Effectiveness of dynamic reordering and rerouting of trains in a complicated and densely occupied station area

Francesco Corman, Andrea D’Ariano, Marco Pranzo, Ingo A. Hansen

RailZurich09

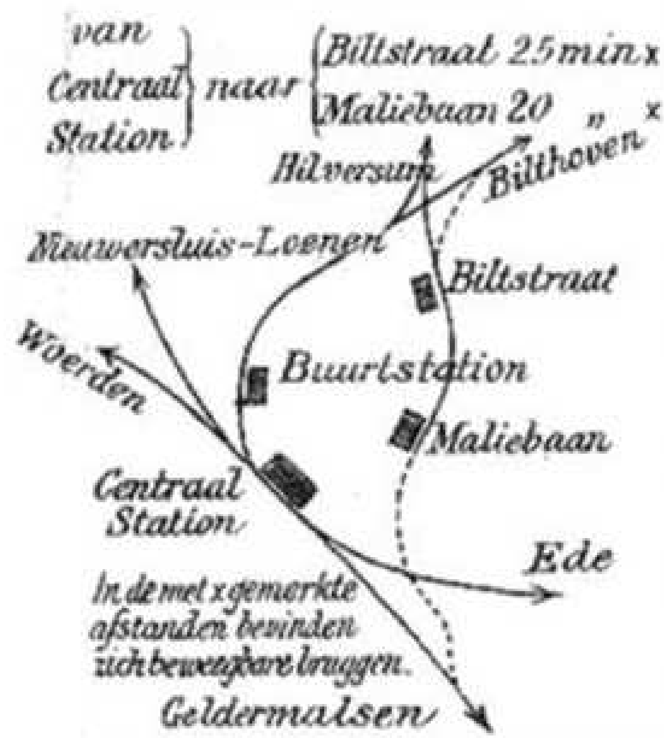
Outline

Introduction

Models, Algorithms

Experiments

Conclusions and future work



Introduction

Introduction

Unforeseen events perturb the planned operations, resulting in delays, missed transfer connections, cancelled train services…

Traffic dispatchers normally rely on their experience, while Conflict Detection and Resolution (CDR) systems result in improved control measures by

- precise forecast of the dynamics of traffic,
- detection of conflicts between train routes,
- optimization of timings, orders and routes,
- coordination of train speeds

Control of dense traffic areas

Investigate the possibility of handling hard instances with dense traffic in complex station areas and multiple operational constraints

Design and implement coupling with databases from ProRail and NS

Develop models and algorithms to ensure proper formulation of railway instances in station areas of increased complexity

Test the algorithms with realization data

Data model

Infrastructure layout

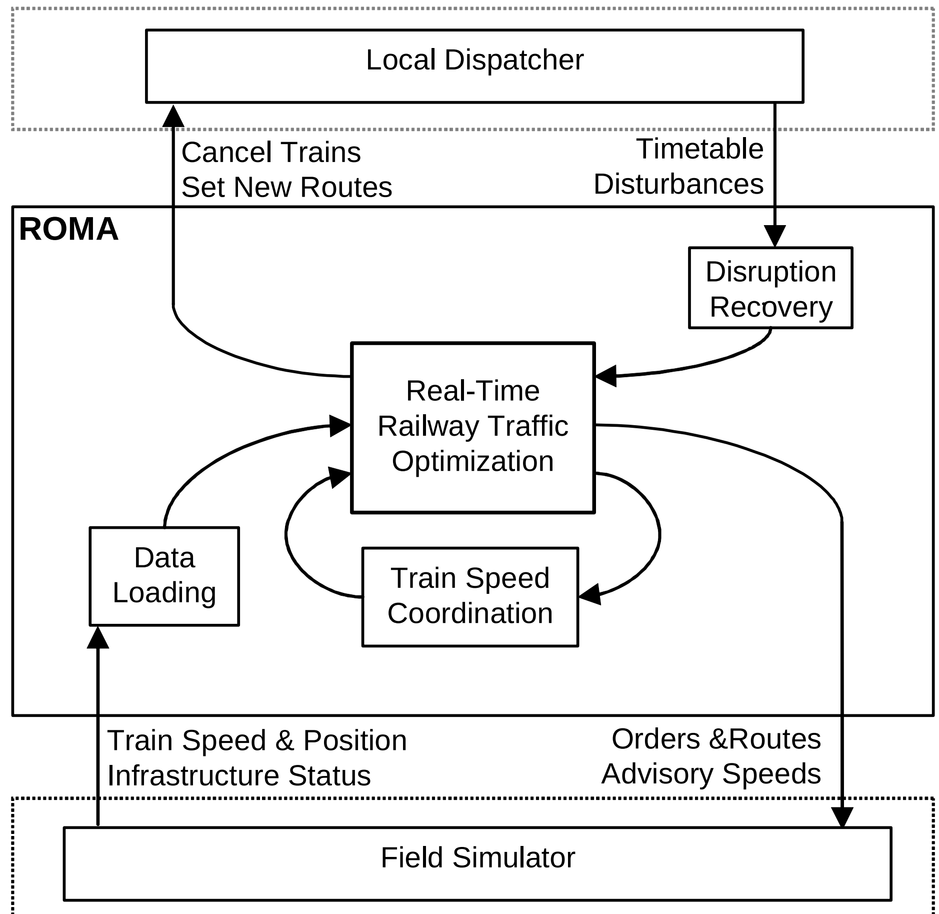
Timetable (times, routes)

Acceleration and braking patterns

System architecture

Real-Time Railway Traffic Optimization is the core of the system that:

- simulate the traffic flow in the network,
- choose optimal train orders and routes



Railway traffic optimization

Conflict situation:
Some trains claim a block section, but only one must occupy the block section at a time.

Which one passes first?

Optimization model

A job shop scheduling problem formulation is used, every block section is considered as a single server with blocking and other constraints

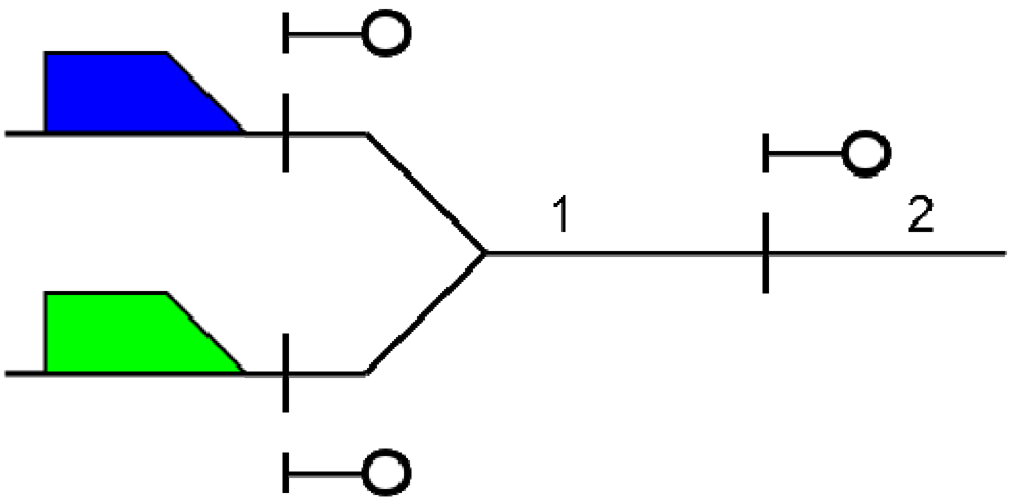
The problem of finding the optimal order for the trains over each block section (machine) is *NP-Hard*

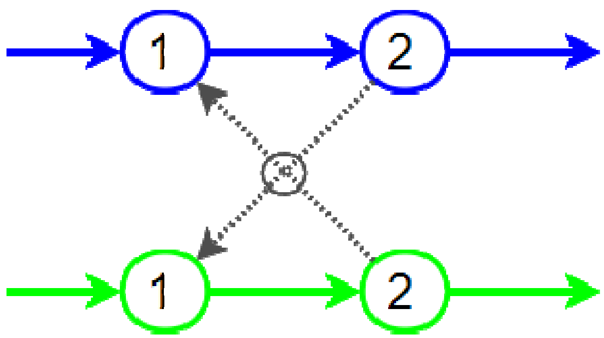
Having dense traffic at station interlocking areas with multiple incompatible routes leads to an increased complexity for the optimization procedure

Alternative graph formulation

Fixed constraints between successive events (running times between two signals)

Alternative constraints for events that must not happen at the same time (orders between trains)





Overview of solving procedures

Fixed timetable orders, priority rules, first come first served rule, look-ahead rules…

We consider all possible CDR solutions (~ 10^3000) in order to report the best solution

A good lower bound based on Jackson Preemptive schedule is adopted by relaxing some constraints

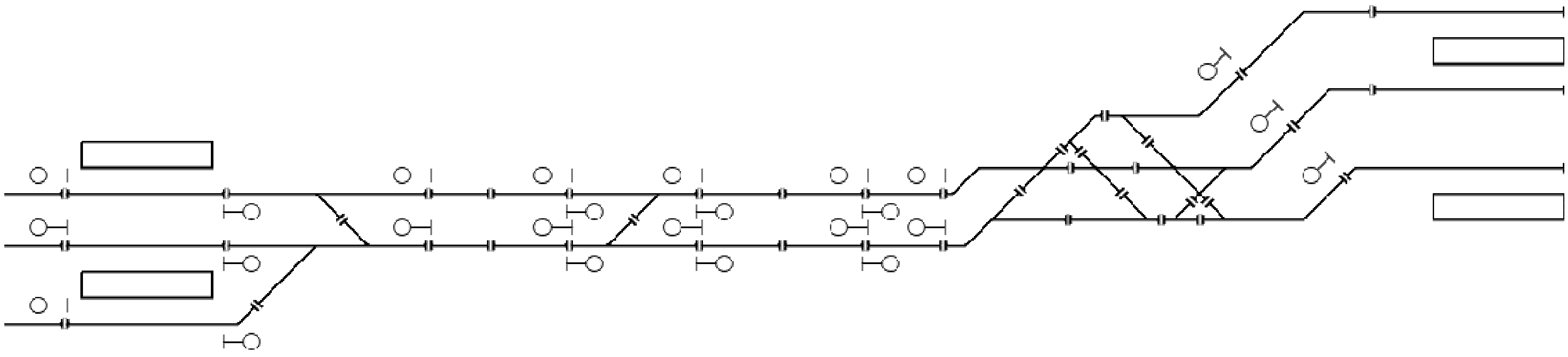
The dispatching rules are used as upper bounds

An exhaustive search procedure (branch and bound) finds near-optimal solutions within a given (short) computation time

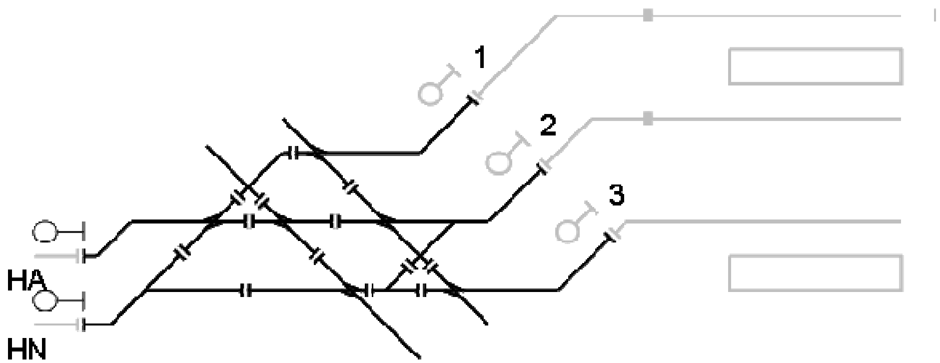
Simplifying

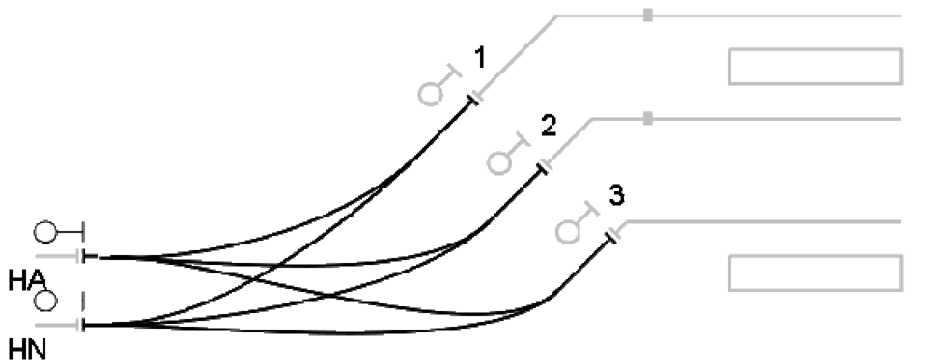
Too many variables and too many railway constraints

How to model properly the situation? And how to include only the "necessary" constraints?



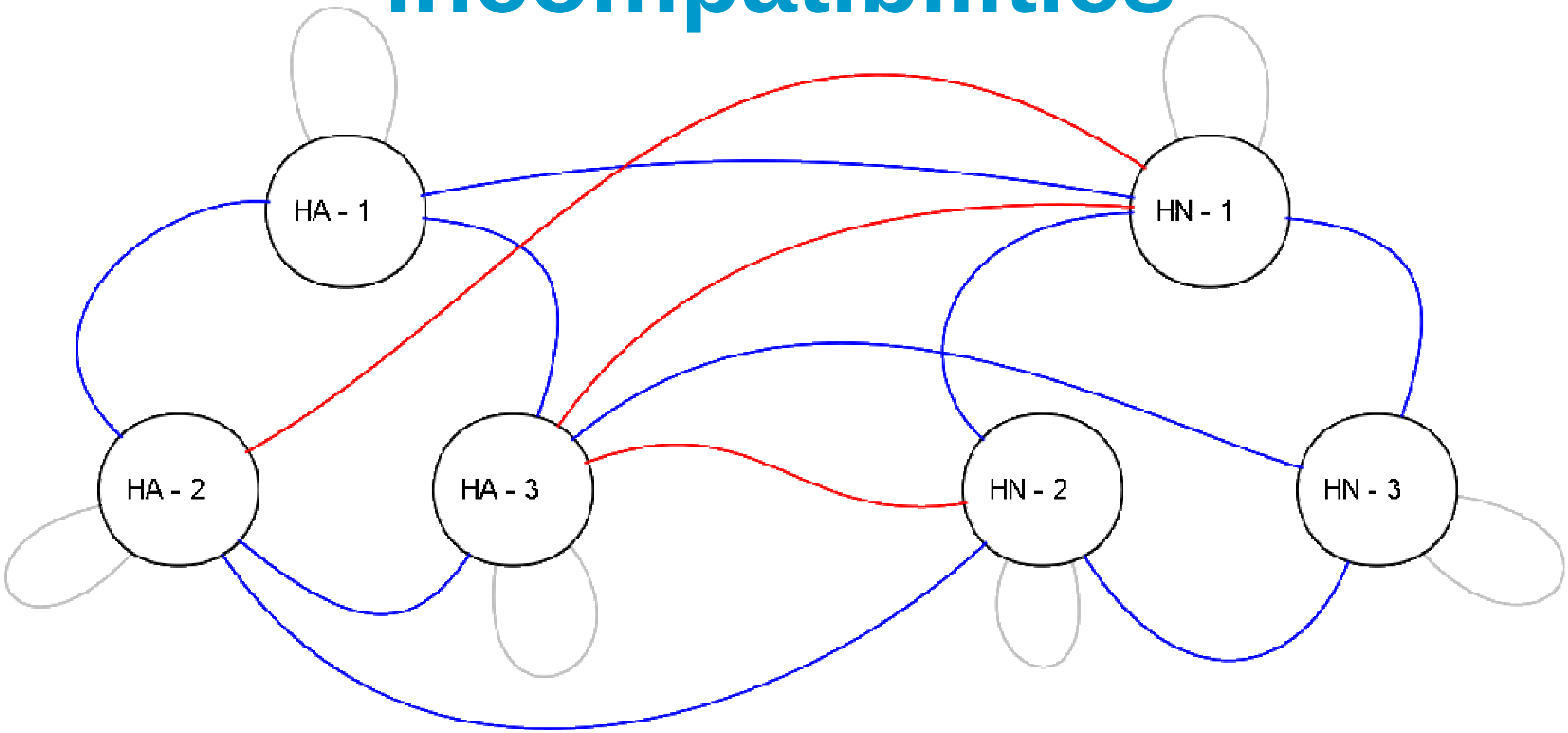
The goal





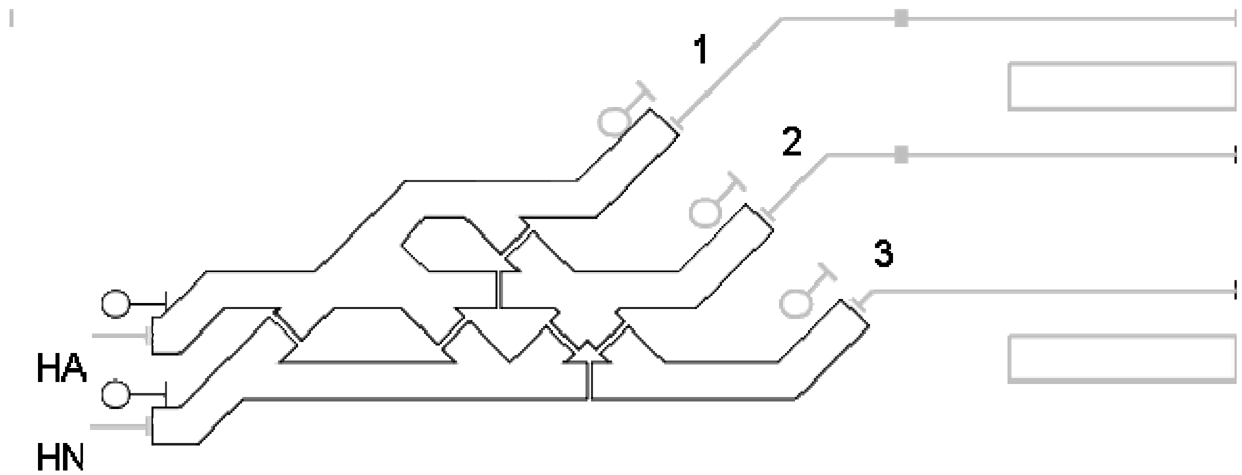
Consecutive block sections between main signals are grouped together in order to properly model route booking in <u>complex station interlocking areas</u>

Incompatibilities



An <u>incompatibility graph</u> is introduced to model properly the situation. A compact representation is obtained by analyzing the graph connectivity.

Virtual machines

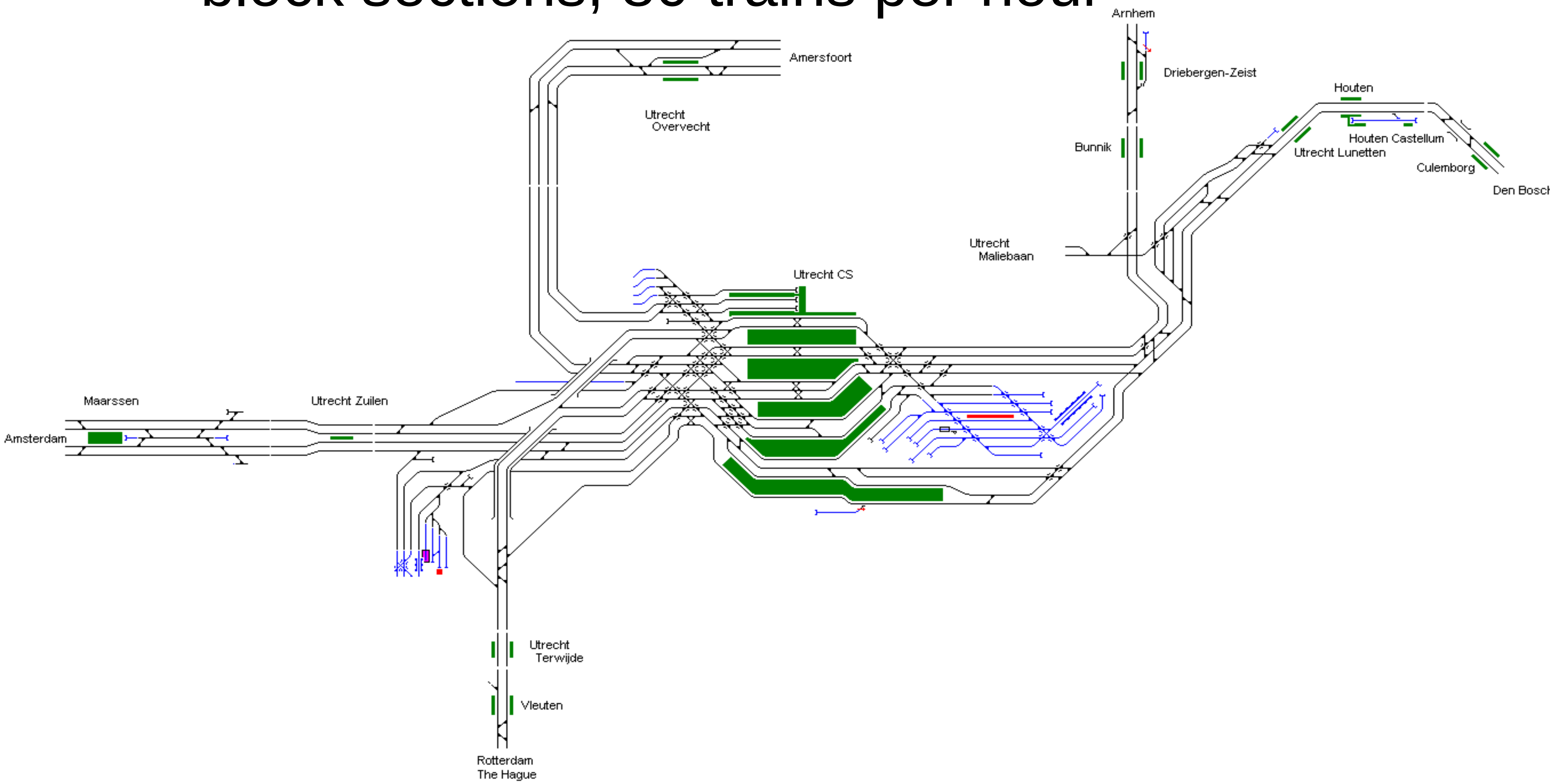


Virtual machines describe the graph connectivity

The lower bound is computed by <u>aggregating</u> the occupation time of trains on each virtual machine, rather than on each block section

Experimental assessment

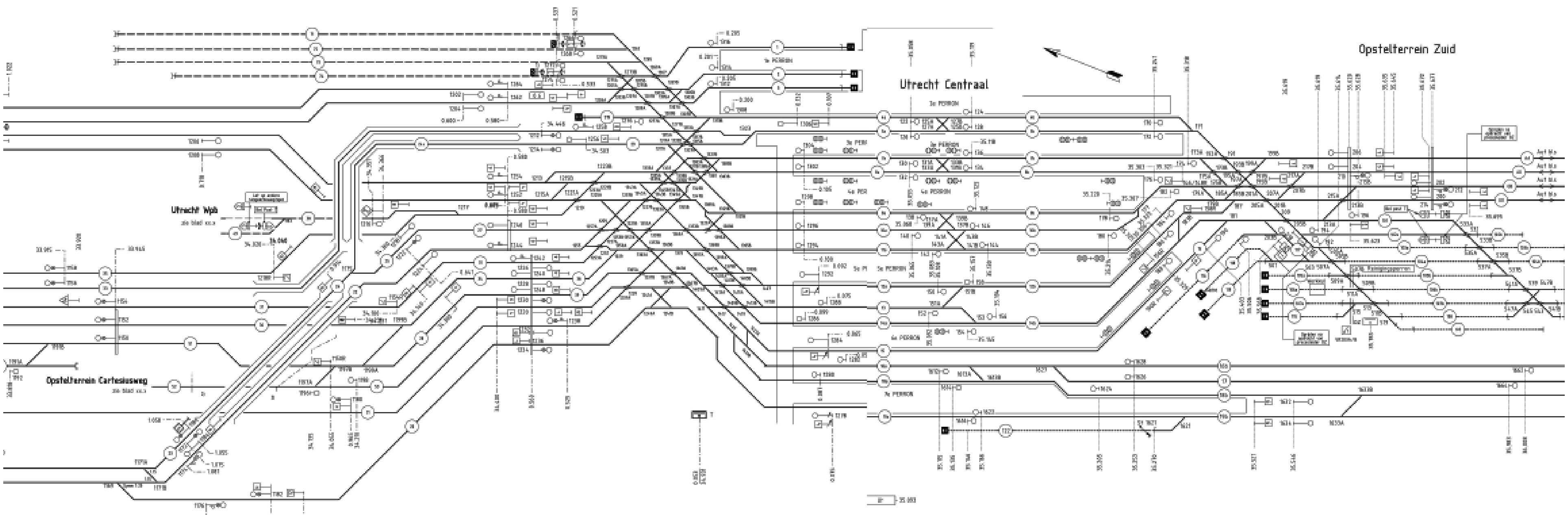
As a real-world test area we use the dispatching area around the Utrecht central station, > 600 block sections, 80 trains per hour



Benefits of aggregation

Without aggregation: ~12000 ordering decisions are to be taken for one hour of traffic prediction

With aggregation: ~200 blocks, ~5000 ordering decisions

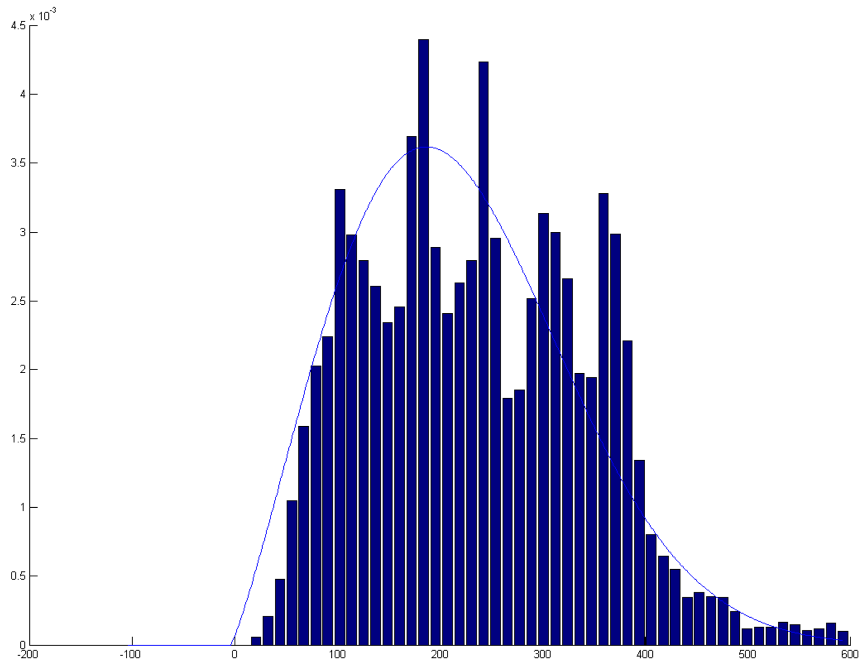


TUDelft

Delays

Entrance delays and dwell time extensions are modeled as Weibull distributions and fitted into the realization data collected at Utrecht CS in April 2008 (>33000 events)

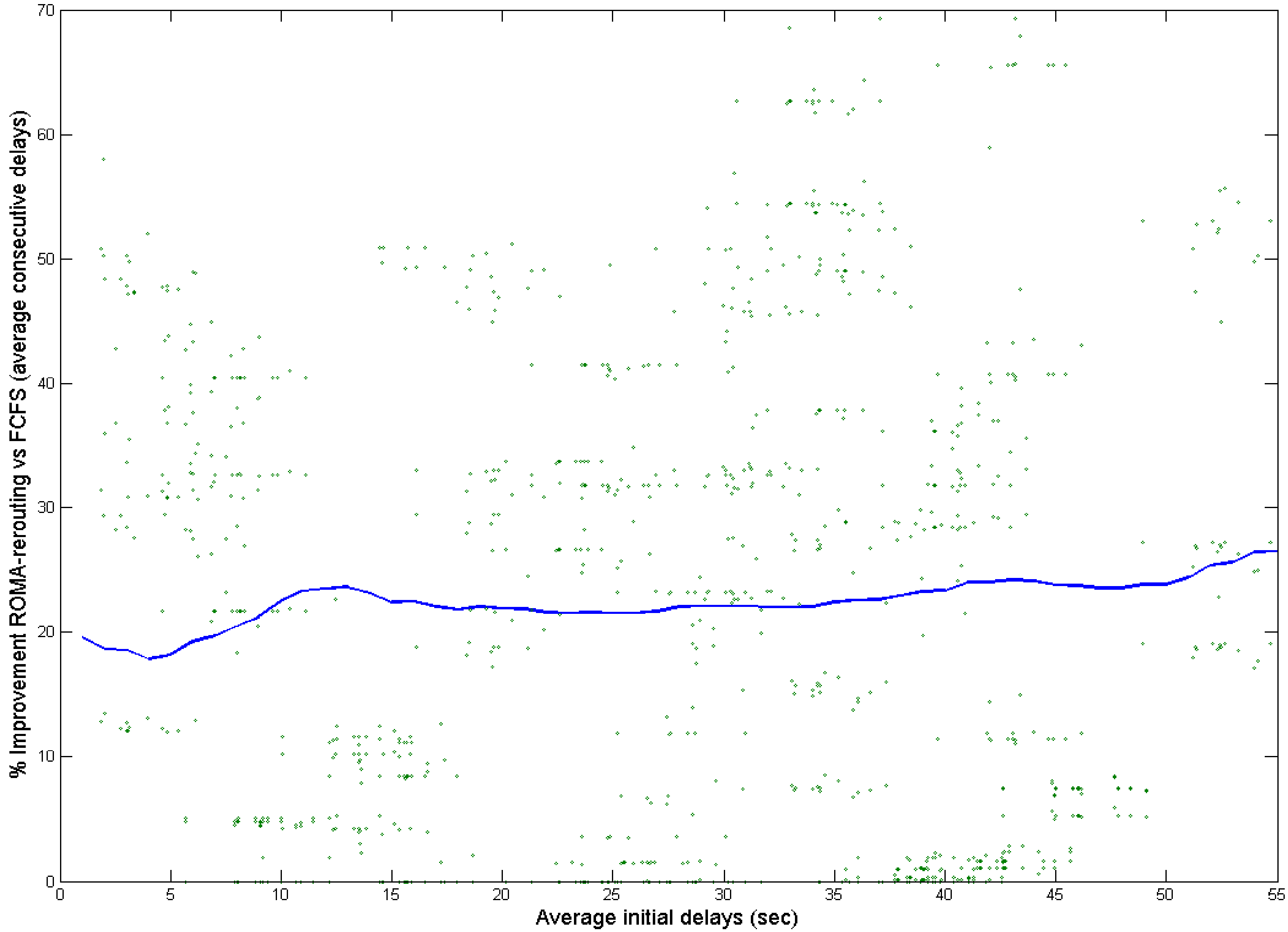
1800 perturbations instances represent the average disturbed traffic situation, including dwell time extensions, train delays and unavailable tracks



Main results

	average initial delay (s)	Compu-tation time (s)	max consecutive delay (s)	average consecutive delay (s)	average total delay (s)	punctuality within 0/3/5 minutes
TIMETABLE	29.3	5.8	622	50.1	94.5	39 / 83 / 88
ARI - ATR	29.3	5.7	446	28.2	74.3	42 / 84 / 93
FCFS	29.3	4.4	397	19	65.5	42 / 89 / 95
ROMA reordering	29.3	5.7	296	15.1	61.2	43 / 91 / 96
ROMA rerouting	29.3	52.3	299	14.6	60.8	43 / 92 / 96

Results - ROMA versus FCFS



Conclusions and future work

ROMA is a laboratory dispatching support tool able to forecast railway traffic and delay propagation using alternative graphs and blocking time theory

An aggregation procedure is used to manage the increased complexity of station interlocking areas

Promising dispatching solutions are found by fast and effective reordering and rerouting algorithms

Next research step will focus on coordination of rescheduling processes on large dispatching areas

Thanks

f.corman@tudelft.nl