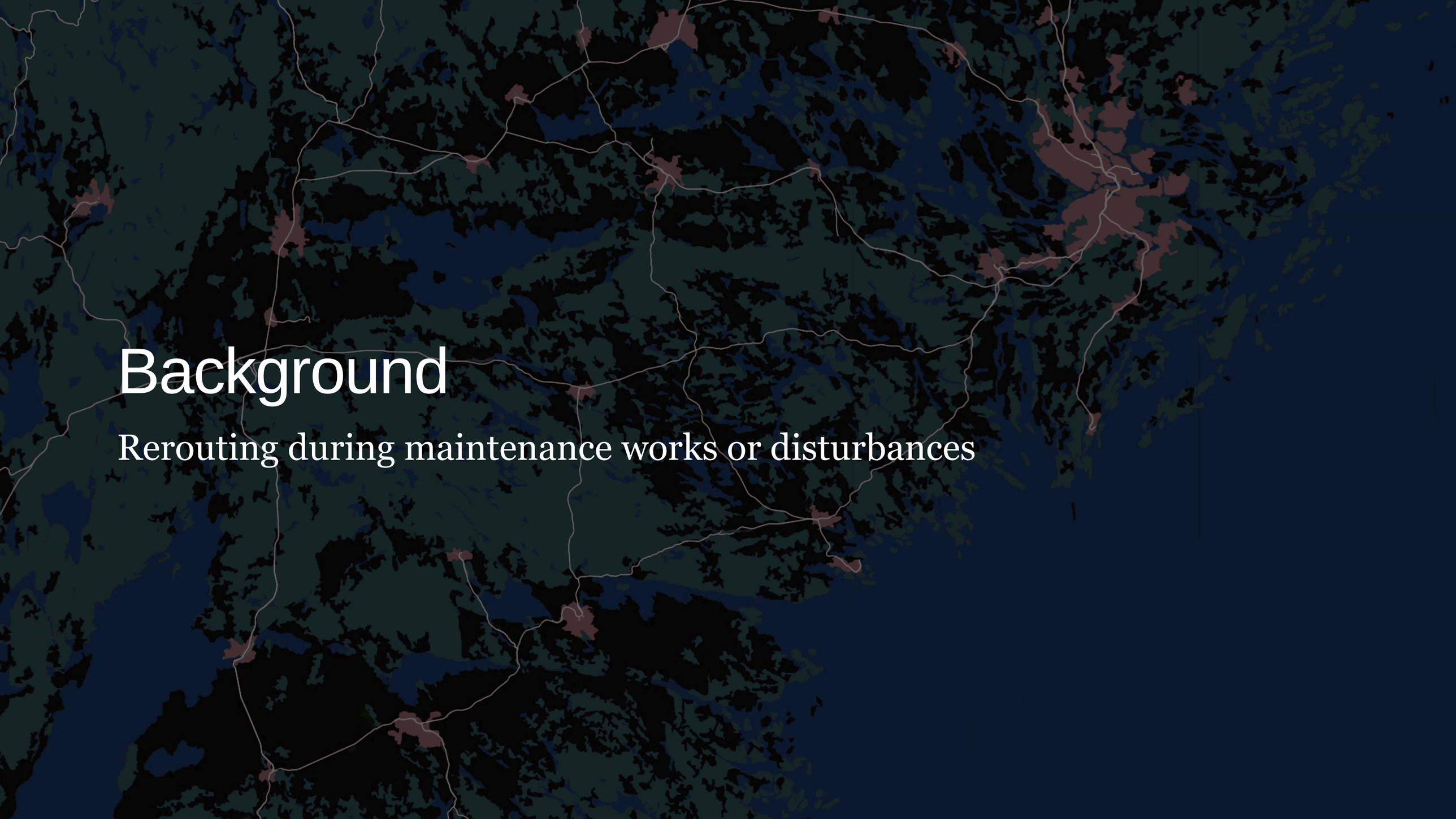
Train Rerouting During Major Disturbances

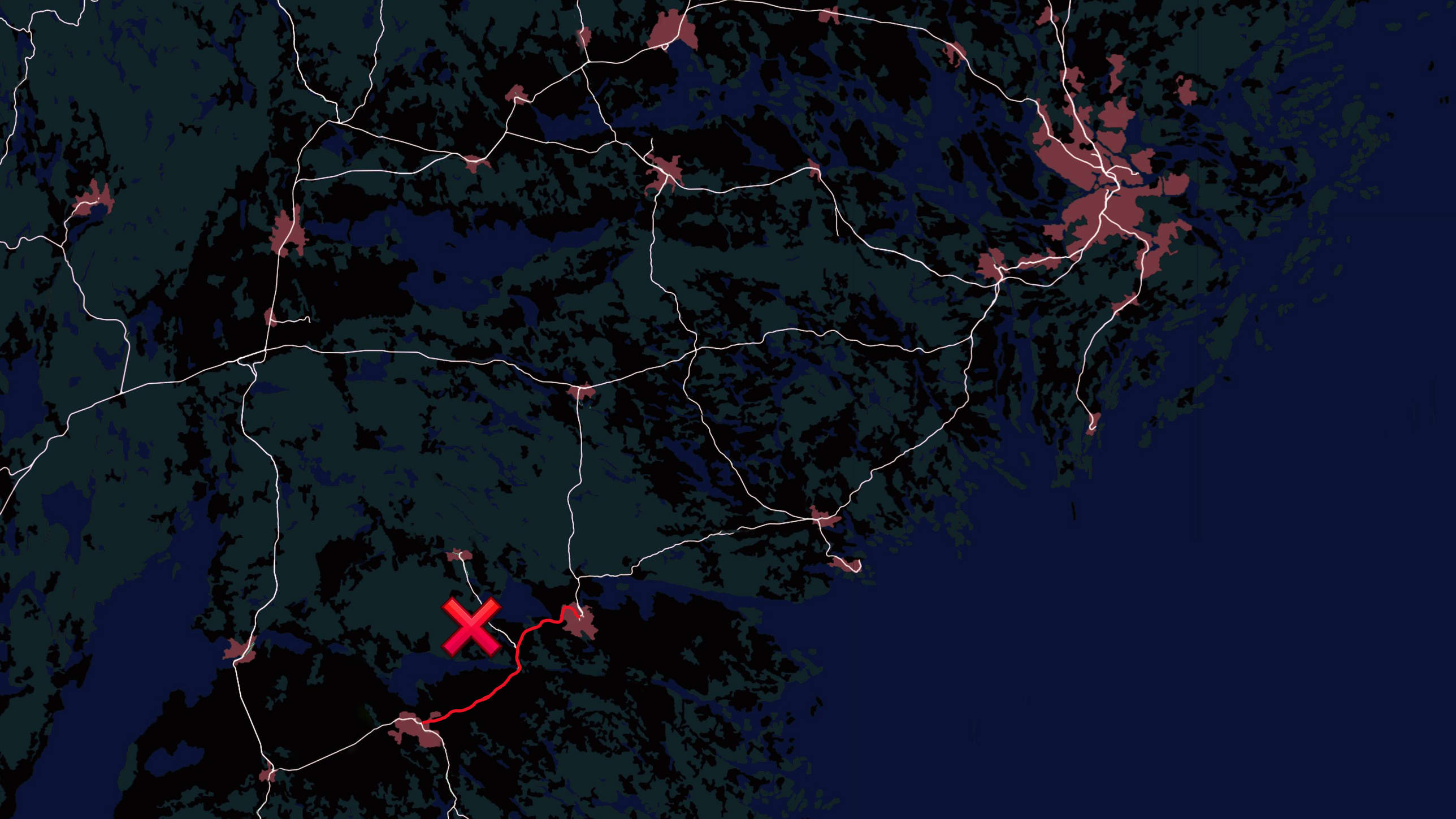
David Dekker, Carl Henrik Häll,
Anders Peterson and Christiane Schmidt

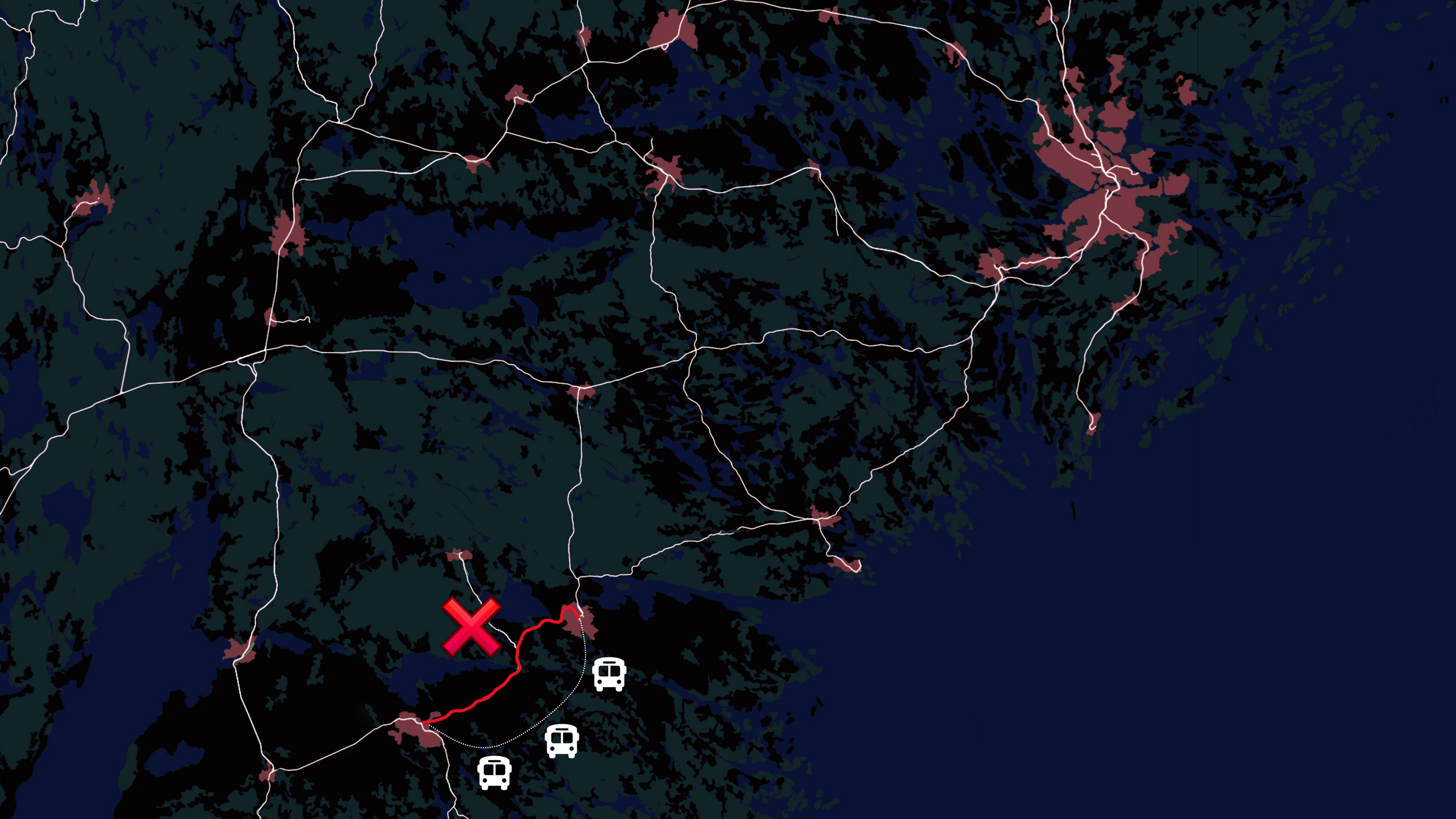
LINKÖPING UNIVERSITY

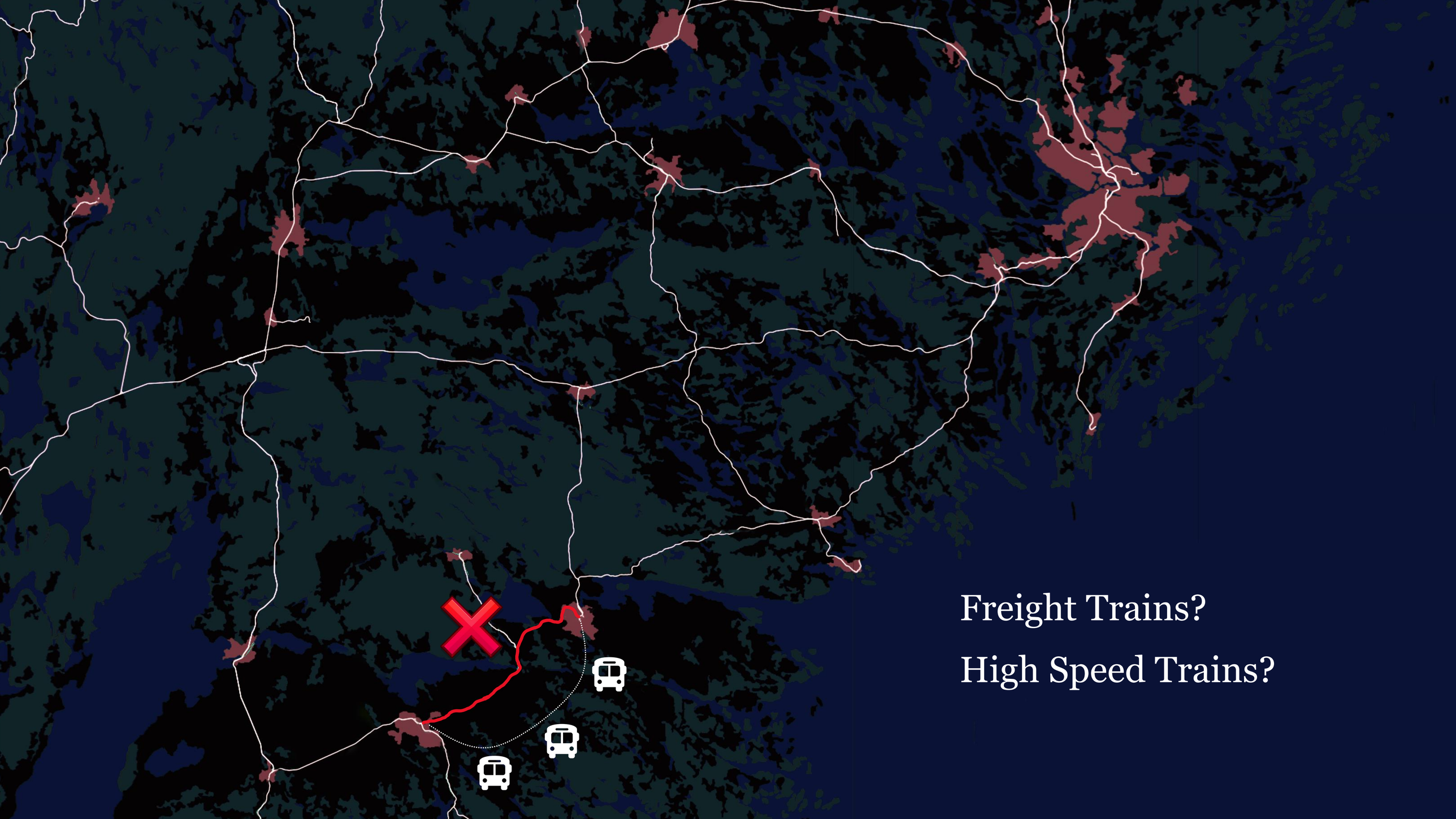


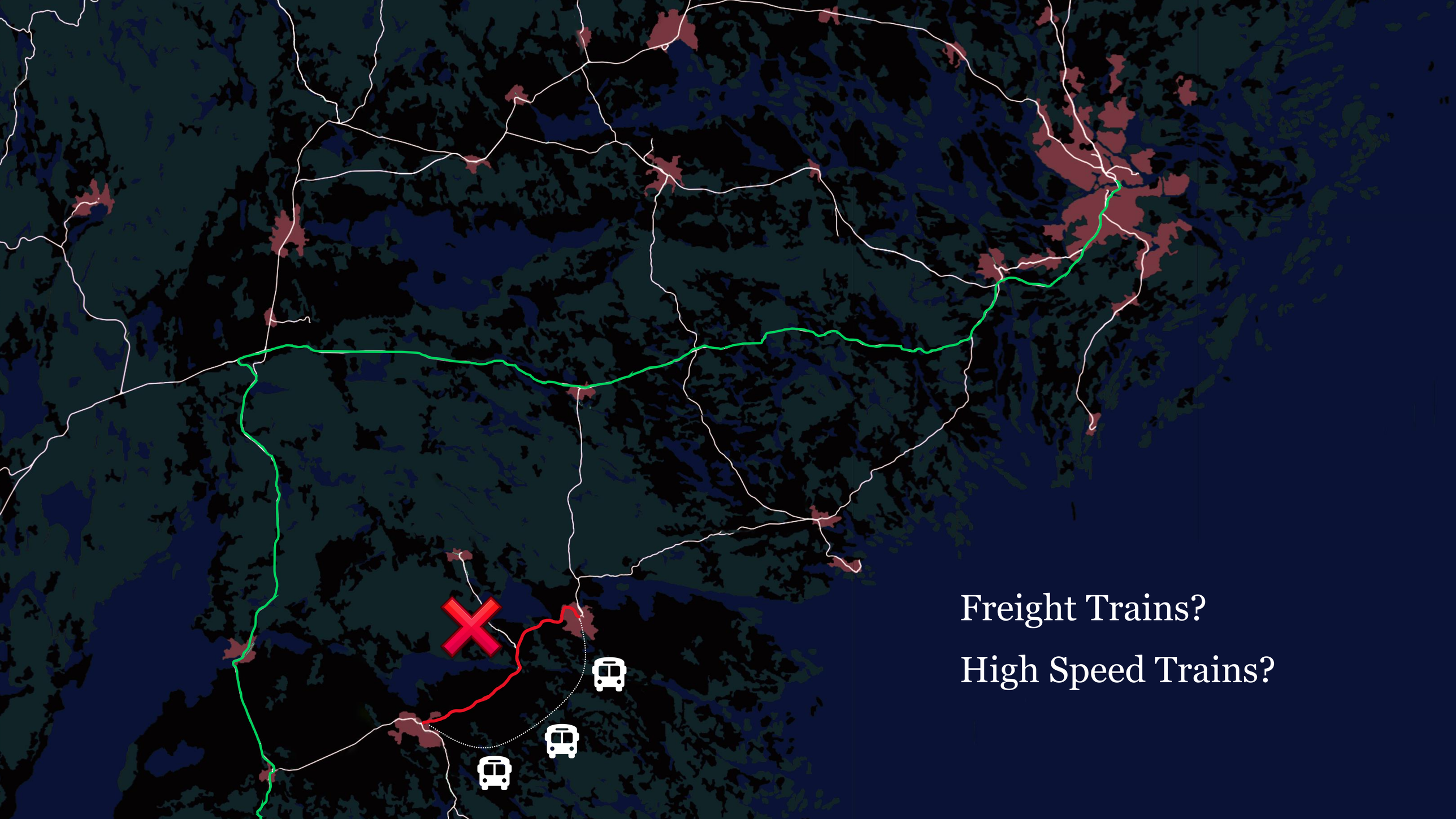
Background

Rerouting during maintenance works or disturbances






Freight Trains?
High Speed Trains?


Freight Trains?
High Speed Trains?

Problem Setting

Input: Railway network with some missing link, an existing timetable and a set of trains that have to be rerouted.

Problem Setting

Input: Railway network with some missing link, an existing timetable and a set of trains that have to be rerouted.

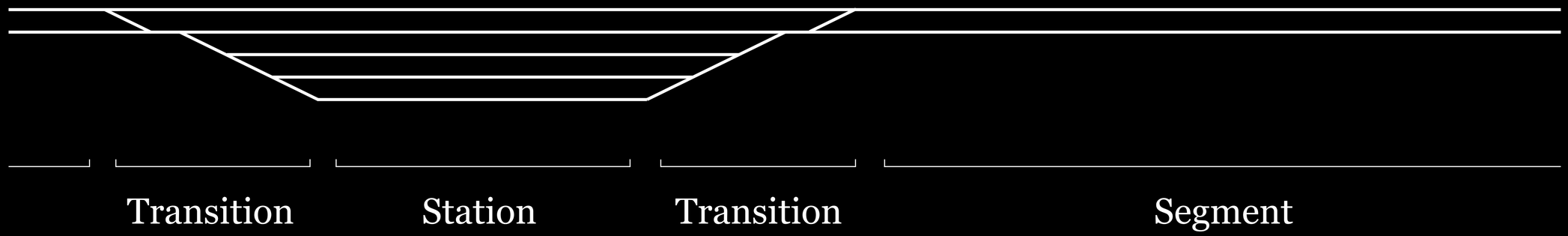
Output: A timetable that includes the rerouted trains.

Problem Setting

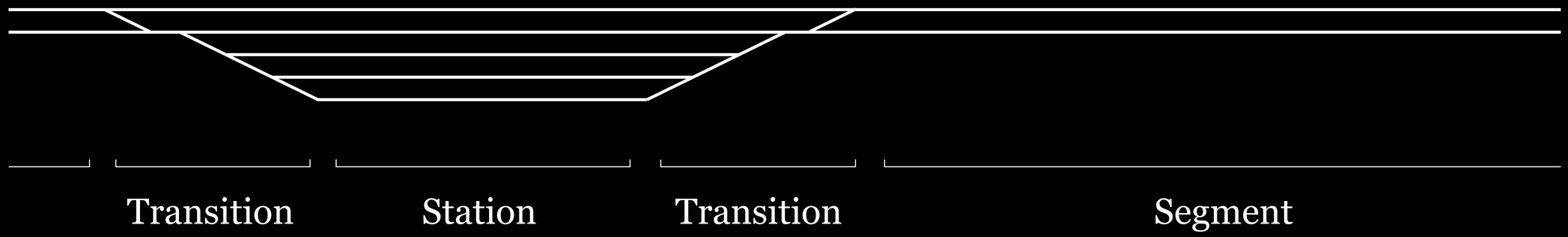
Input:	Railway network with some missing link, an existing timetable and a set of trains that have to be rerouted.
Output:	A timetable that includes the rerouted trains.
Assumptions:	The existing timetable is fixed.

Our Model

Our Model



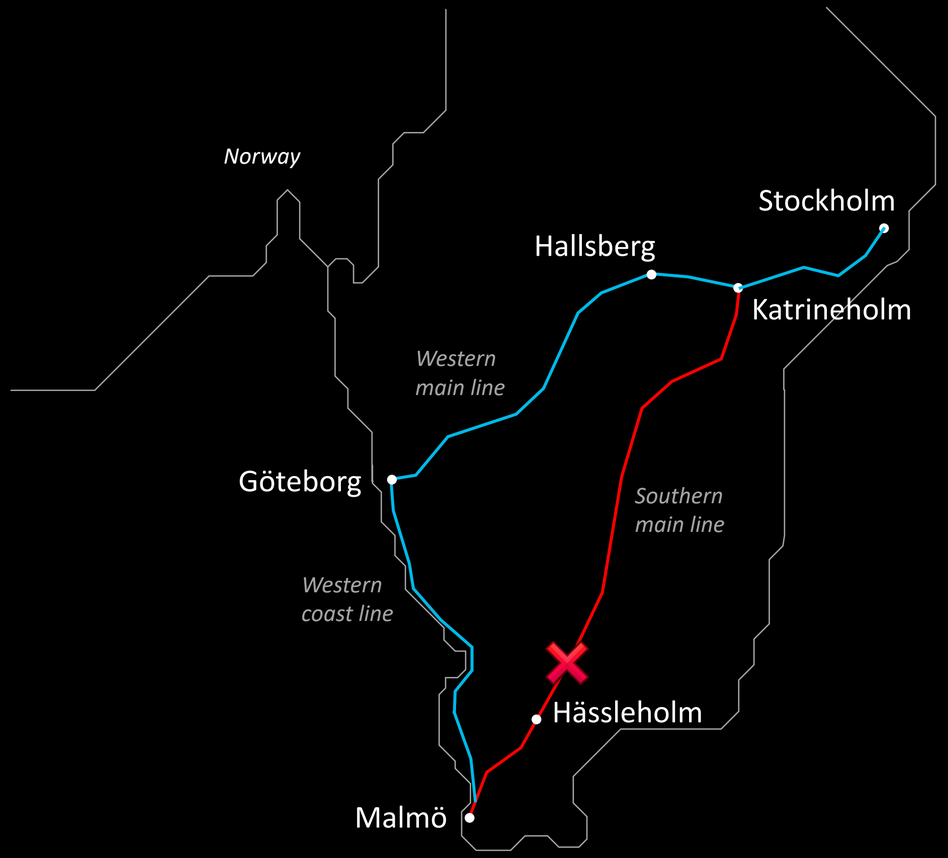
Our Model



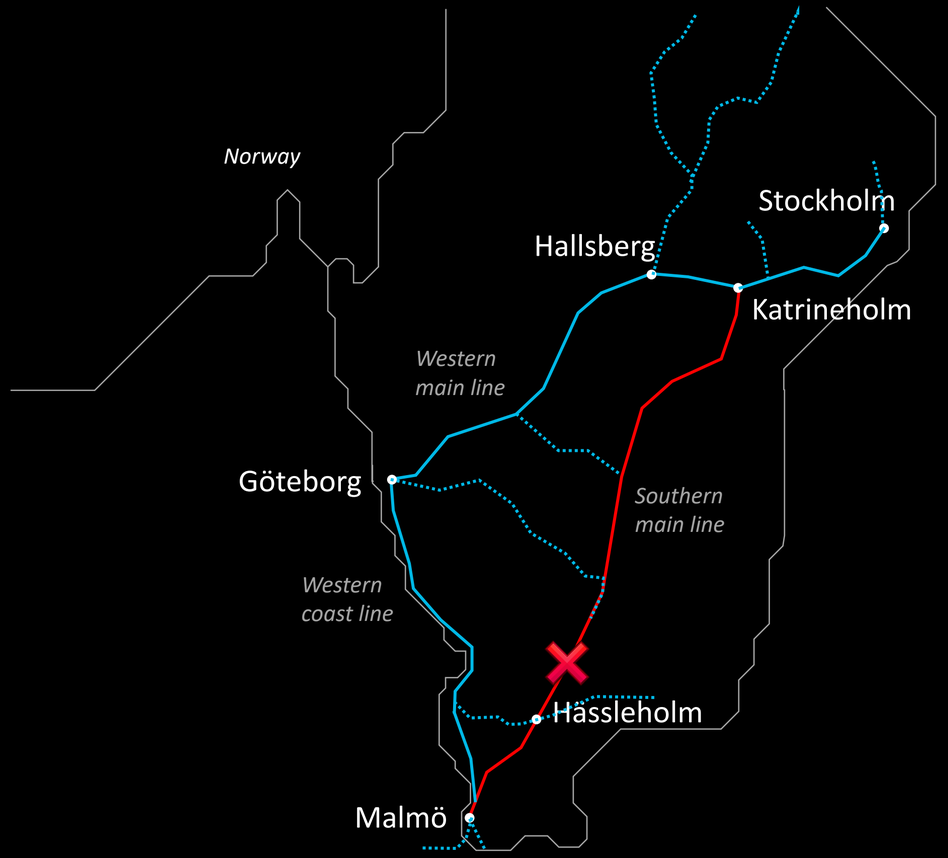
This enables us to use macroscopic data, while also avoiding conflicts on a microscopic level.

The latter requires knowledge of minimum time differences between trains on given tracks; we assume general values here.

Our Model

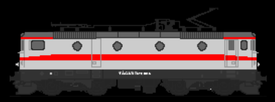

Norway
Stockholm
Hallsberg
Katrineholm
Western
main line
Göteborg
Southern
main line
Western
coast line
Hässleholm
Malmö

Our Model


Norway
Stockholm
Hallsberg
Katrineholm
Western
main line
Göteborg
Southern
main line
Western
coast line
Hässleholm
Malmö

Algorithmic Framework

Algorithmic Framework

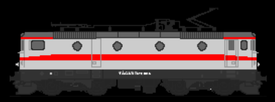


Algorithmic Framework

Stockholm C	07:20
Stockholms södra	07:22
Årstaberg	07:23
…	

Hallsbergs rbg	04:59
Tälle	05:05
Östansjö	05:10
…	

Falköpings C	05:10
Floby	05:21
Källeryd	05:27
…	



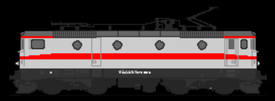
Kristinehamn	05:45
Björneborg	05:58
Strömtorp	06:08
…	

Algorithmic Framework

Cst	07:20
Sst	07:22
Åbe	07:23
…	

Hrbg	04:59
Täl	05:05
Öj	05:10
…	

F	05:10
Fby	05:21
Kä	05:27
…	



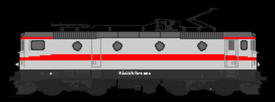
Khn	05:45
Bjb	05:58
Srt	06:08
…	

Algorithmic Framework

Cst	07:20	Cst	06:45	Cst	07:32	Cst	06:45
Sst	07:22	Sst	06:46	Sst	07:34	Sst	06:47
Åbe	07:23	Åbe	06:48	Åbe	07:35	Åbe	06:48
...		...		...		...	

Hrbg	04:59	Hrbg	04:21	Hrbg	05:20	Hrbg	05:46
Täl	05:05	Täl	04:27	Täl	05:26	Täl	05:52
Öj	05:10	Öj	04:32	Öj	05:31	Öj	05:57
...		...		...		...	

F	05:10	F	04:32	F	04:59	F	05:46
Fby	05:21	Fby	04:43	Fby	05:10	Fby	05:57
Kä	05:27	Kä	04:49	Kä	05:16	Kä	06:03
...		...		...		...	



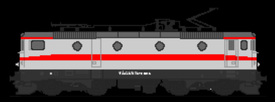
Khn	05:45	Khn	05:12	Khn	06:23	Khn	07:03
Bjb	05:58	Bjb	05:26/34	Bjb	06:36	Bjb	07:16
Srt	06:08	Srt	05:46	Srt	06:46	Srt	07:26
...		...		...		...	

Algorithmic Framework

Cst	07:20	Cst	06:45	Cst	07:32	Cst	06:45
Sst	07:22	Sst	06:46	Sst	07:34	Sst	06:47
Åbe	07:23	Åbe	06:48	Åbe	07:35	Åbe	06:48
...		...		...		...	

Hrbg	04:59	Hrbg	04:21	Hrbg	05:20	Hrbg	05:46
Täl	05:05	Täl	04:27	Täl	05:26	Täl	05:52
Öj	05:10	Öj	04:32	Öj	05:31	Öj	05:57
...		...		...		...	

F	05:10	F	04:32	F	04:59	F	05:46
Fby	05:21	Fby	04:43	Fby	05:10	Fby	05:57
Kä	05:27	Kä	04:49	Kä	05:16	Kä	06:03
...		...		...		...	



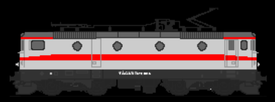
Khn	05:45	Khn	05:12	Khn	06:23	Khn	07:03
Bjb	05:58	Bjb	05:26/34	Bjb	06:36	Bjb	07:16
Srt	06:08	Srt	05:46	Srt	06:46	Srt	07:26
...		...		...		...	

Algorithmic Framework

Cst	**07:20**	Cst	06:45	Cst	07:32	Cst	06:45
Sst	**07:22**	Sst	06:46	Sst	07:34	Sst	06:47
Åbe	**07:23**	Åbe	06:48	Åbe	07:35	Åbe	06:48
…		…		…		…	

Hrbg	04:59	**Hrbg**	**04:21**	Hrbg	05:20	Hrbg	05:46
Täl	05:05	**Täl**	**04:27**	Täl	05:26	Täl	05:52
Öj	05:10	**Öj**	**04:32**	Öj	05:31	Öj	05:57
…		**…**		…		…	

F	**05:10**	F	04:32	F	04:59	F	05:46
Fby	**05:21**	Fby	04:43	Fby	05:10	Fby	05:57
Kä	**05:27**	Kä	04:49	Kä	05:16	Kä	06:03
…		…		…		…	



Khn	05:45	Khn	05:12	**Khn**	**06:23**	Khn	07:03
Bjb	05:58	Bjb	05:26/34	**Bjb**	**06:36**	Bjb	07:16
Srt	06:08	Srt	05:46	**Srt**	**06:46**	Srt	07:26
…		…		**…**		…	

Algorithmic Framework

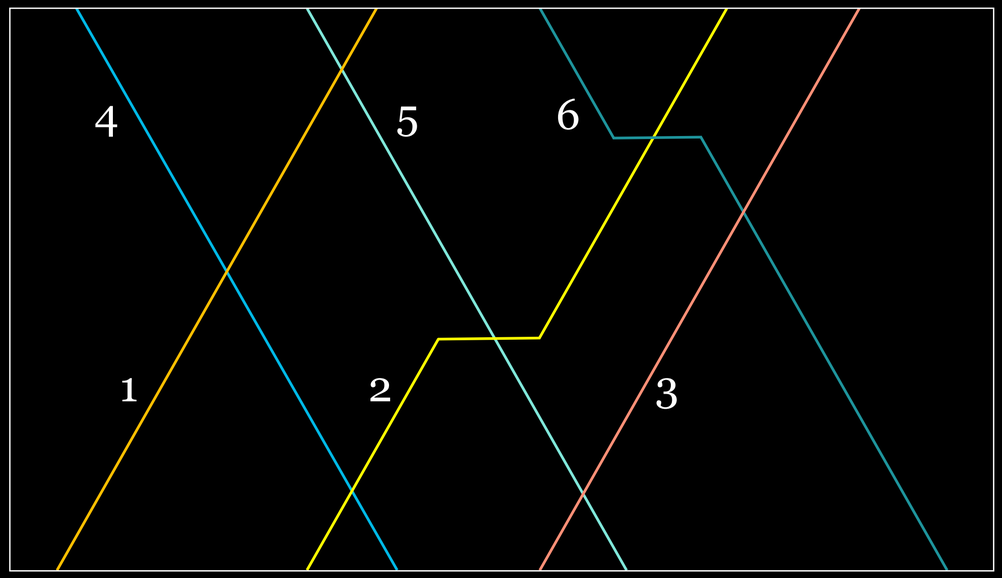
: options 1, 2 and 3.

: options 4, 5 and 6.

Algorithmic Framework

: options 1, 2 and 3.

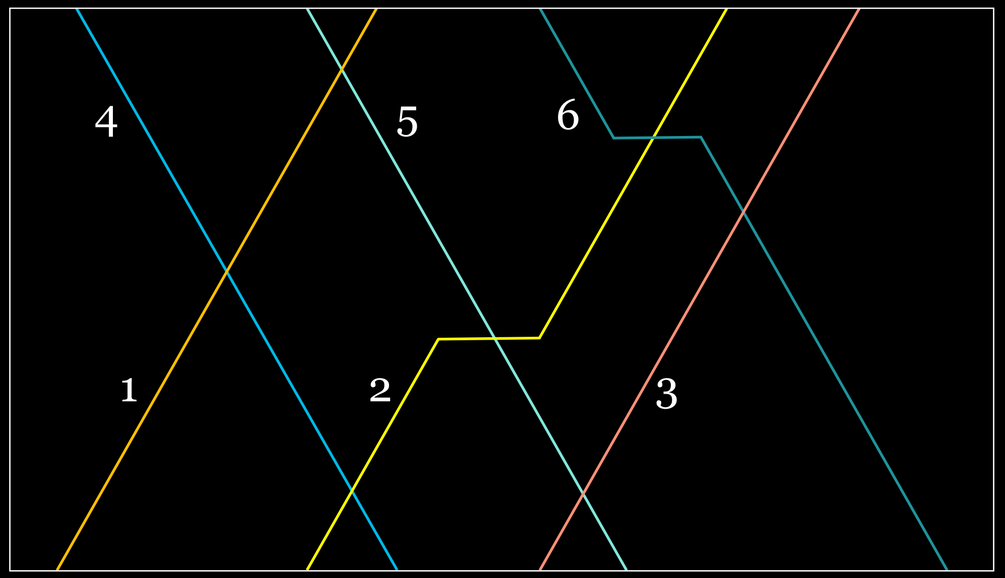
: options 4, 5 and 6.

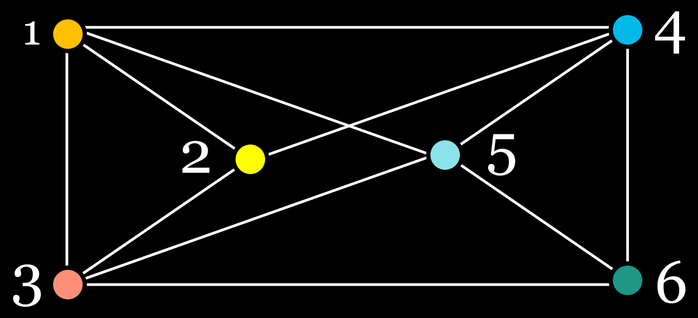


Algorithmic Framework

: options 1, 2 and 3.

: options 4, 5 and 6.





Algorithmic Framework

General idea:

Generate multiple possible paths for each train that has to be rerouted, and associate a cost to each path.

Algorithmic Framework

General idea:

Generate multiple possible paths for each train that has to be rerouted, and associate a cost to each path.

Then select one path for each train, such that the cost is minimized, without creating conflicts between pairs of paths.

Cost Function

Three main parameters:

Cost Function

Three main parameters:

1. The total travel time.

Cost Function

Three main parameters:

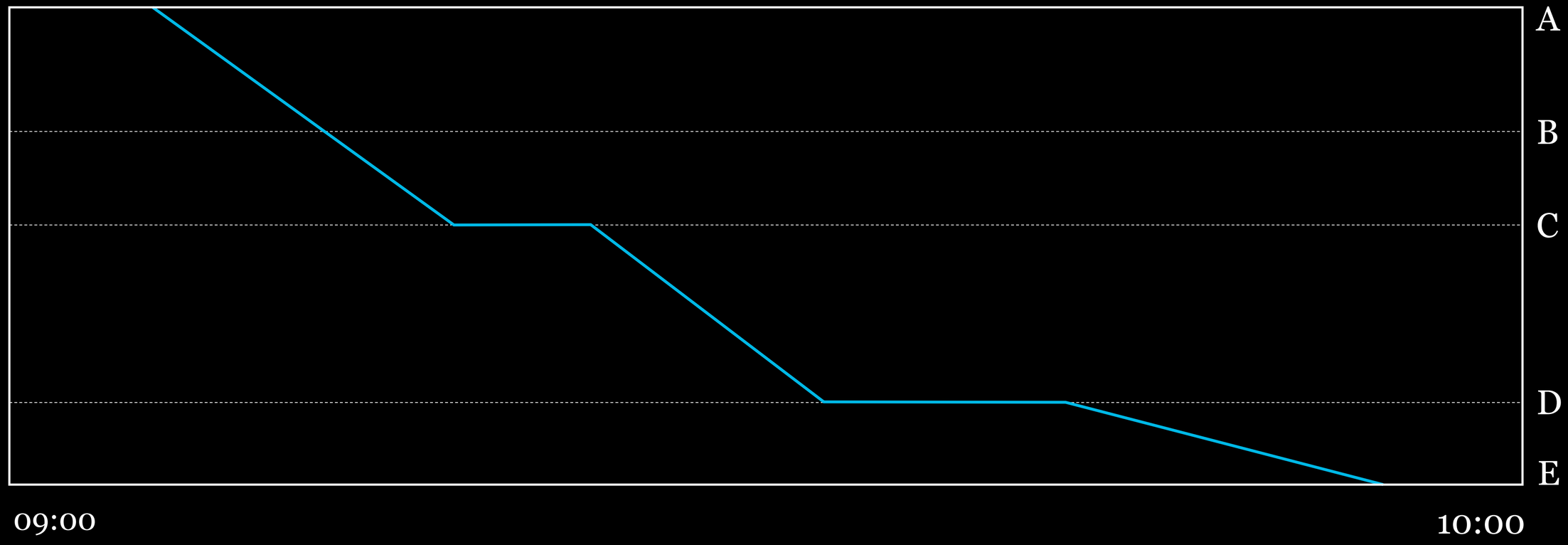
1. The total travel time.
2. Difference with requested departure time.

Cost Function

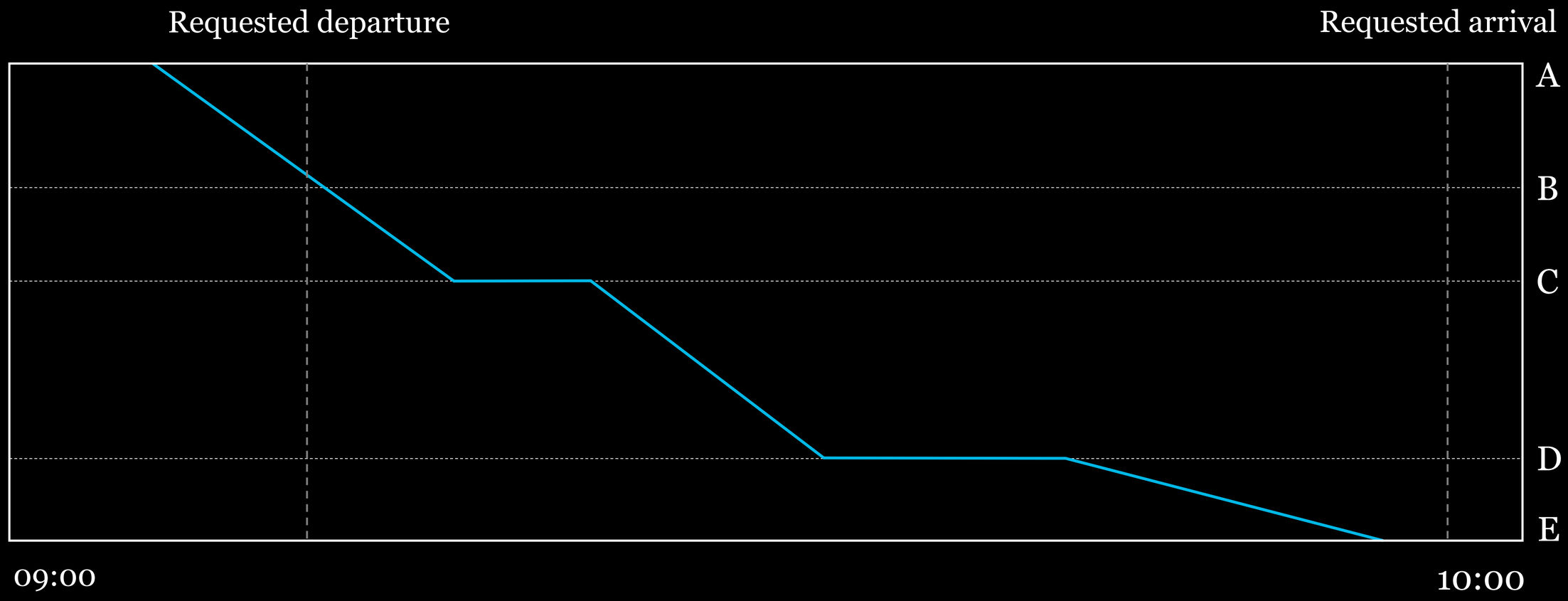
Three main parameters:

1. The total travel time.
2. Difference with requested departure time.
3. Difference with requested arrival time.

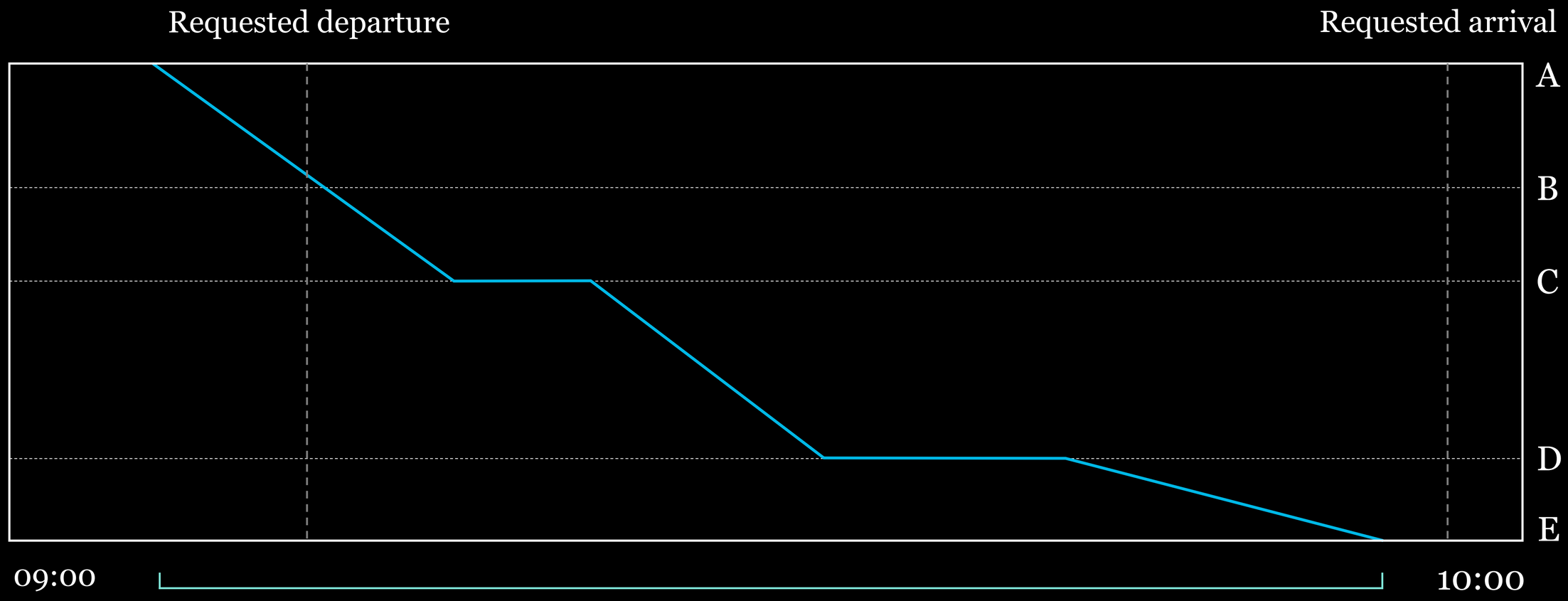
Cost Function



Cost Function

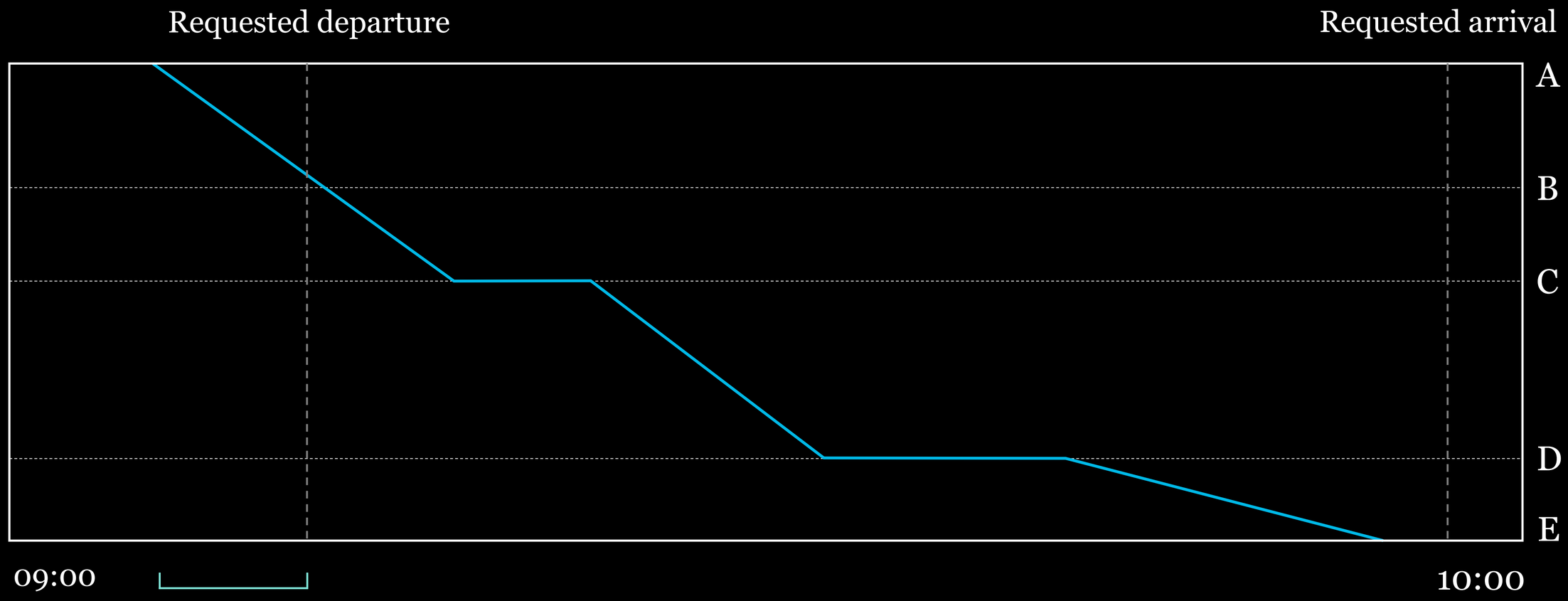


Cost Function



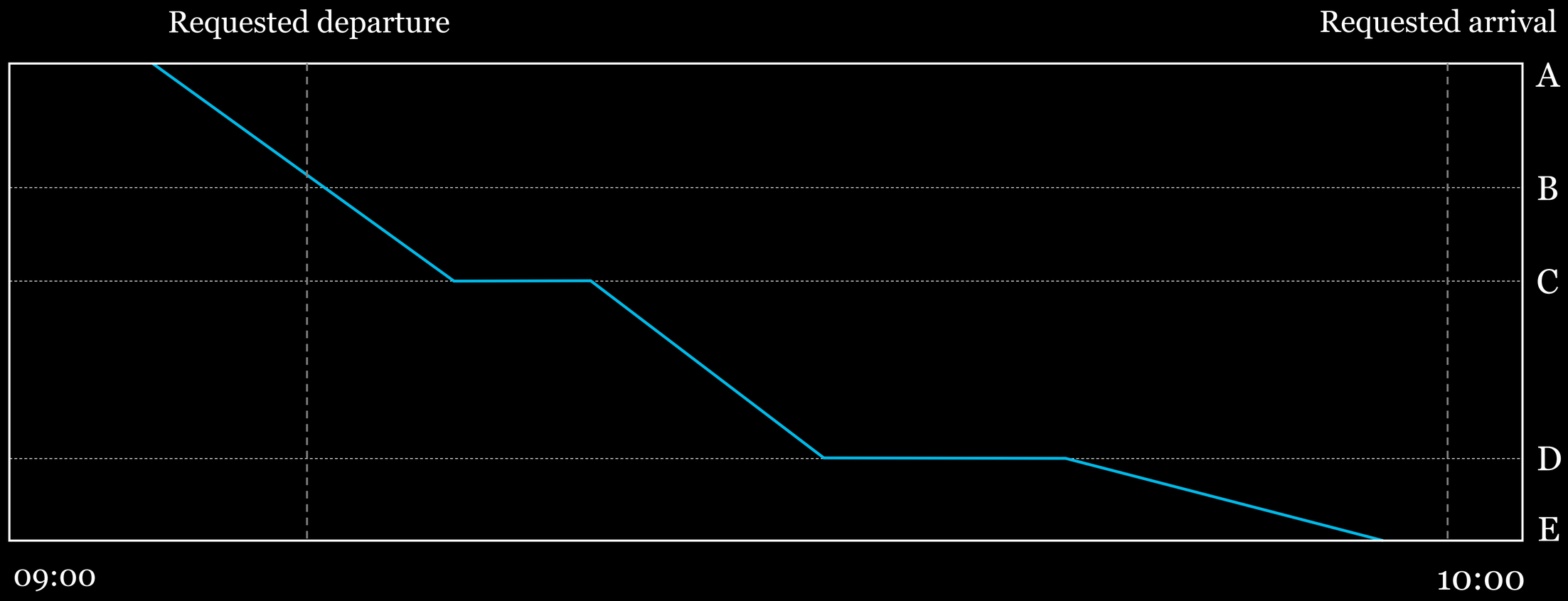
Travel time: 45 minutes.

Cost Function



Travel time: 45 minutes. Difference with requested departure: 6 minutes.

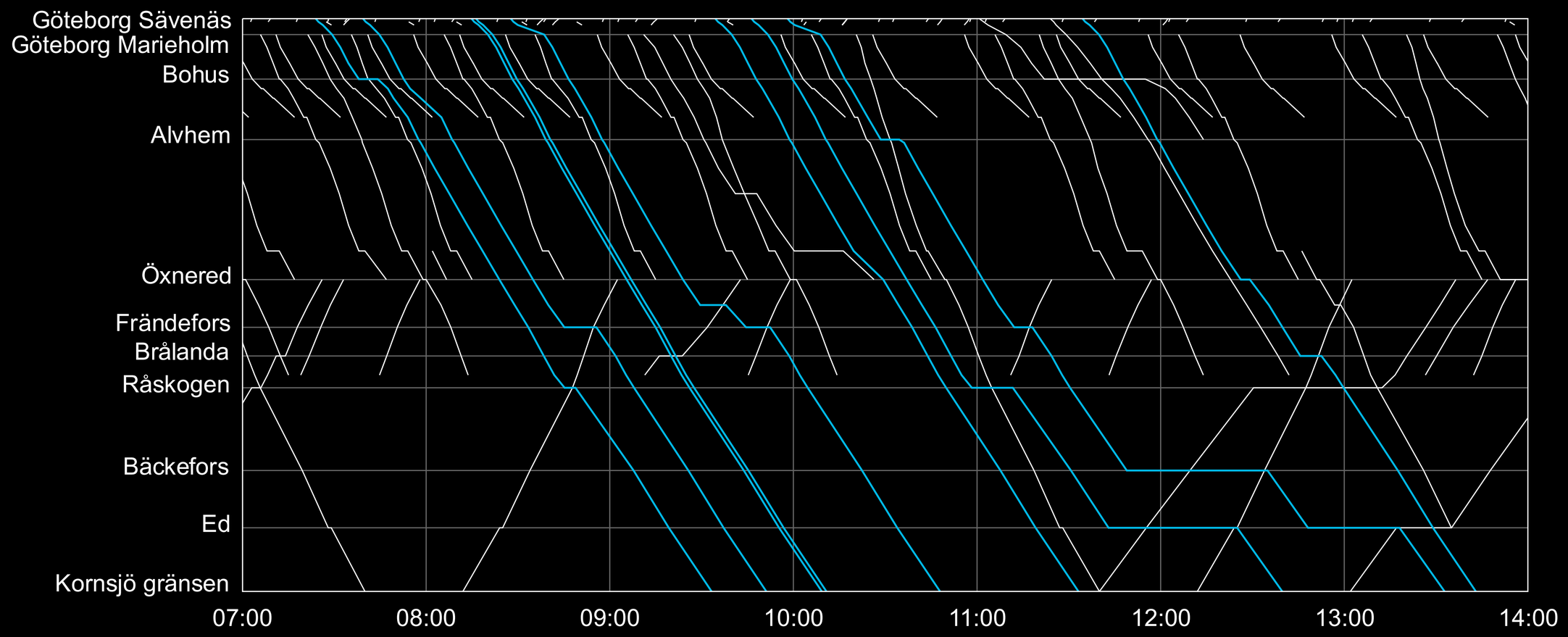
Cost Function



Travel time: 45 minutes. Difference with requested departure: 6 minutes. With arrival: 0 minutes.

Greedy Algorithm

Greedy Algorithm



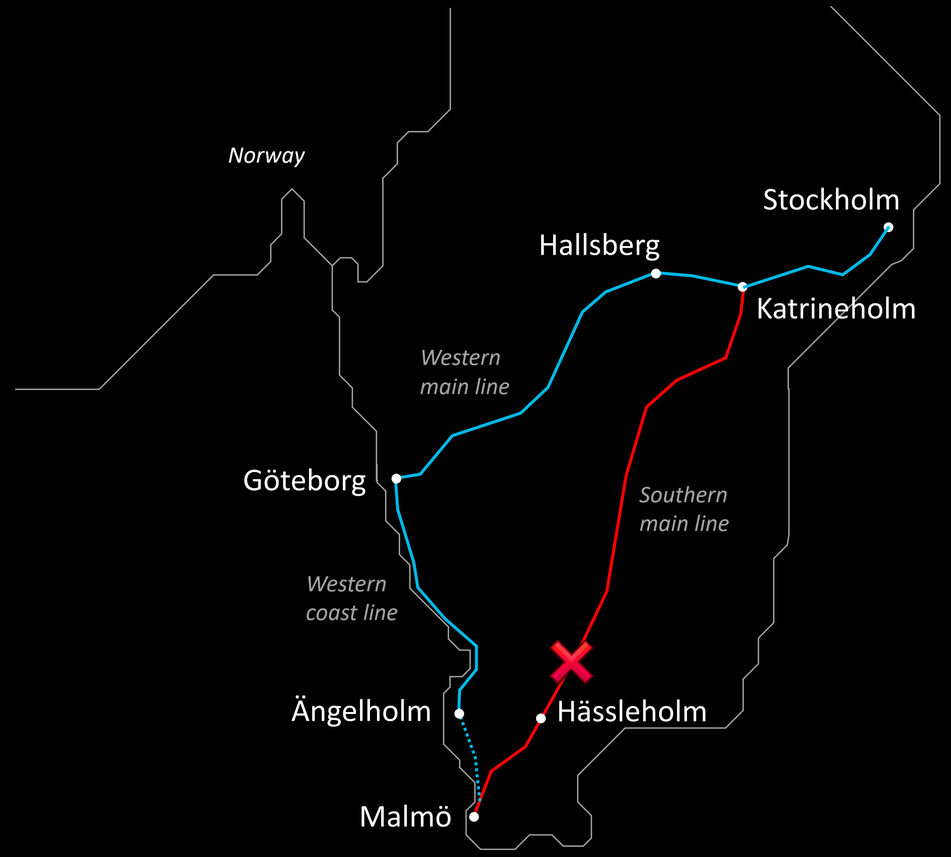
Case Study

- 50 trains to be rerouted via Göteborg during a weekend: 660 km long line.
- For now, we only consider the double track part north of Ängelholm.
- Cancellation is allowed.

Case Study

Solution cost for different methods:
(lower is better)

- 59.0: lower bound, where every train gets their optimal path, without solving conflicts between them.
- 63.2: greedy, passenger trains first.
- 67.3: two-phase independent set.
- 67.4: greedy, freight trains first.
- 75.8: independent set approach.



Future work

- Implementing iterative path generation to improve the independent set approach.
- Shifting focus to TTR: how should pre-planned freight paths look like?


Thank you!