
- allows for query times in the microseconds range (compared to seconds for straightforward Dijkstra)

Crucial operation in CH preprocessing: **Node Contraction**



When contracting/removing node v insert shortcuts such that shortest path distances are preserved.

Problem: The α_i and hence the actual edge weights are only revealed at query time, so decision whether shortcut between neighbors u and w needs to be created seems difficult.
Naive Solution: Examine *all* possible α_i values ... not a great idea!

Facets of Lower Envelopes

Consider the k-dimensional space $\mathcal{S}$ where the first $k-1$ dimensions correspond to the choice of $\alpha_1, \ldots \alpha_{k-1}$, the k-th dimension to the combined cost.

- a path π between neighbors u and w of node v to be contracted corresponds to a hyperplane h_π in $\mathcal{S}$
- π is shortest for some choice of the α_i **iff** h_π appears on the lower envelope of all paths between u and w

$\Rightarrow$ A shortcut between u and w needs to be inserted **iff** the hyperplane h_{uvw} appears on the lower envelope.

Strategy:

- if we encounter some α_i for which uvw is optimal $\Rightarrow$ create shortcut uw
- h_{uvw} still supporting facet of the lower envelope $\Rightarrow$ compute optimal path for deliberately chosen α_i such that facet volume is decreased by constant factor α
- show upper and lower bounds for facet volume

It follows that $O(d^2 log_\alpha(nMd))$ shortest path computations suffice to decide on membership of h_{uvw} on lower envelope/necessity of shortcut uw.

Experimental Results

California 11M nodes, 23M edges

Metrics	travel time + fuel costs
Plain Dijkstra	2097ms
CH-Dijkstra	2ms
speed-up	>1000

Metrics	travel time + distance
Plain Binary Search (CSP-APX)	20827ms
CH Binary Search (CSP-APX)	18ms
speed-up	>1000

Metrics	travel time + fuel costs + distance
Plain Dijkstra	3299ms
CH-Dijkstra	16ms
Plain Trinary Search (CSP-APX)	342795ms
CH Trinary Search (CSP-APX)	1374ms
speed-up	> 200