Karlsruhe Institute of Technology

Public Transit Routing with Unrestricted Walking

Dorothea Wagner and Tobias Zündorf

ATMOS · September 7th, 2017

INSTITUTE OF THEORETICAL INFORMATICS · ALGORITHMICS GROUP

Overview

Karlsruhe Institute of Technology

Introduction:

- Problem Statement
- State of the Art

Our Contribution:

- Our Profile Algorithm
- Direct Walking

Evaluation:

- Instances & Experimental Setup
- Runtime Performance
- Travel Time Comparison

Institute of Theoretical Informatics
Algorithmics Group

Problem – Public Transit & Walking

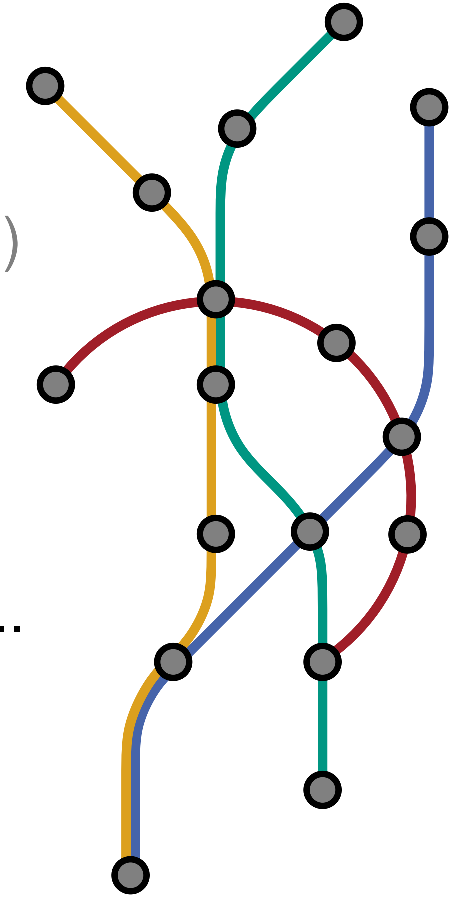
Public Transit Routing:

Given:

- Stops (where vehicles can be entered or alighted)
- The timetable (routes, trips, connections)
- Footpaths (possible transfers between stops)

Goal:

- Optimal journey between two stops
- With respect to travel time, number of transfers, ...



Institute of Theoretical Informatics
Algorithmics Group

Problem – Public Transit & Walking

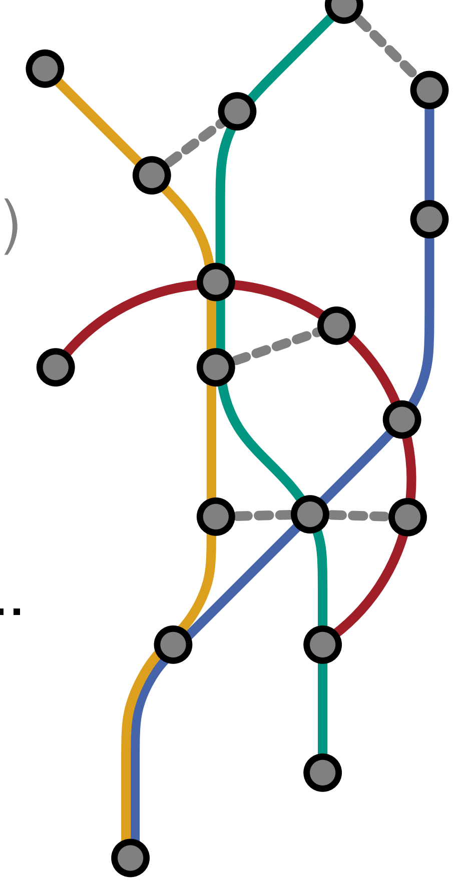
Public Transit Routing:

Given:

- Stops (where vehicles can be entered or alighted)
- The timetable (routes, trips, connections)
- Footpaths (possible transfers between stops)

Goal:

- Optimal journey between two stops
- With respect to travel time, number of transfers, ...



Institute of Theoretical Informatics
Algorithmics Group

Problem – Public Transit & Walking

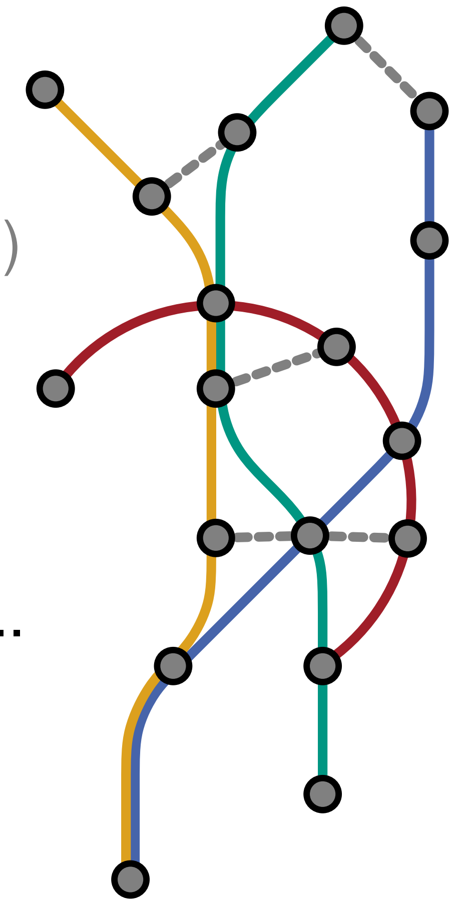
Public Transit Routing:

Given:

- Stops (where vehicles can be entered or alighted)
- The timetable (routes, trips, connections)
- Footpaths (possible transfers between stops)

Goal:

- Optimal journey between two stops
- With respect to travel time, number of transfers, ...



Question:

- How many footpaths are needed?
- How are travel times affected by different footpaths?

Institute of Theoretical Informatics
Algorithmics Group

Problem – Public Transit & Walking

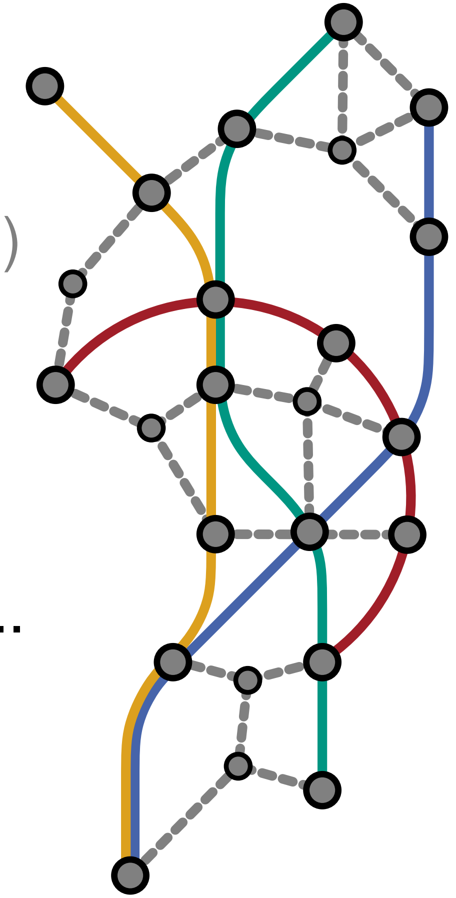
Public Transit Routing:

Given:

- Stops (where vehicles can be entered or alighted)
- The timetable (routes, trips, connections)
- Footpaths (possible transfers between stops)

Goal:

- Optimal journey between two stops
- With respect to travel time, number of transfers, ...



Question:

- How many footpaths are needed?
- How are travel times affected by different footpaths?

Institute of Theoretical Informatics
Algorithmics Group

Problem – Public Transit & Walking

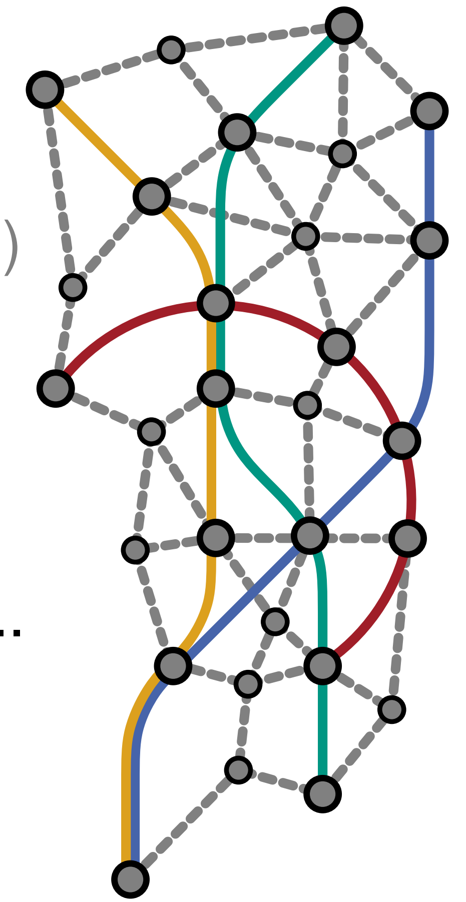
Public Transit Routing:

Given:

- Stops (where vehicles can be entered or alighted)
- The timetable (routes, trips, connections)
- Footpaths (possible transfers between stops)

Goal:

- Optimal journey between two stops
- With respect to travel time, number of transfers, ...



Question:

- How many footpaths are needed?
- How are travel times affected by different footpaths?

Institute of Theoretical Informatics
Algorithmics Group

Related Work

Karlsruhe Institute of Technology

State of the Art:

- Many different footpath models
- Often require restrictions

Common Restrictions:

Algorithm	Footpaths
RAPTOR [Delling et al. ’12/’14]	Transitively closed
CSA [Dibbelt et al. ’13/’14]	Transitively closed
Trip-Based Routing [Witt ’15]	Transitively closed
Transfer Patterns [Bast et al. ’10/’16]	Max. 400 meters
Frequency-Based [Bast, Storandt ’14]	Max. 15 minutes
Public Transit Labeling [Delling et al. ’15]	As specified by the timetable

Reasons for Restricted Walking

Karlsruhe Institute of Technology

Common Arguments:

- Passengers do not want to walk far
- Restricted walking is sufficient for optimal solutions

Institute of Theoretical Informatics
Algorithmics Group

Reasons for Restricted Walking

Common Arguments:

- Passengers do not want to walk far
- Restricted walking is sufficient for optimal solutions

But:

- Some passengers might want to walk far
- Decision is made without knowing how much walking is required
- I has not been proven how much walking is required

Institute of Theoretical Informatics
Algorithmics Group

How to Evaluate the Importance of Walking

Karlsruhe Institute of Technology

Important Aspect:

- Departure time (day vs. night)
- Query distance (short range vs. long range)
- Source and target location (rural vs. urban)

Institute of Theoretical Informatics
Algorithmics Group

How to Evaluate the Importance of Walking

Important Aspect:

- Departure time (day vs. night)
- Query distance (short range vs. long range)
- Source and target location (rural vs. urban)

Thus, we need:

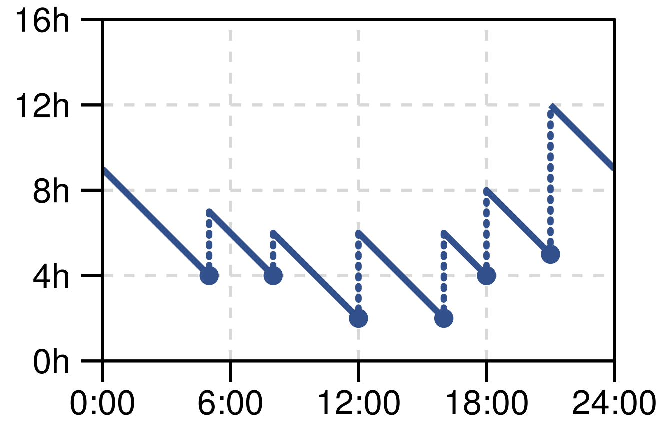
- Profile algorithms (analyze travel times for a whole day)
- Realistic queries for several source to target distances
- Instances with different amounts of footpaths

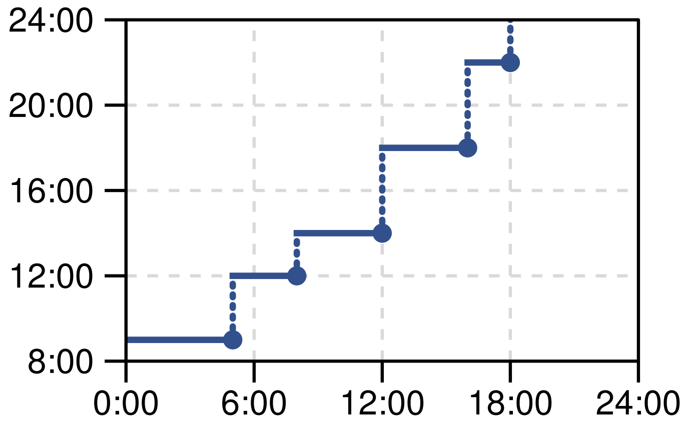
Institute of Theoretical Informatics
Algorithmics Group

Profiles & Profile Algorithms

Definition:

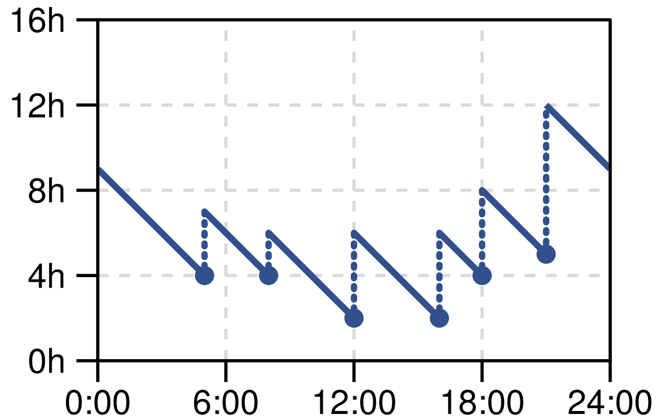
- Function mapping departure time to either travel time or arrival time

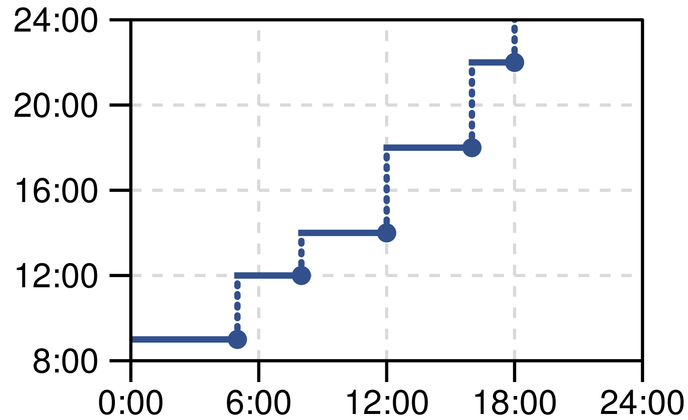




Definition:

- Function mapping departure time to either travel time or arrival time





Profile Algorithms for Unrestricted Walking:

- Most algorithms are not suitable for unrestricted walking
- Multimodal Multicriteria RAPTOR (MCR) [Delling et al. ’13]
  - Can handle arbitrary walking
  - Only earliest arrival queries

Profiles – RAPTOR & MCR

MCR Profile Algorithm?:

- MCR is based on RAPTOR
- RAPTOR can be used to compute profiles (rRAPTOR)
- Why does this not work for MCR?

Institute of Theoretical Informatics
Algorithmics Group

Profiles – RAPTOR & MCR

MCR Profile Algorithm?:

- MCR is based on RAPTOR
- RAPTOR can be used to compute profiles (rRAPTOR)
- Why does this not work for MCR?

rRAPTOR Profile Query: (rough outline)

- Profile entires are limited by number of trips departing from source
- Collect all departure times at the source
- Run RAPTOR once for every departure time

Institute of Theoretical Informatics
Algorithmics Group

Profiles – RAPTOR & MCR

MCR Profile Algorithm?:

- MCR is based on RAPTOR
- RAPTOR can be used to compute profiles (rRAPTOR)
- Why does this not work for MCR?

rRAPTOR Profile Query: (rough outline)

- Profile entires are limited by number of trips departing from source
- Collect all departure times at the source
- Run RAPTOR once for every departure time

Problem with direct walking / MCR:

- Passenger can walk to arbitrary stop before boarding the first trip
- Every departure time needs to be considered

  $\Rightarrow$ To many calls of RAPTOR

Our Algorithm

Karlsruhe Institute of Technology

Goal:

- Use earliest arrival algorithm as black box (MCR)
- Earliest arrival algorithm computes single profile entry
- Number of algorithm calls $\approx$ number of profile entries

Institute of Theoretical Informatics
Algorithmics Group

Our Algorithm

Karlsruhe Institute of Technology

Goal:

- Use earliest arrival algorithm as black box (MCR)
- Earliest arrival algorithm computes single profile entry
- Number of algorithm calls $\approx$ number of profile entries

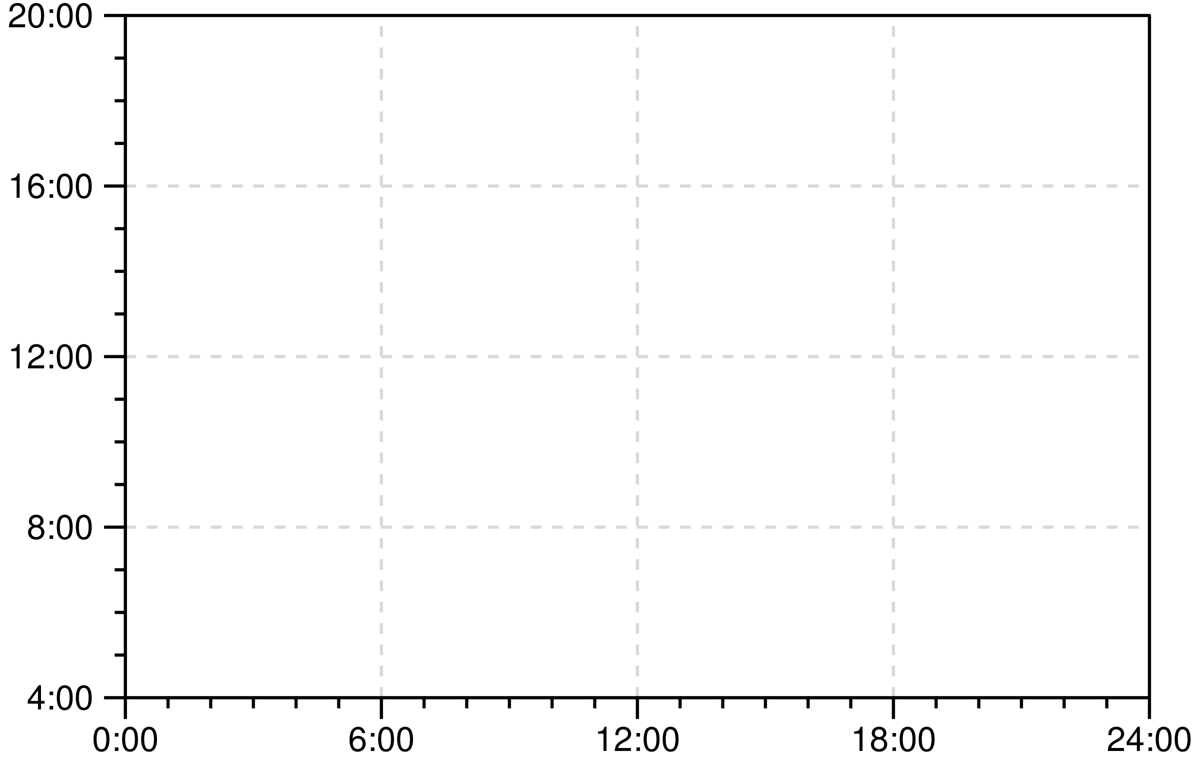
Overview:

- Establish profile one entry at a time
- Use two black box calls per entry (forward + backward)
  - Start with earliest possible departure
  - Forward query $\Rightarrow$ earliest possible arrival
  - Backward query $\Rightarrow$ latest possible departure for that arrival
  - Continue with latest possible departure $+ \varepsilon$

Our Algorithm – Example

Goal:

- Compute a profile for the interval [0:00, 24:00]



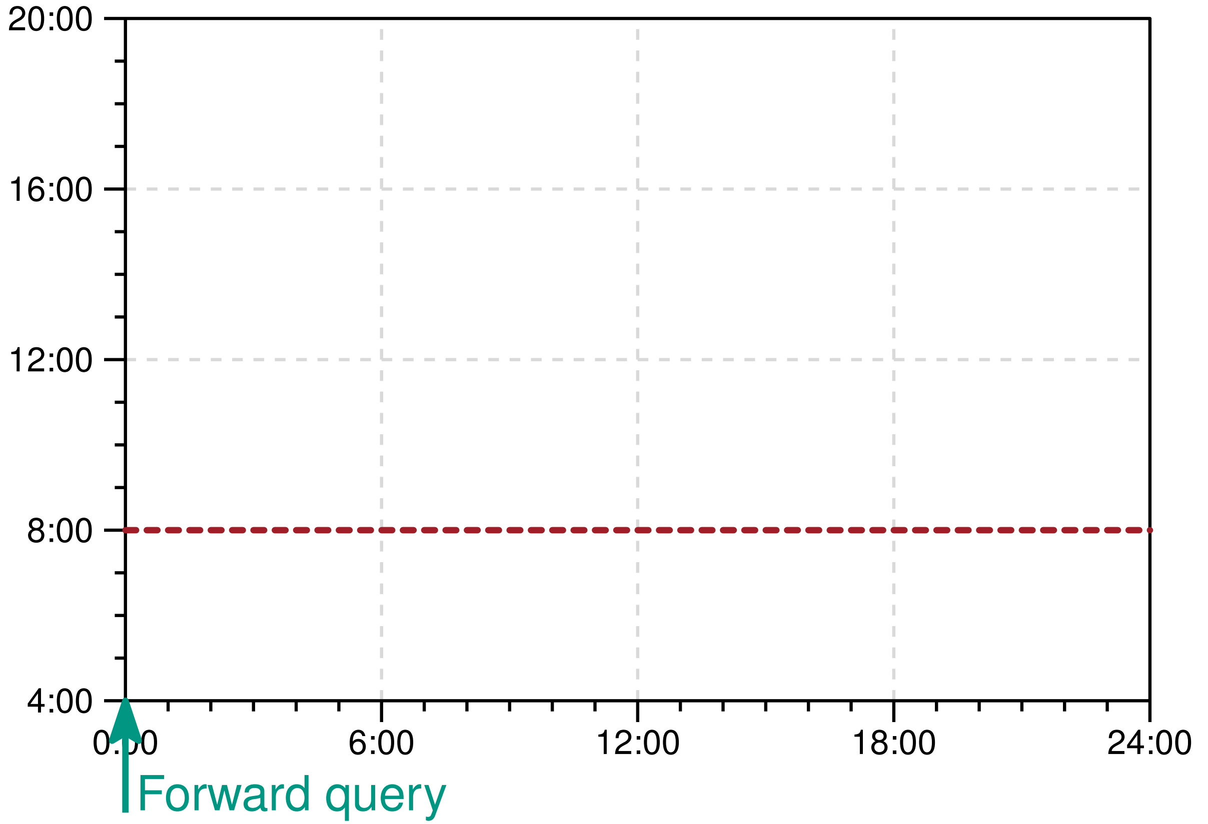
Institute of Theoretical Informatics
Algorithmics Group

Our Algorithm – Example

Goal:

- Compute a profile for the interval [0:00, 24:00]

Step 1: Run a forward query from the earliest departure time (0:00)



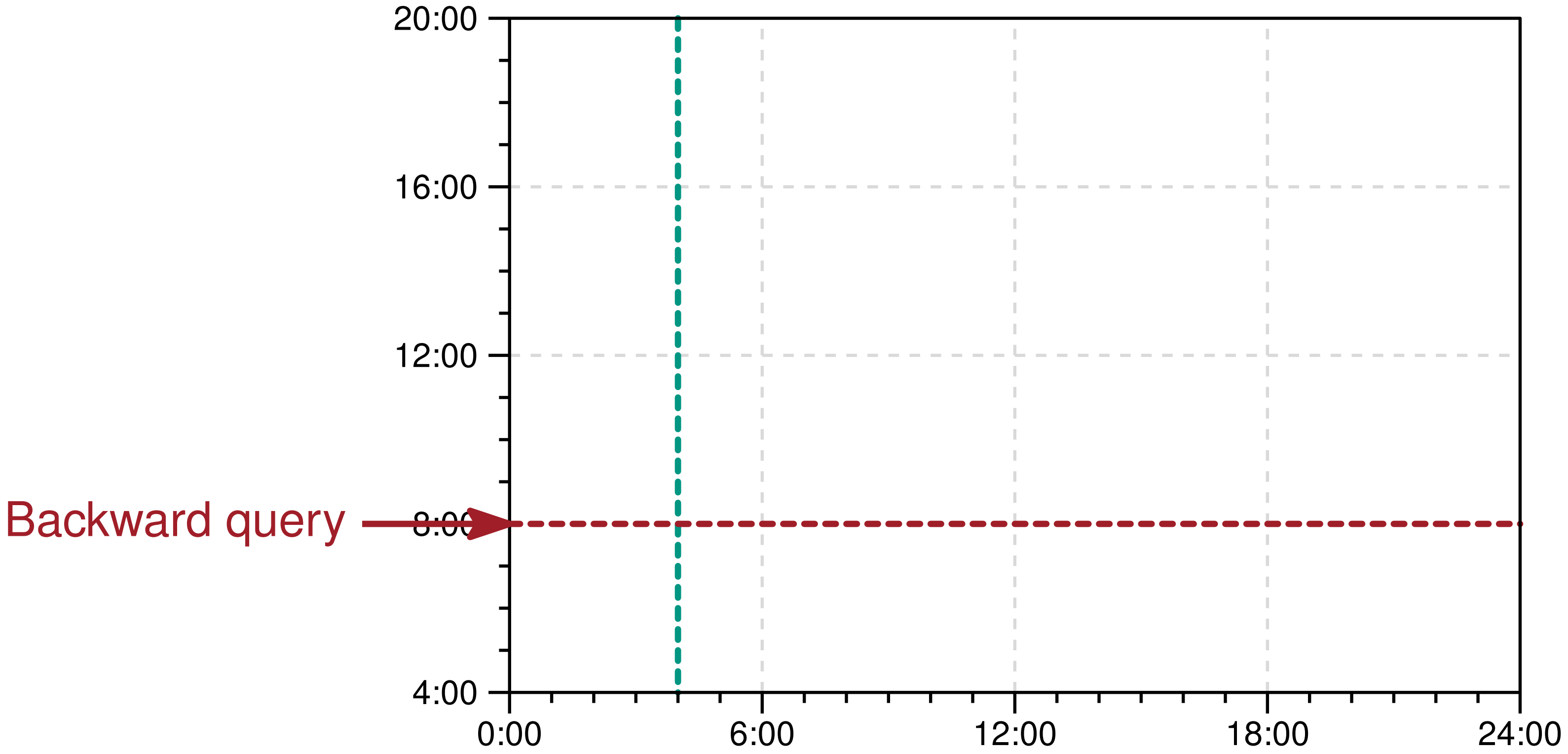
Institute of Theoretical Informatics
Algorithmics Group

Our Algorithm – Example

Goal:

- Compute a profile for the interval [0:00, 24:00]

Step 2: Run a backward query from the resulting arrival time (8:00)



Institute of Theoretical Informatics
Algorithmics Group

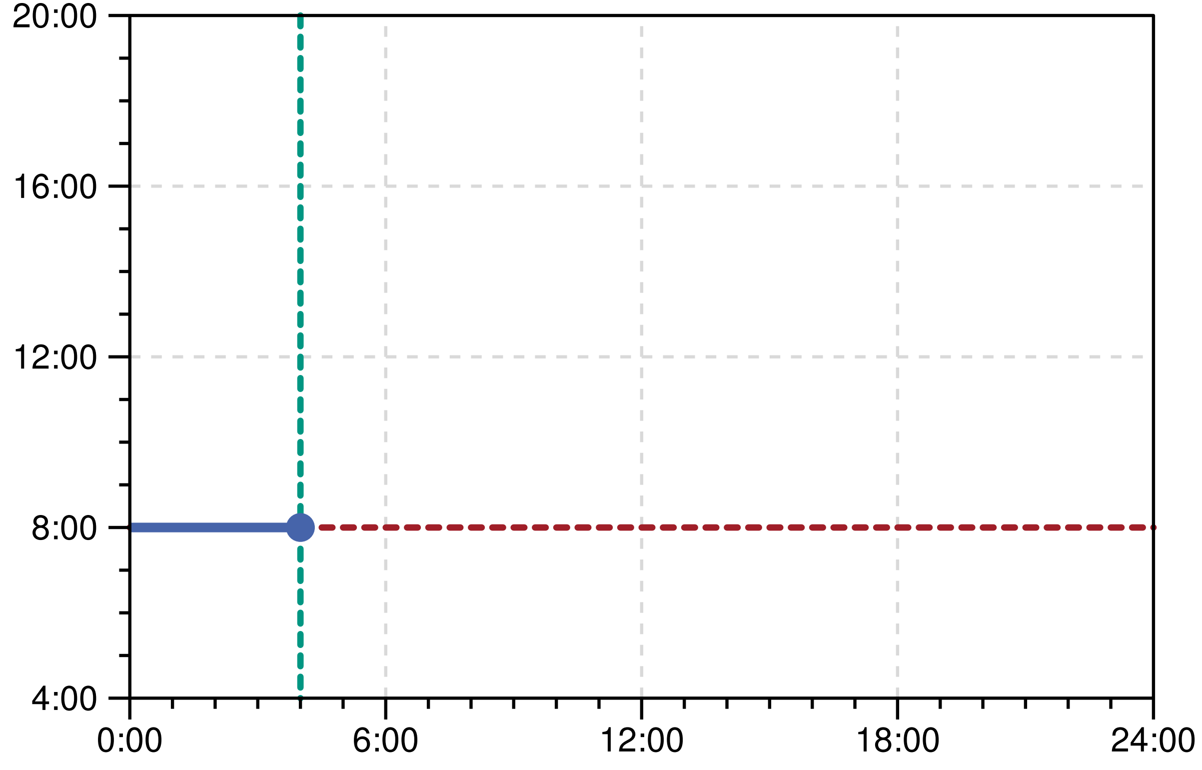
Our Algorithm – Example

Karlsruhe Institute of Technology

Goal:

- Compute a profile for the interval [0:00, 24:00]

Step 3: Both results define one point of the profile



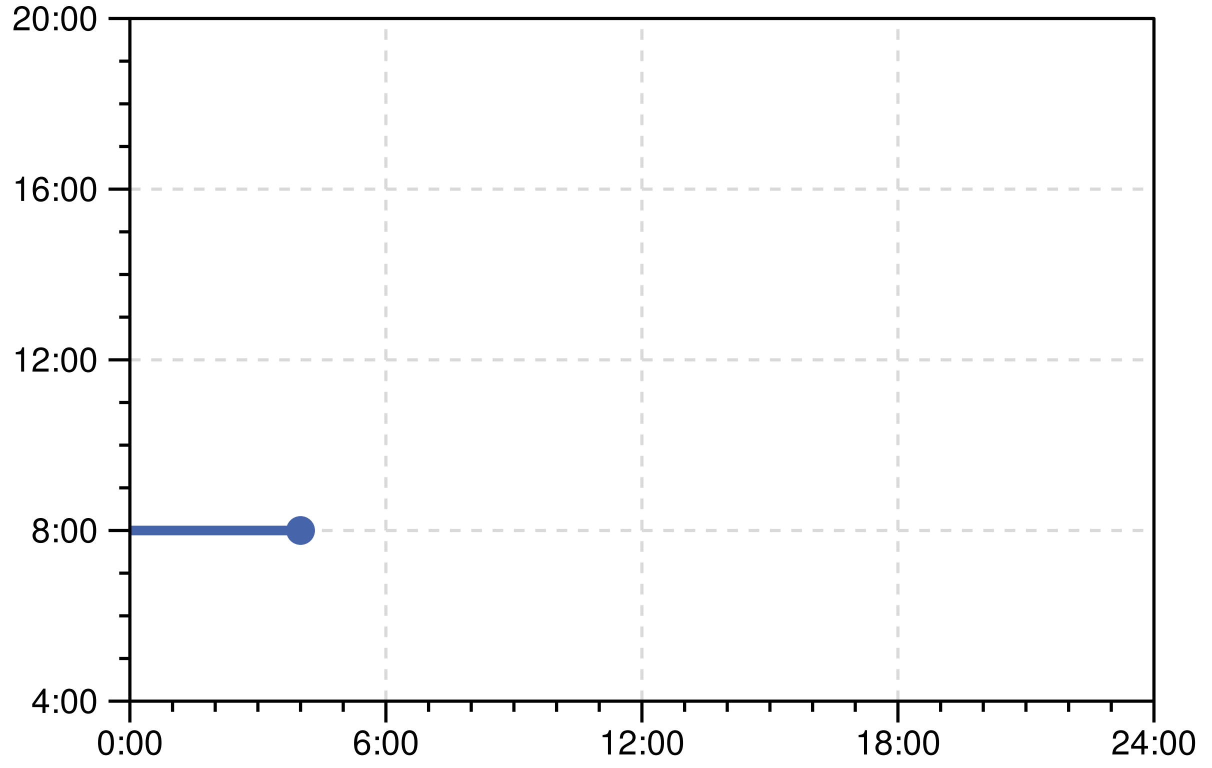
Institute of Theoretical Informatics
Algorithmics Group

Our Algorithm – Example

Goal:

- Compute a profile for the interval [0:00, 24:00]

Step 4: The profile is correct for the interval [0:00, 4:00]

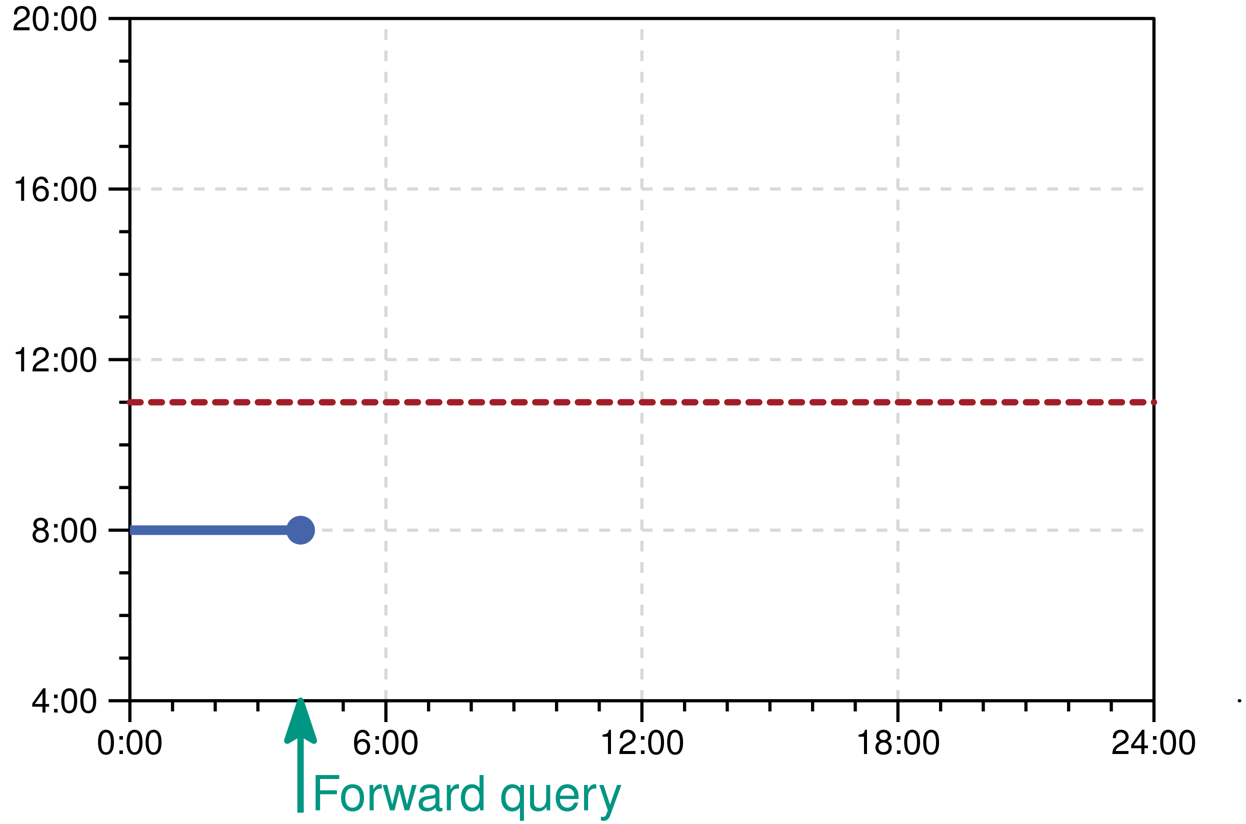


Our Algorithm – Example

Goal:

- Compute a profile for the interval (4:00, 24:00]

Step 1: Run a forward query from the earliest departure time (4:00 + ε)

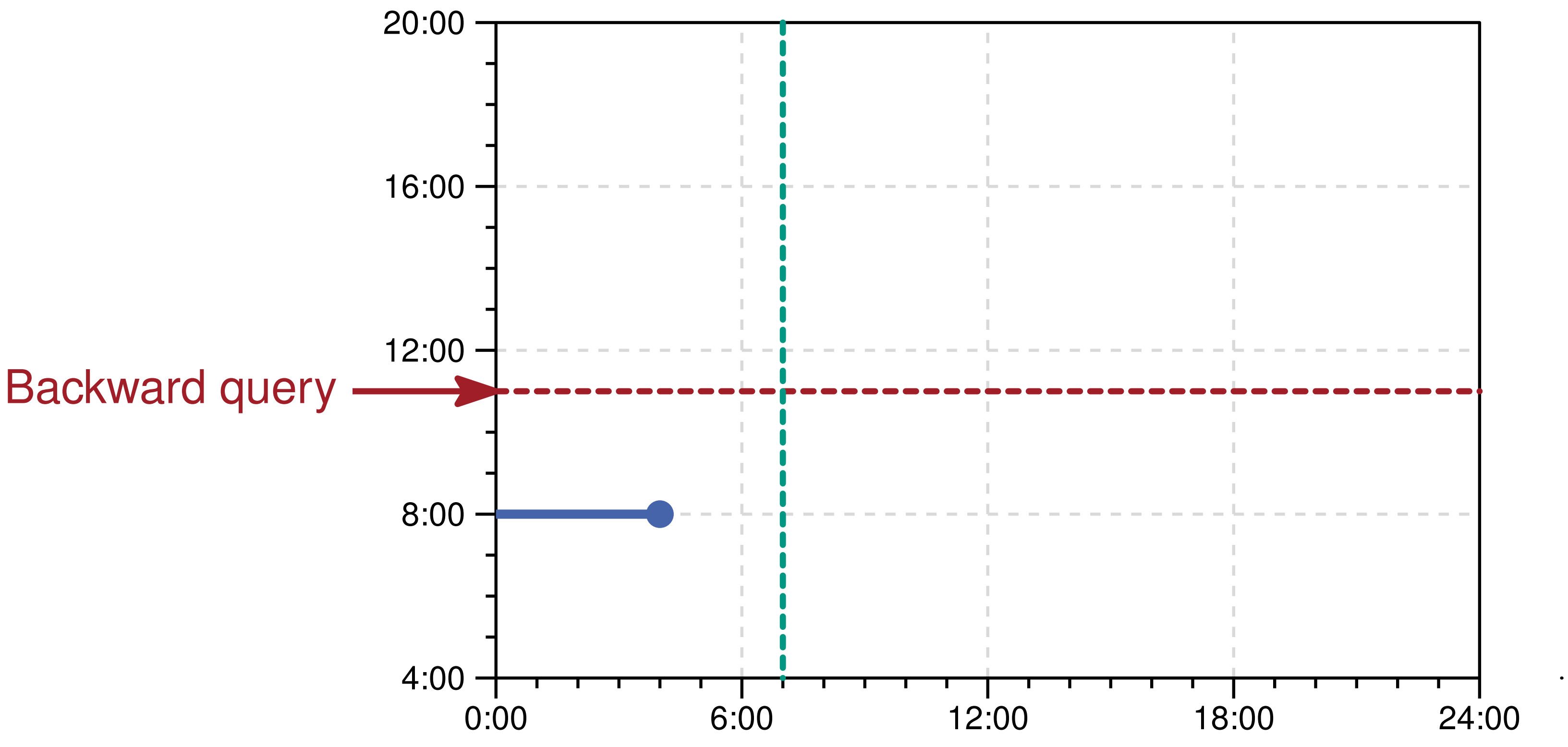


Our Algorithm – Example

Goal:

- Compute a profile for the interval (4:00, 24:00]

Step 2: Run a backward query from the resulting arrival time (11:00)



Institute of Theoretical Informatics
Algorithmics Group

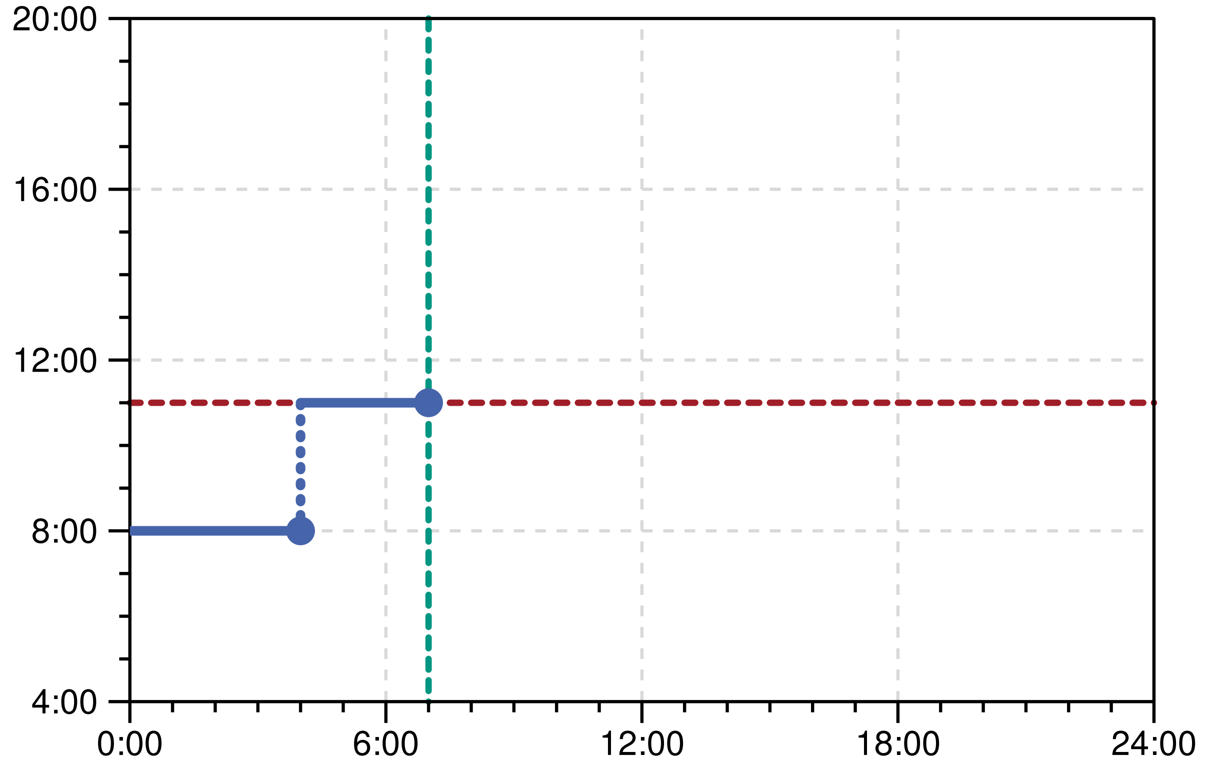
Our Algorithm – Example

Karlsruhe Institute of Technology

Goal:

- Compute a profile for the interval (4:00, 24:00]

Step 3: Both results define one point of the profile



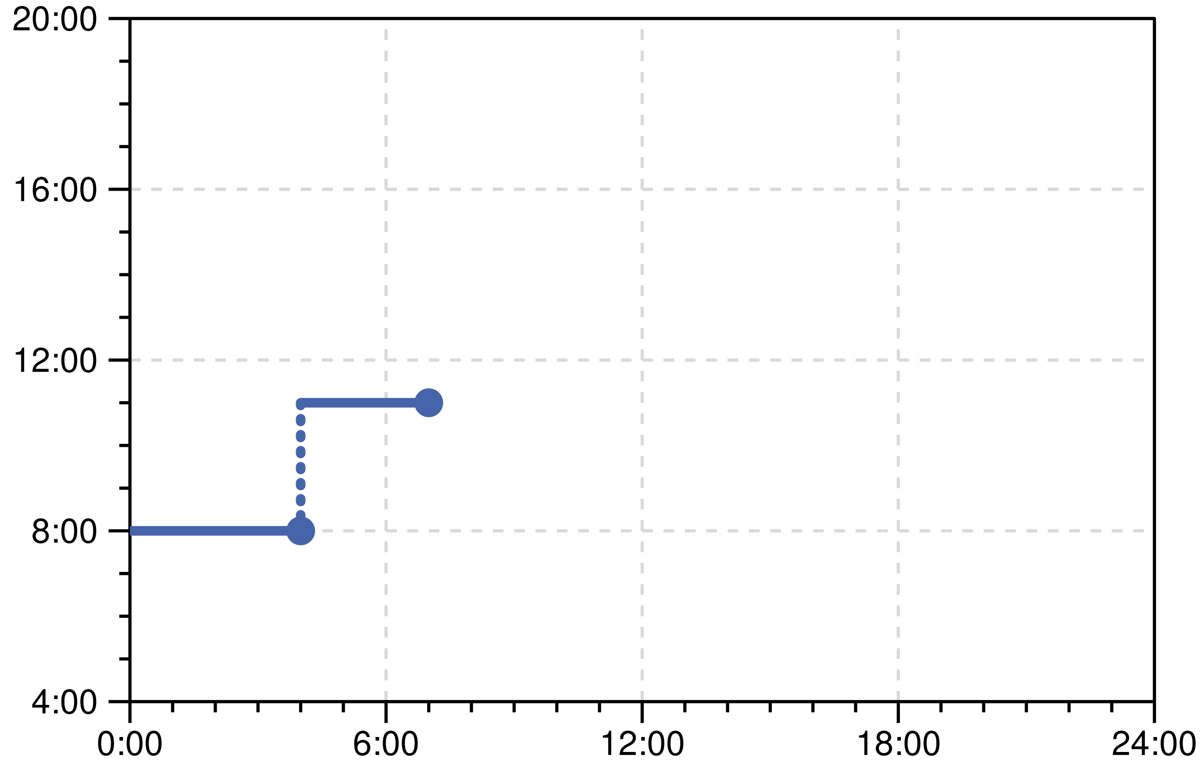
Institute of Theoretical Informatics
Algorithmics Group

Our Algorithm – Example

Goal:

- Compute a profile for the interval (4:00, 24:00]

Step 4: The profile is correct for the interval [0:00, 7:00]

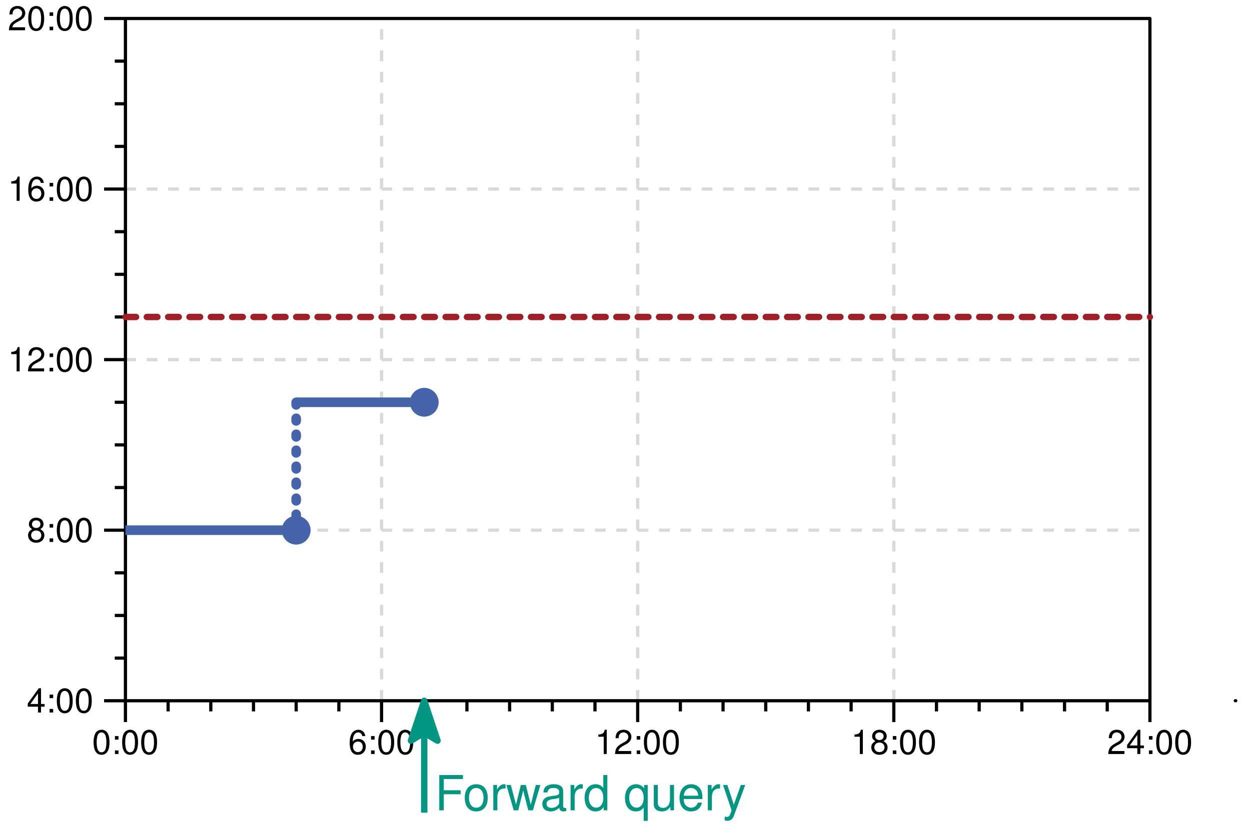


Our Algorithm – Example

Goal:

- Compute a profile for the interval (7:00, 24:00]

Step 1: Run a forward query from the earliest departure time (7:00 + ε)



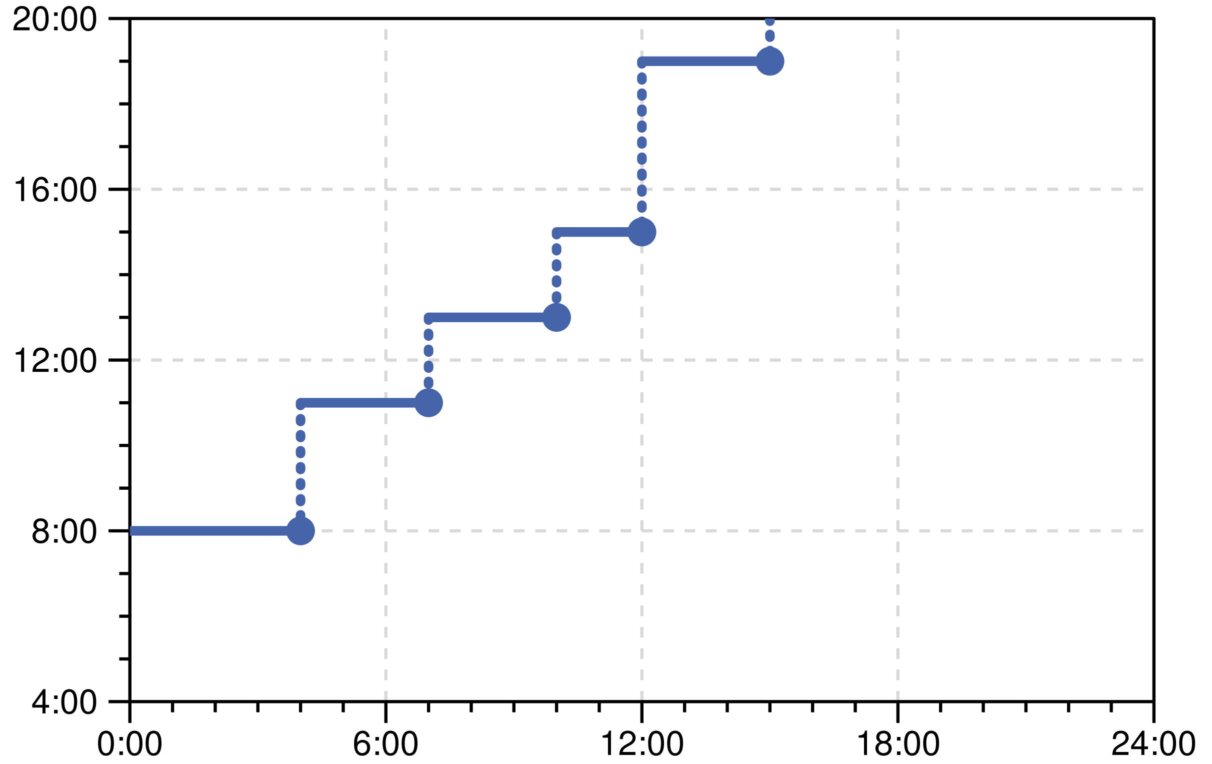
Institute of Theoretical Informatics
Algorithmics Group

Our Algorithm – Example

Goal:

- Compute a profile for the interval (7:00, 24:00]

End: The algorithm terminates when the profile is complete



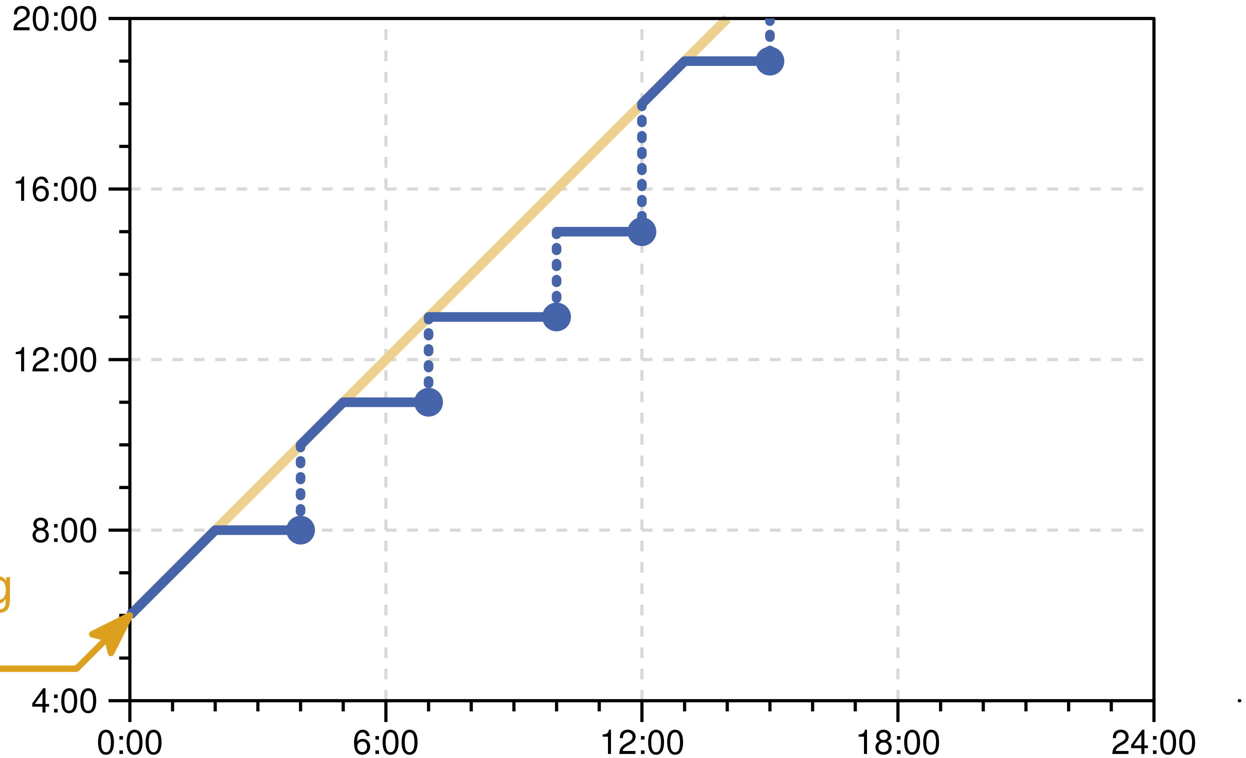
Our Algorithm – Direct Walking

Problem:

- Direct walking prevents progress

Example:

Direct Walking takes 6 hours



Institute of Theoretical Informatics
Algorithmics Group

Our Algorithm – Direct Walking

Karlsruhe Institute of Technology

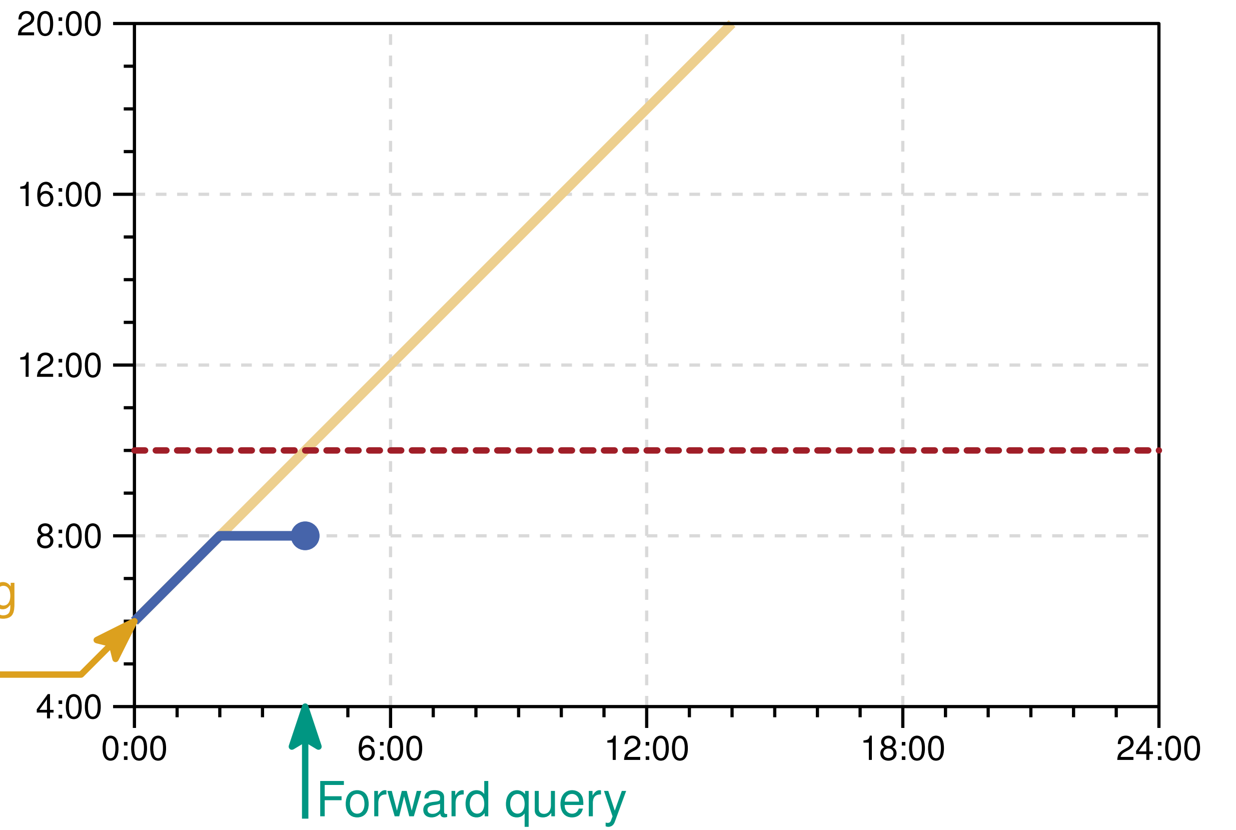
Goal:

- Compute a profile for the interval (4:00, 24:00]

Step 1: Run a forward query from the earliest departure time (4:00 + ε)

Example:

Direct Walking takes 6 hours



Institute of Theoretical Informatics
Algorithmics Group

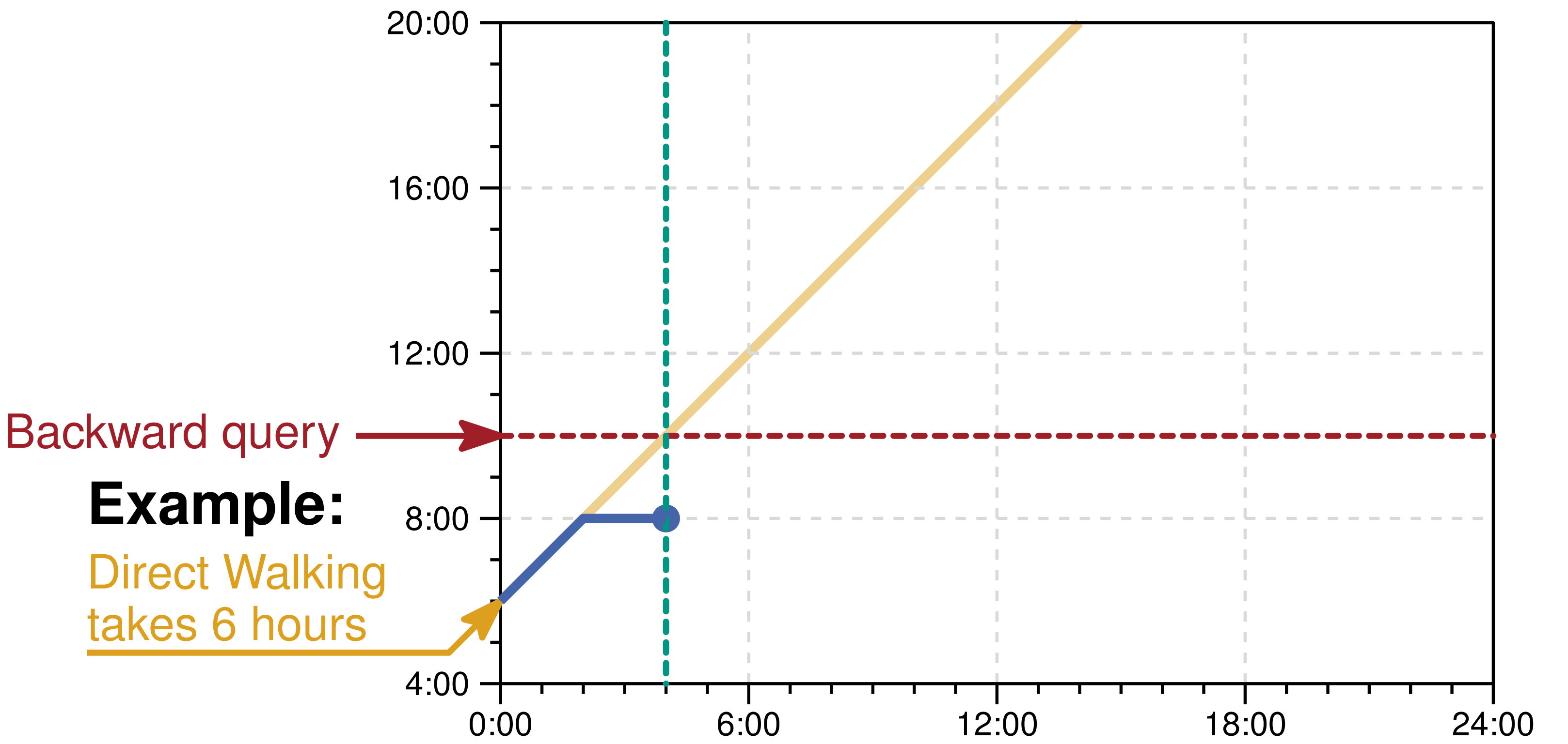
Our Algorithm – Direct Walking

Karlsruhe Institute of Technology

Goal:

- Compute a profile for the interval (4:00, 24:00]

Step 2: Run a backward query from the resulting arrival time (10:00)



Our Algorithm – Direct Walking

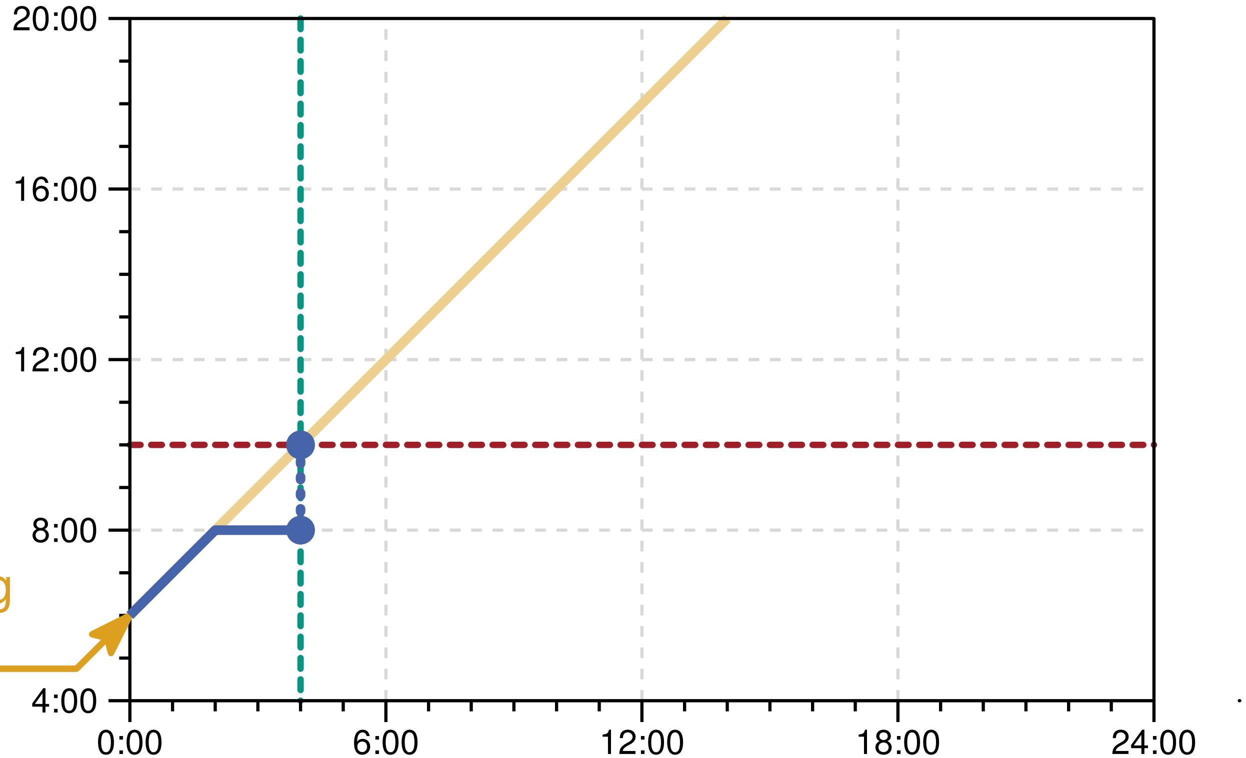
Goal:

- Compute a profile for the interval (4:00, 24:00]

Step 3: Both results define one point of the profile

Example:

Direct Walking takes 6 hours



Institute of Theoretical Informatics
Algorithmics Group

Our Algorithm – Direct Walking

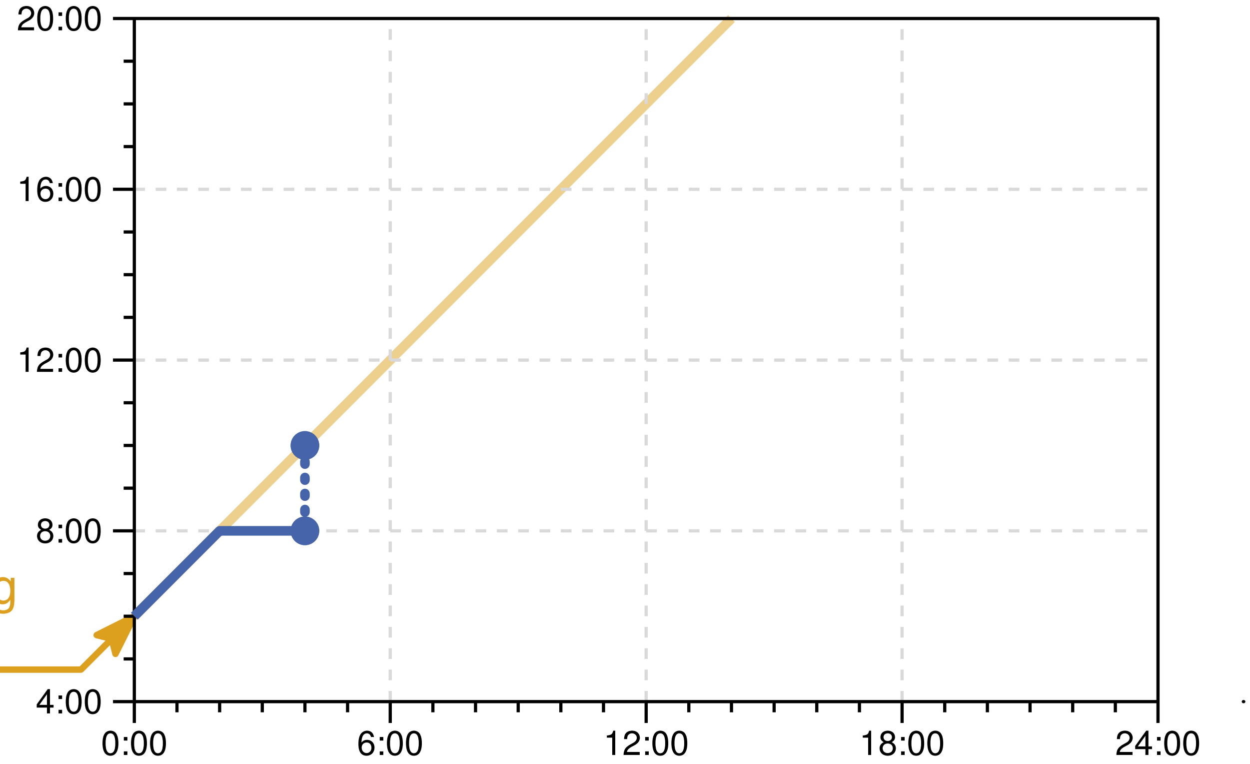
Goal:

- Compute a profile for the interval (4:00, 24:00]

Step 4: The profile is still only correct for the interval [0:00, 4:00]

Example:

Direct Walking takes 6 hours



Institute of Theoretical Informatics
Algorithmics Group

Our Algorithm – Direct Walking

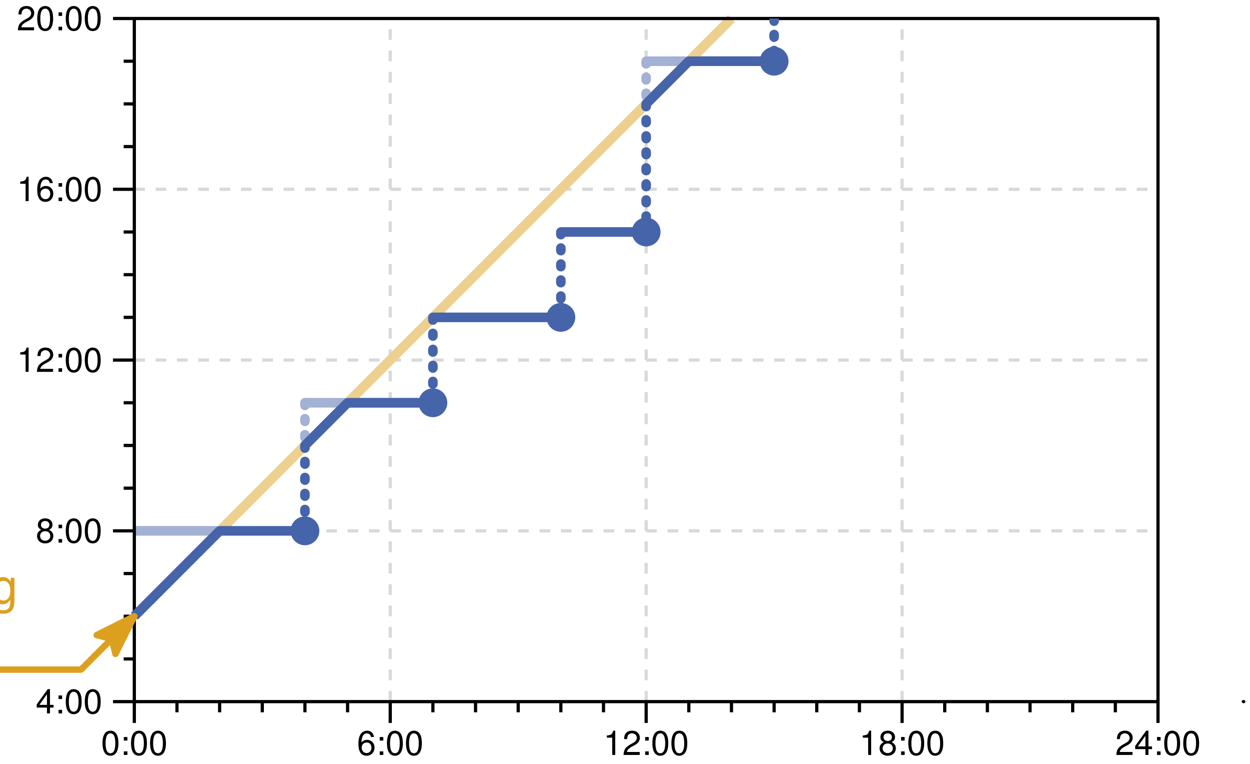
Problem:

- Direct walking prevents progress

Solution: Compute profile in network without direct walking
& Add direct Walking path to the solution afterwards

Example:

Direct Walking
takes 6 hours



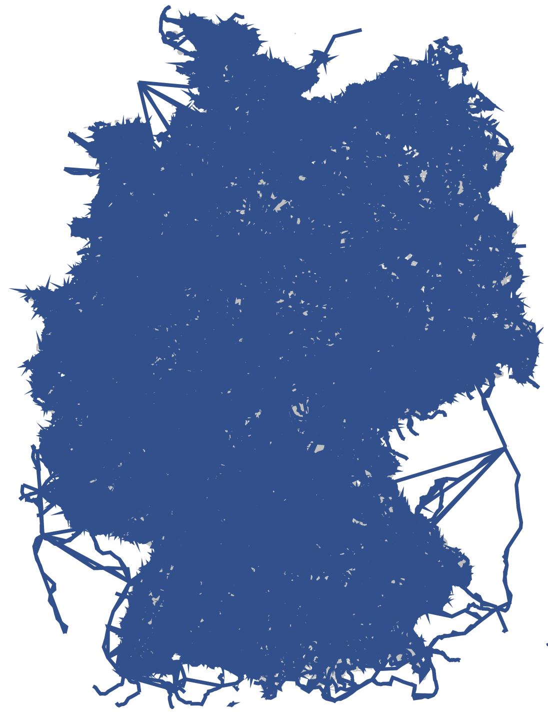
Institute of Theoretical Informatics
Algorithmics Group

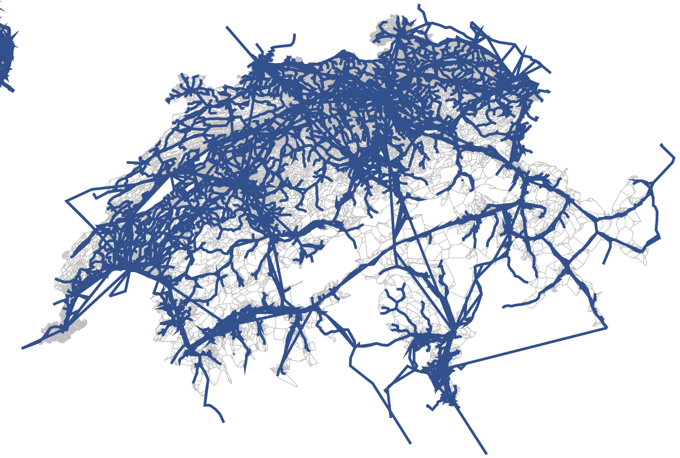
Evaluation

Karlsruhe Institute of Technology

Instances:

- Germany (from `bahn.de`)
- Switzerland (GTFS feed)
- Footpath graph from OpenStreetMap (OSM)



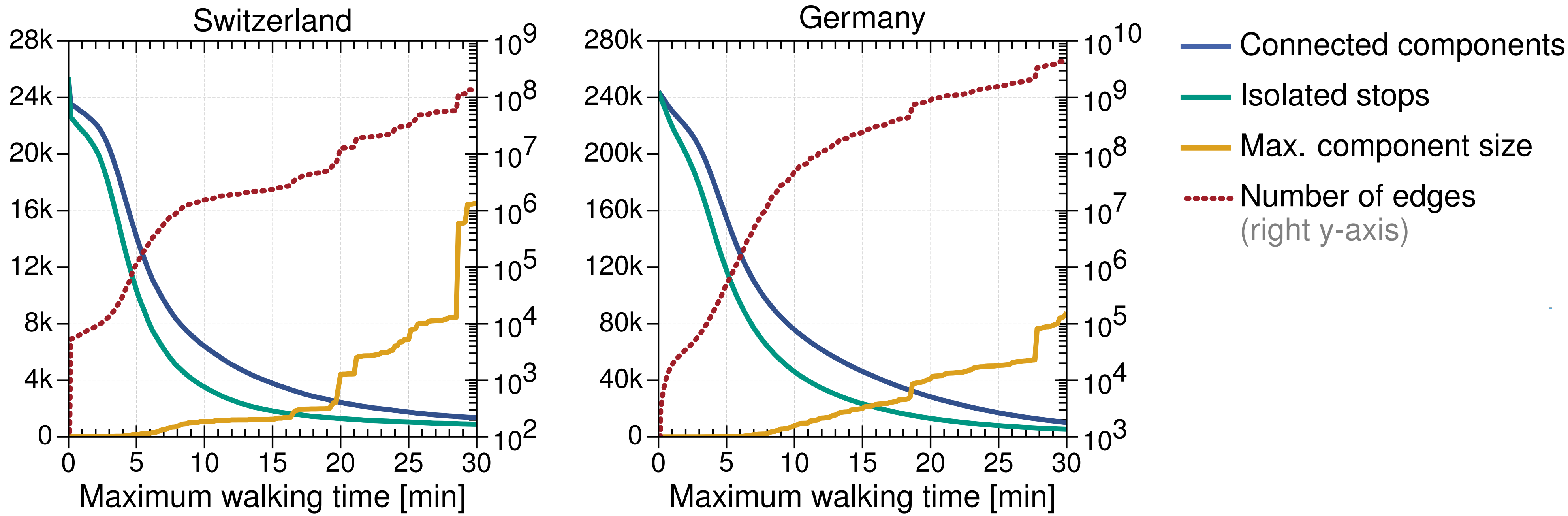


PT network	Footpaths	Stops	Vertices	Edges	Connections	Max. deg.
Germany	original	244 245	244 245	95 036	46 119 896	18
	partial	244 245	244 245	26 193 136	46 119 896	2 622
	complete	244 245	6 876 758	21 382 408	46 119 896	21
Switzerland	original	25 427	25 427	5 604	4 373 268	25
	partial	25 427	25 427	3 104 974	4 373 268	1 246
	complete	25 427	604 230	1 844 286	4 373 268	25

Evaluation – Partial Instances

Goal:

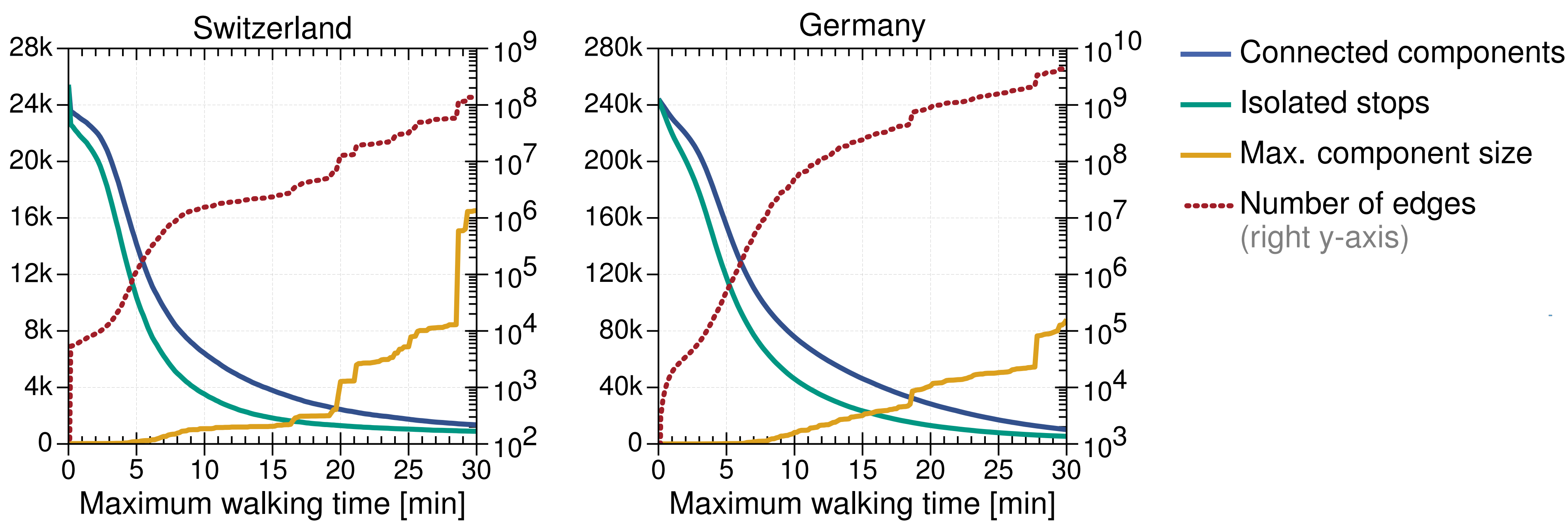
- Preserve paths between stops up to a certain walking time
- Obtain a transitively closed graph of reasonable size



Evaluation – Partial Instances

Goal:

- Preserve paths between stops up to a certain walking time
- Obtain a transitively closed graph of reasonable size
- Reasonable size $\approx$ average vertex degree $\leq$ 100
  $\Rightarrow$ Switzerland 15 min walking
  $\Rightarrow$ Germany 8 min walking



Evaluation – Queries

Problem:

- Random queries do not reflect reality
- Could result in an overestimation of the importance of walking
- Real queries are not available

Institute of Theoretical Informatics
Algorithmics Group

Evaluation – Queries

Karlsruhe Institute of Technology

Problem:

- Random queries do not reflect reality
- Could result in an overestimation of the importance of walking
- Real queries are not available

Assumption:

- Stop is often used as source/target $\Leftrightarrow$ Stop is part of many trips

Our Queries:

- Choose source at random
- Probability is proportional to number of trips containing the stop
- Choose target at random from stops with certain distance rank

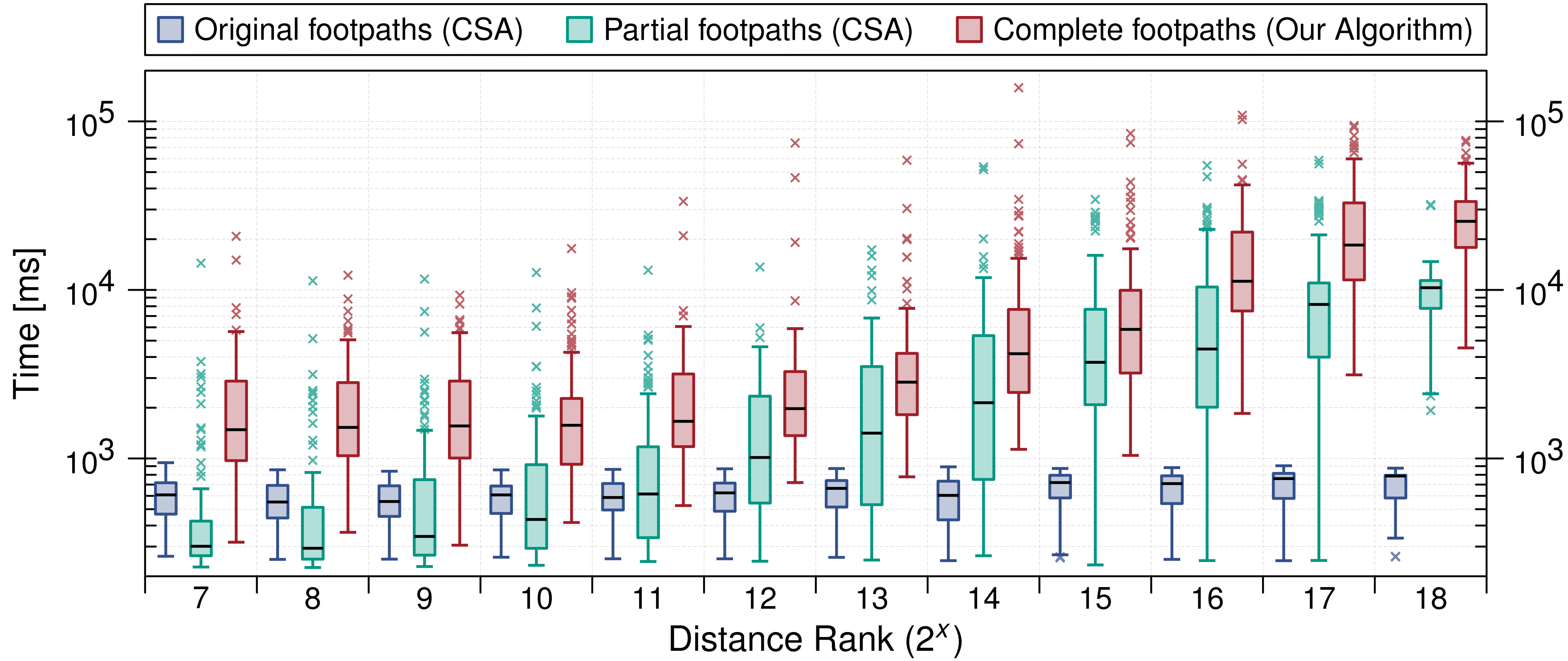
Institute of Theoretical Informatics
Algorithmics Group

Evaluation – Running Time

Karlsruhe Institute of Technology

Run time comparison:

- Switzerland public transit network
- Computation of Pareto-profiles (travel time, number of transfers)
- CSA for the transitively closed instances
- Our algorithm for the complete instance

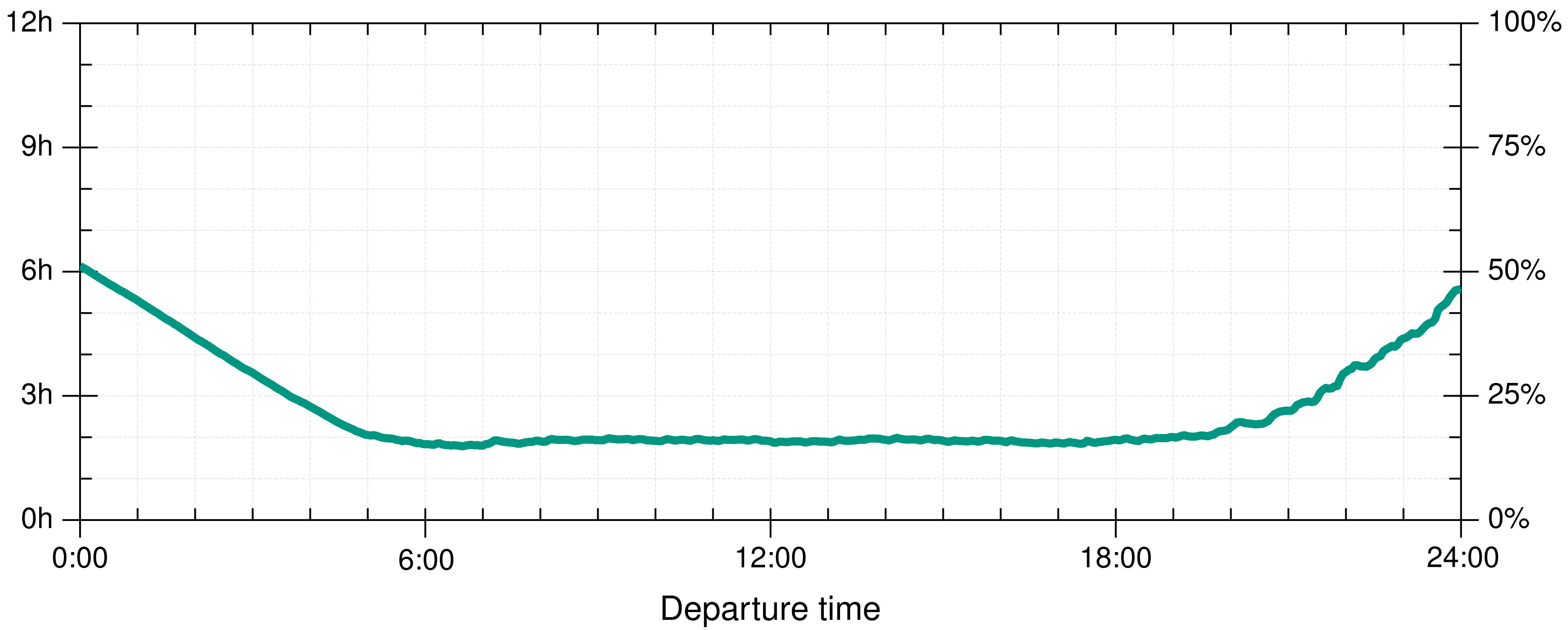


Evaluation – Travel Time Results

Travel time comparison:

- Switzerland complete vs. Switzerland original (Distance rank 16)

Average travel time (complete)

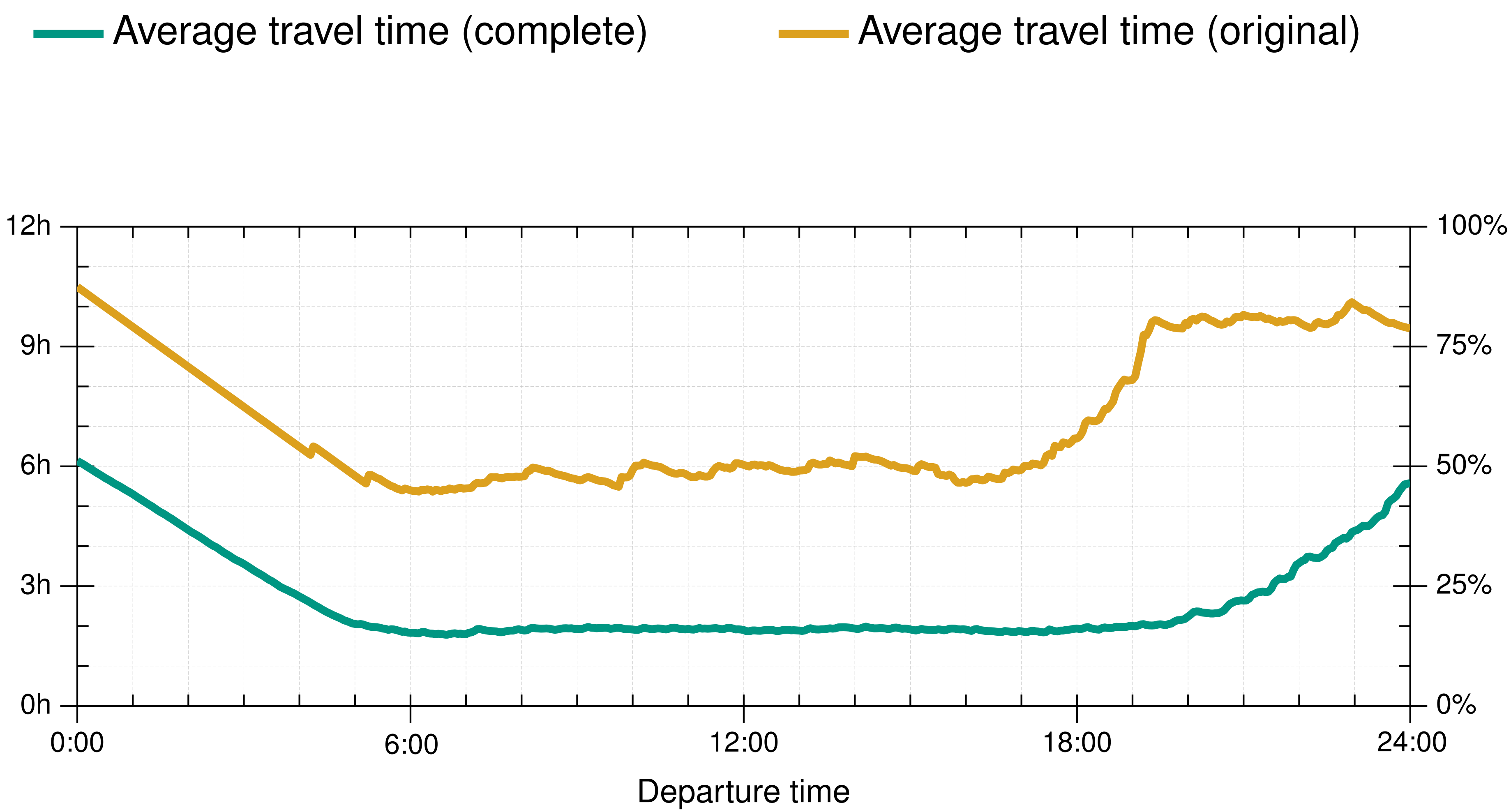


Institute of Theoretical Informatics
Algorithmics Group

Evaluation – Travel Time Results

Travel time comparison:

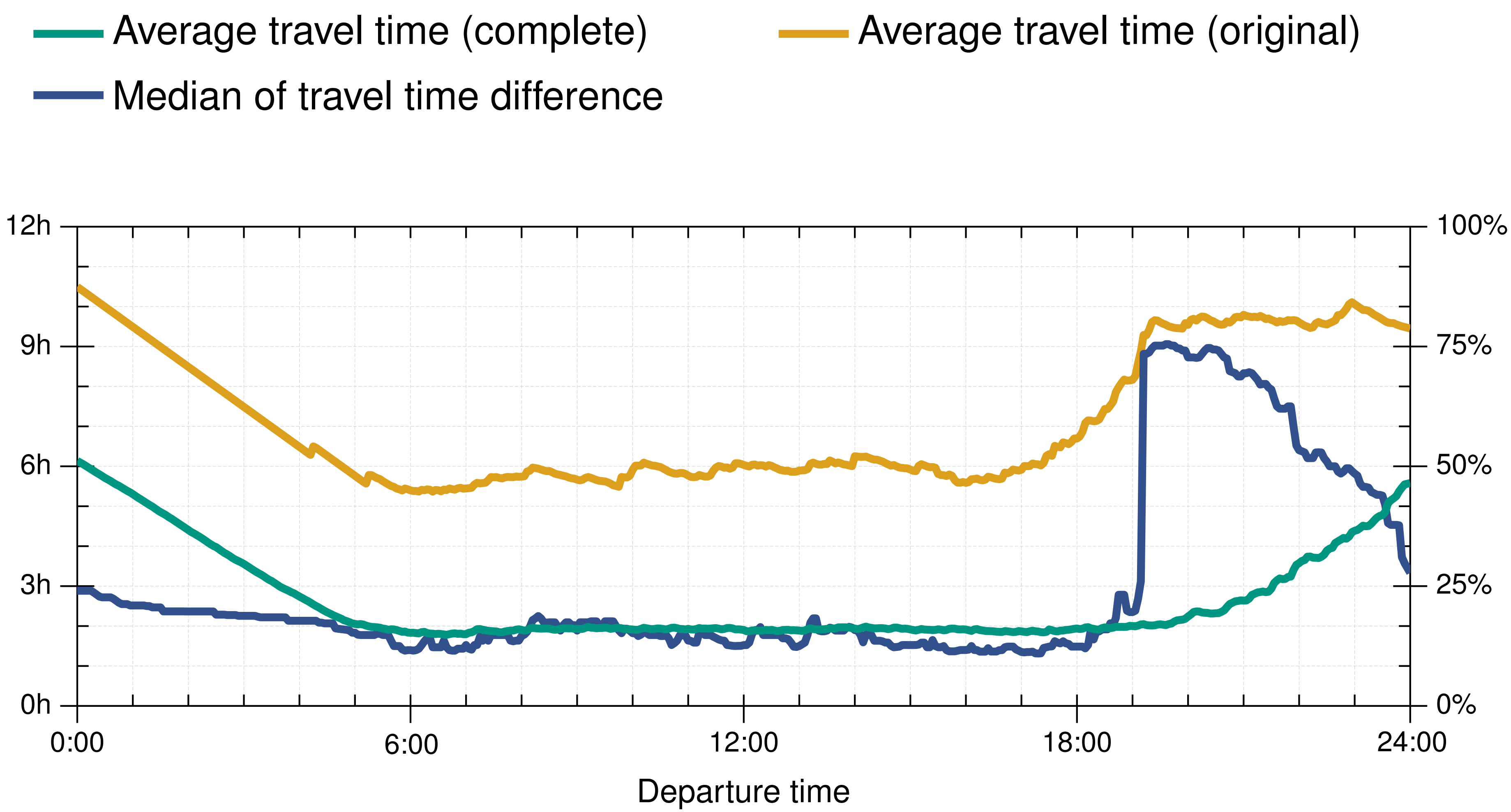
- Switzerland complete vs. Switzerland original (Distance rank 16)



Evaluation – Travel Time Results

Travel time comparison:

- Switzerland complete vs. Switzerland original (Distance rank 16)

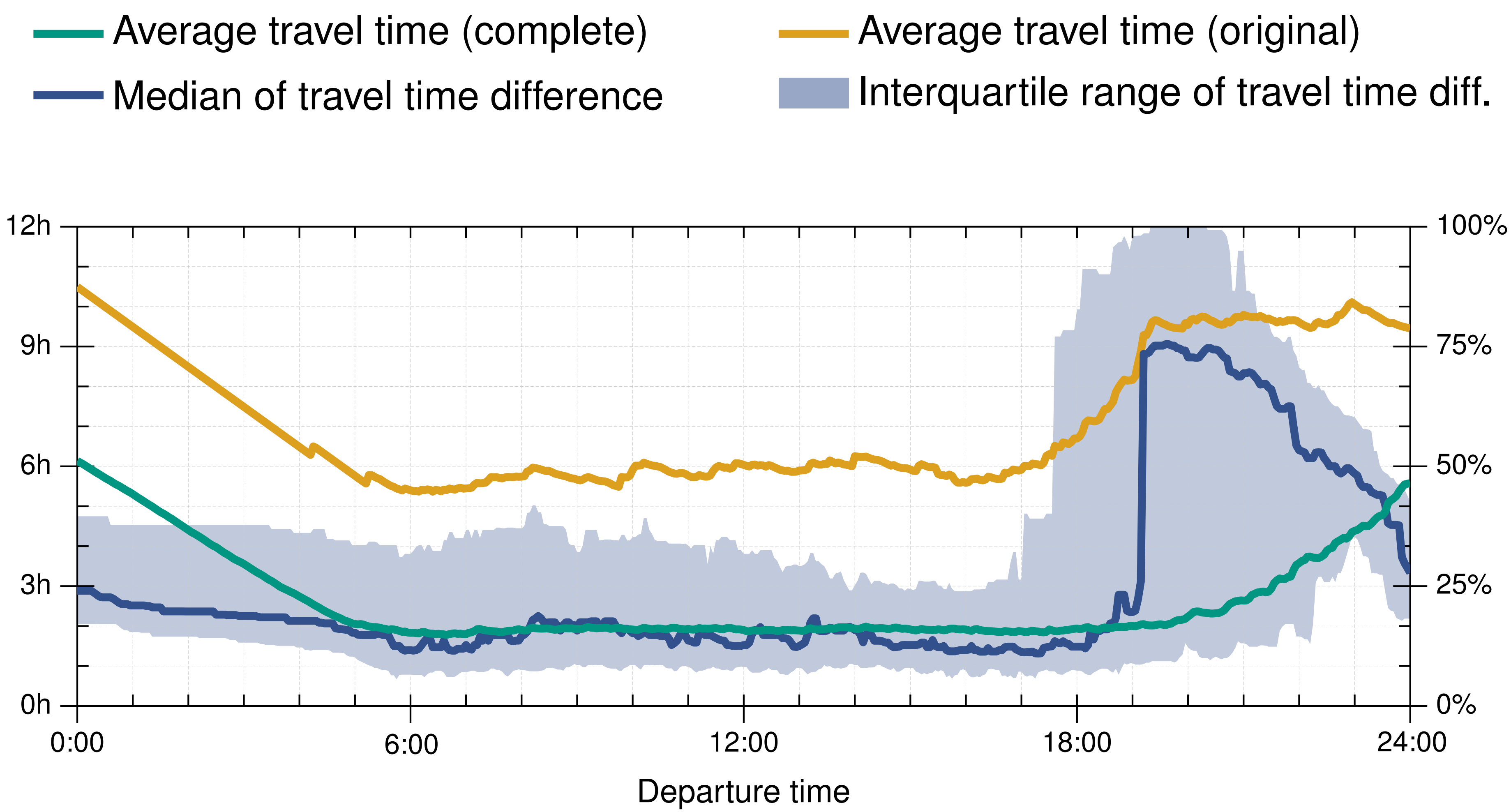


Evaluation – Travel Time Results

Karlsruhe Institute of Technology

Travel time comparison:

- Switzerland complete vs. Switzerland original (Distance rank 16)



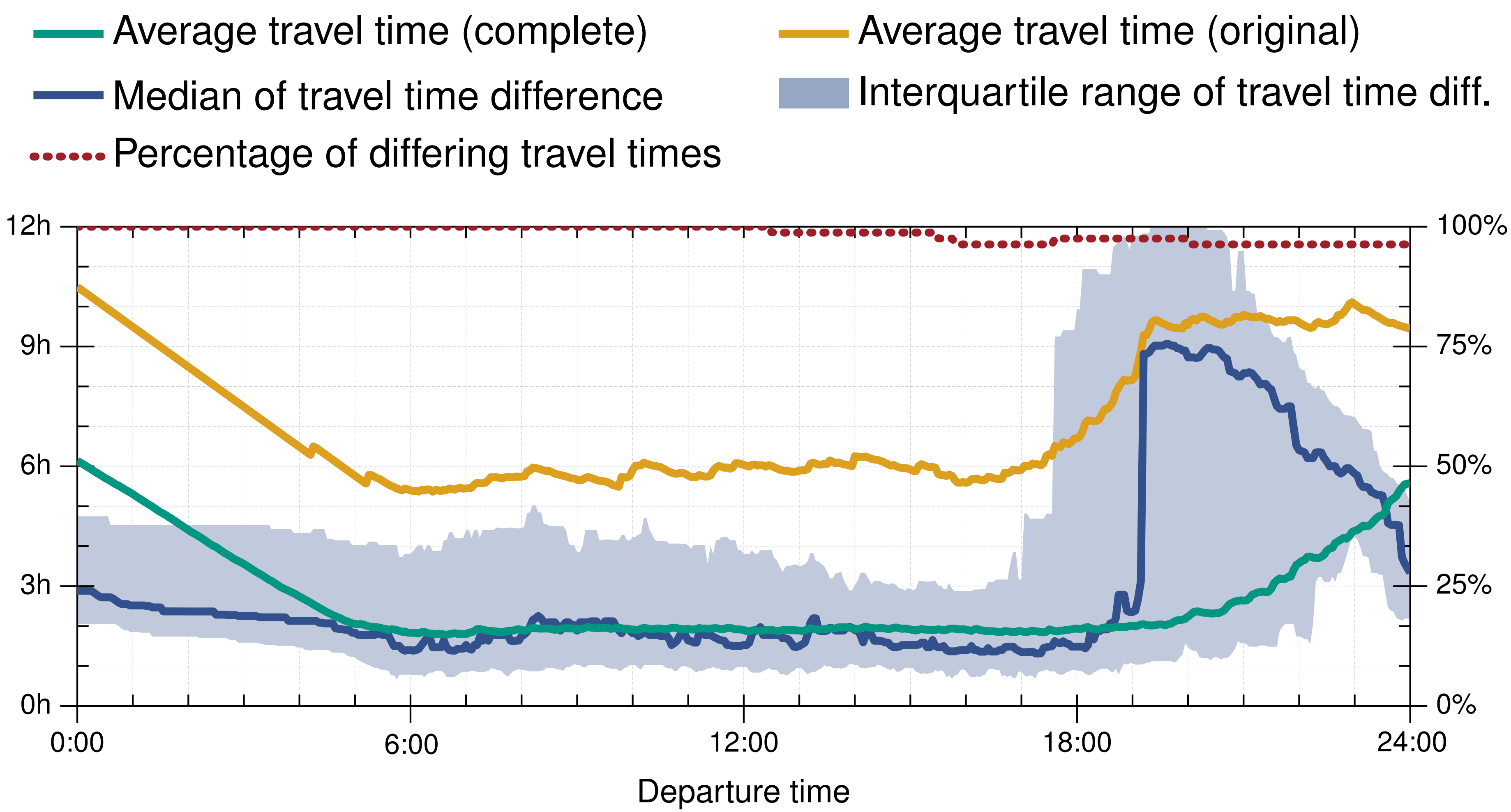
Institute of Theoretical Informatics
Algorithmics Group

Evaluation – Travel Time Results

Karlsruhe Institute of Technology

Travel time comparison:

- Switzerland complete vs. Switzerland original (Distance rank 16)

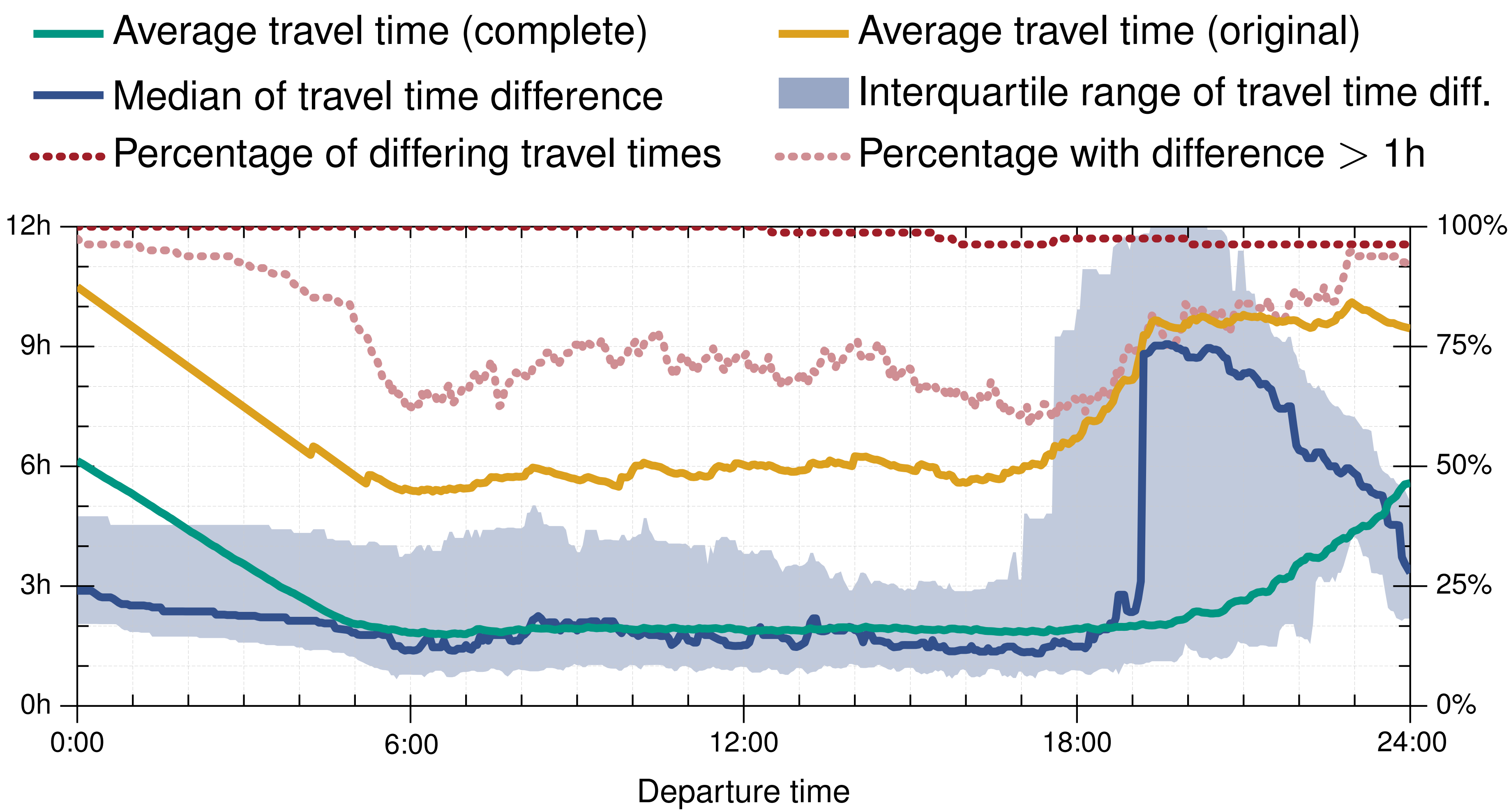


Institute of Theoretical Informatics
Algorithmics Group

Evaluation – Travel Time Results

Travel time comparison:

- Switzerland complete vs. Switzerland original (Distance rank 16)

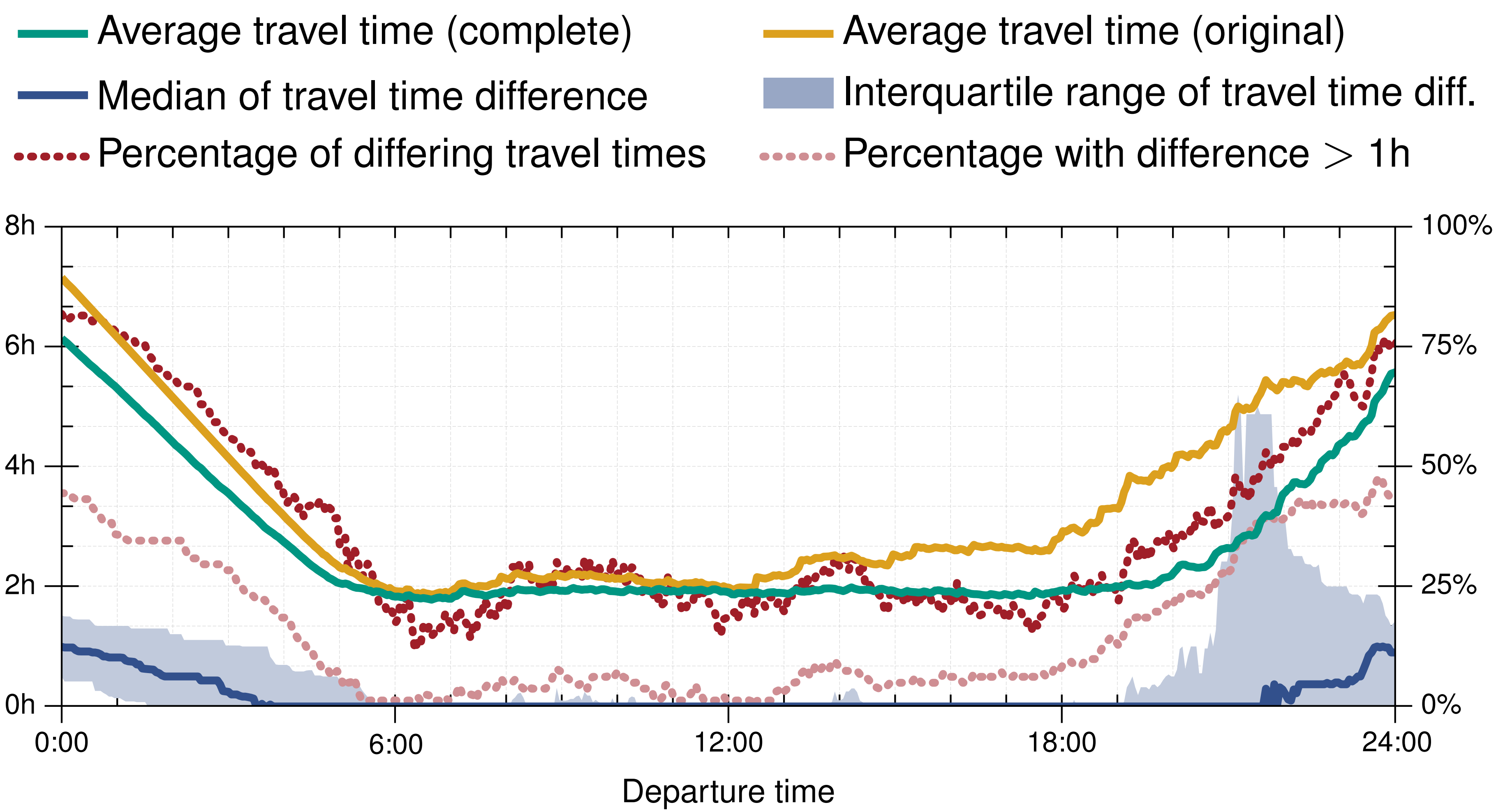


Institute of Theoretical Informatics
Algorithmics Group

Evaluation – Travel Time Results

Travel time comparison:

- Switzerland complete vs. Switzerland partial (Distance rank 16)



Conclusion

Karlsruhe Institute of Technology

Results:

- New public transit profile algorithm for unrestricted walking
- Detailed comparison of common footpath graphs
- Walking has a strong influence on the travel time
- Footpaths specified in the timetable are not sufficient

Future work:

- Initial and final walking vs. walking between trips
- More efficient algorithms for unrestricted walking

Institute of Theoretical Informatics
Algorithmics Group

Conclusion

Results:

- New public transit profile algorithm for unrestricted walking
- Detailed comparison of common footpath graphs
- Walking has a strong influence on the travel time
- Footpaths specified in the timetable are not sufficient

Future work:

- Initial and final walking vs. walking between trips
- More efficient algorithms for unrestricted walking

Thank you for your attention!

Institute of Theoretical Informatics
Algorithmics Group