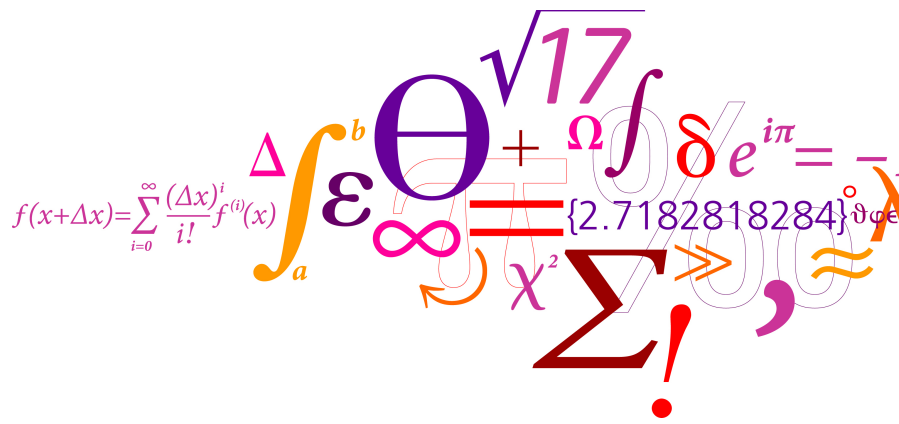
Simultaneously Recovering Rolling Stock Schedules and Depot Plans Under Disruption

Jørgen T. Haahr Richard M. Lusby David Pisinger Jesper Larsen

Department of Management Engineering
Technical University of Denmark

CASPT 2015,
July 23rd 2015, Rotterdam



Agenda

1. Introduction
2. Methodology
3. Results
4. Concluding Remarks

Agenda

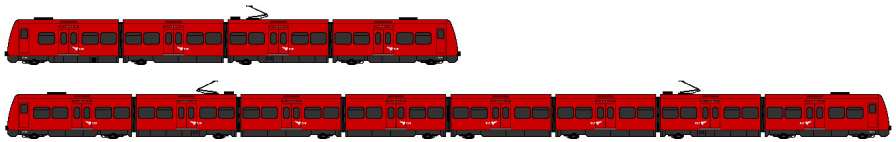
Introduction

- Part of a larger interdisciplinary project RobustRailS
- Improve the robustness and recoverability of the Danish rail system
- Focus of this project is on rolling stock disruption management
- Combine Rolling Stock Scheduling & Depot Planning
- Provide fast and reliable decision support systems
- Suburban Railway Network Operator in Copenhagen (DSB S-tog)



What is Rolling Stock Scheduling?

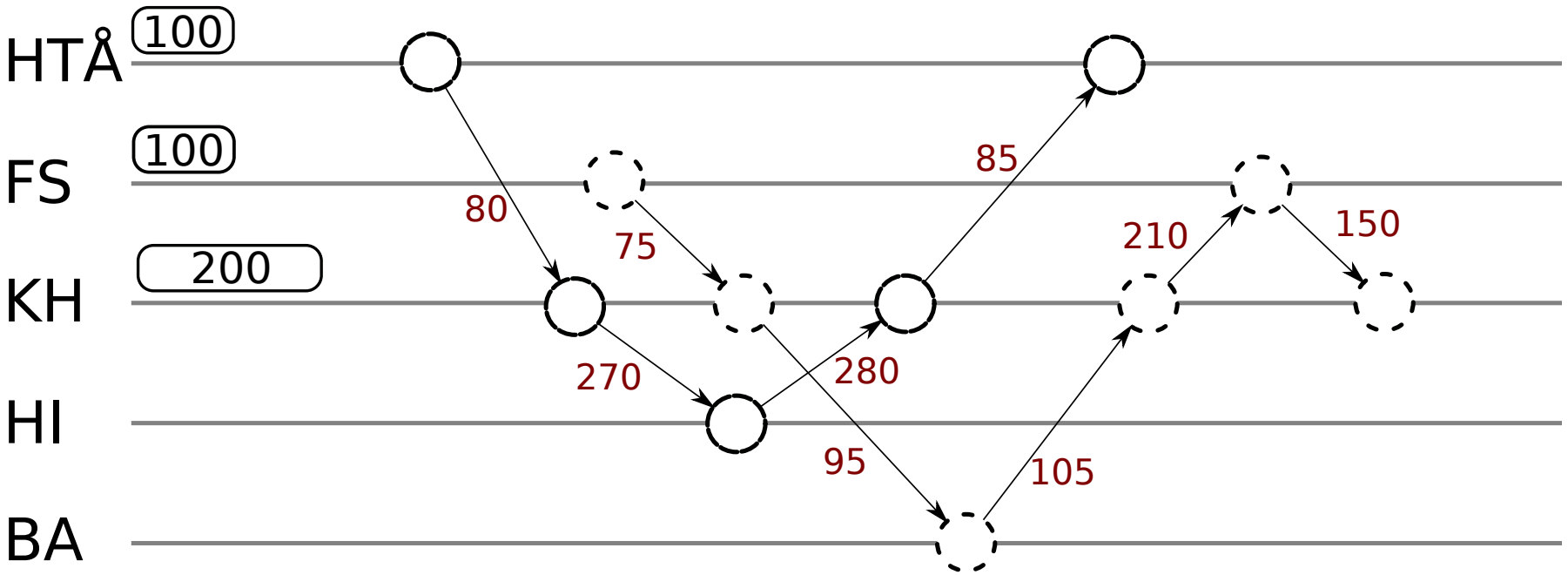
- Unit types



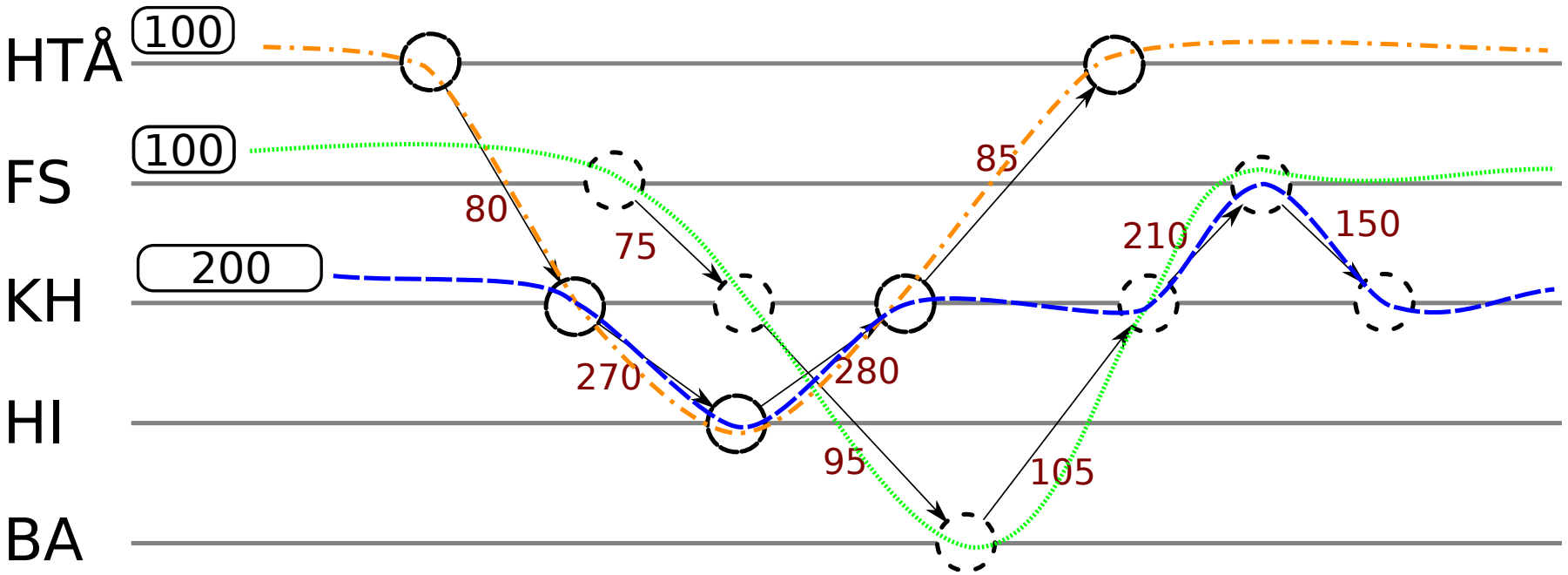
- Can be combined in different compositions

- Given a physical network and a set of timetabled trips, minimize the operational cost incurred in allocating enough units (in the form of compositions) to trips while ensuring various operational requirements are satisfied (unit flow, depot capacity, train length, etc.).

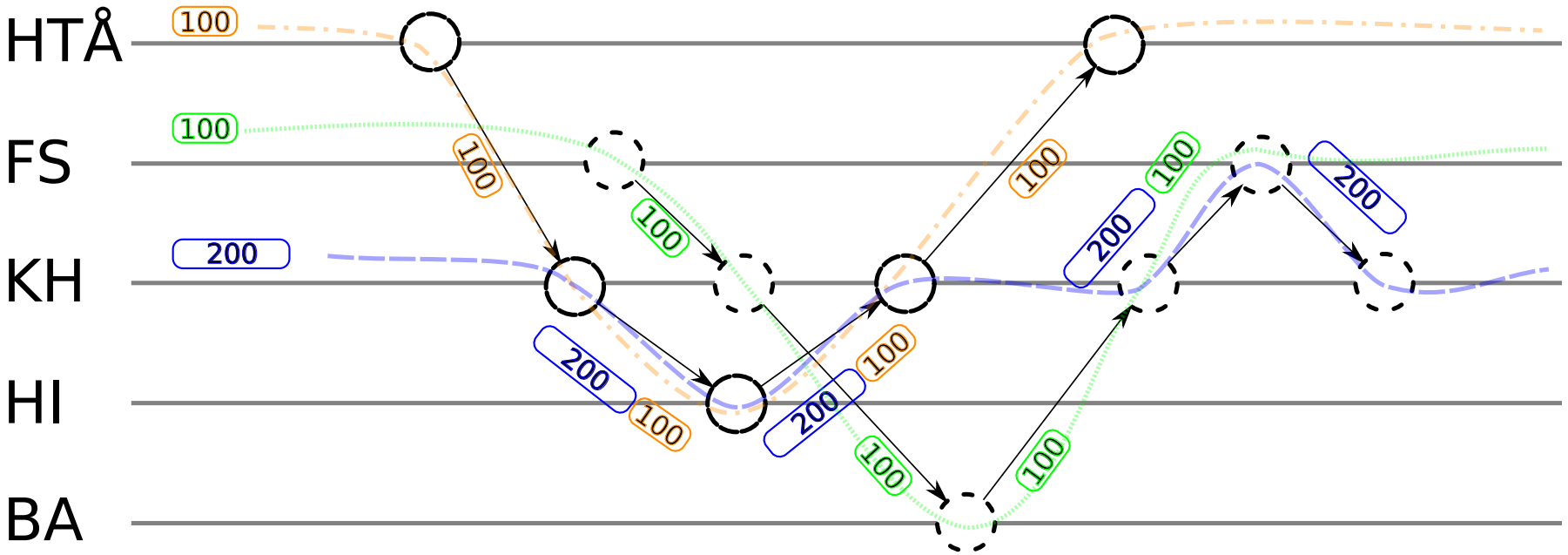
Rolling Stock Problem Example



Rolling Stock Problem Example



Rolling Stock Problem Example



What is Depot Planning?

- Also known as the Train Unit Shunting Problem (TUSP)
- A shunting movement is induced whenever a composition changes
- Each depot typically consists of multiple parallel tracks
- Ordering restrictions on unit retrieval
  - Last-In-First-Out (LIFO)
  - Unit must be accessible
- Park units in a conflict-free manner
- Objectives
  - Feasibility
  - Routing costs
  - Split compositions
- Closely related to rolling stock scheduling

What is Depot Planning?

- Also known as the Train Unit Shunting Problem (TUSP)
- A shunting movement is induced whenever a composition changes
- Each depot typically consists of multiple parallel tracks
- Ordering restrictions on unit retrieval
  - Last-In-First-Out (LIFO)
  - Unit must be accessible
- Park units in a conflict-free manner
- Objectives
  - Feasibility
  - Routing costs
  - Split compositions
- Closely related to rolling stock scheduling

Motivation

- Both problems have been studied independently
- Individual depots shouldn't be considered in isolation
  - Infeasibilities can propagate through the network
  - Exception if depot capacity is abundant
- Aim is to better coordinate rolling stock and depot planning
- Simultaneously solve both problems
- Focus on operational planning ($\rightarrow$ fast run times)
  - Equally applicable at earlier planning levels

Agenda

Proposed Framework

- Generate shunting feasible rolling stock schedules
- Embed depot(s) as separation routines in rolling stock routing framework
- **Idea:** Cut on (or repair) depot infeasible solutions when routing
- Large Branch-and-Price-and-Cut algorithm

Rolling Stock Rescheduling

- Based on the work of Haahr et al. (2014)
- Unit based path model
- Solved using delayed column generation
- Complete Branch-and-Price framework enforces integrality
- Assigns physical units (and compositions) to trips
  - Individual unit restrictions can be included when routing
- Includes aggregated depot constraint
- Efficient algorithm for rolling stock rescheduling
- Slightly different perspective to the approach of Neilsen et al. (2012)
  - Anonymous unit types assigned to trips

Depot Planning (TUSP)

- Not overly studied
- Effectively consists of two smaller problems
  - Train Unit Matching Problem
  - Track Assignment Problem/ Train Unit Parking (TUP)
  - Stems from anonymous compositions at routing phase
- Proposed approach circumvents the matching phase
  - Only the TUP is needed
- Compare different approaches for the (TUP)
  - Column Generation approach of Freling at al. (2005)
  - Extend this method with a constraint branching routine
  - Two Branch-and-Cut Algorithms
- Need to be able to prove/disprove feasibility quickly

Dealing with infeasibilities

- Conflicting shunting movements
- Multiple unit routes could yield the same composition assignment
- Shunting movements could actually be feasible for a different routing
- Heuristically identify potential unit swaps
  - Iterative procedure that attempts to insert unparked units
  - Unparked units are considered in order of "flexibility"
  - At most one swap per track considered at any iteration
- Shouldn't be seen as general remedy for infeasibility
- If still infeasible, combination of train compositions starting/ending at the infeasible depot is prohibited

Agenda

Test Instances - TUP

Instance	Events	Tracks	L_{max}	Horizon (s)	Types	Unit Lengths
data0	66	6	300.0	17113	2	[35,70]
data1	69	6	500.0	17785	2	[35,70]
data2	62	5	850.0	21103	2	[42,84]
data3	75	5	850.0	24837	3	[30,60,90]
data4	72	5	700.0	21602	2	[42,84]
data5	59	5	740.0	21602	3	[30,60,90]
data6	79	5	800.0	25202	2	[35,70]
data7	79	6	790.0	25202	3	[35,50,75]
data8	78	7	900.0	24897	1	[42]
data9	101	8	1000.0	24964	2	[42,84]
data10	109	8	400.0	28529	2	[42,84]

Method Comparison - TUP

		BAP Framework				BAC_1	BAC_2	
Instance	Z^*	*cols*	*n*	*l*	*t* (s)	*t* (s)	*t* (s)	%
data0	8	1030	37	18	1.33	3.21	*	12.50
data1	6	1317	41	20	1.74	0.36	*	16.67
data2	7	727	13	6	0.75	0.12	5.05	0.00
data3	8	1181	23	11	1.71	0.11	7.63	0.00
data4	11	834	25	12	1.05	0.14	*	9.09
data5	4	754	29	14	0.98	0.05	1.09	0.00
data6	11	841	17	8	1.18	1.99	*	36.36
data7	8	1740	41	20	2.18	1.76	*	12.50
data8	1	3236	99	49	4.86	0.05	0.36	0.00
data9	3	5125	87	43	17.33	0.42	*	66.67
data10	0	2413	147	73	6.14	0.23	1.29	0.00

Machine: Intel(R) Xeon(R) CPU X5550 @ 2.67GHz with 24GB ram

System: Ubuntu 14.04 , GCC 4.8.1, Coin-OR BCP 1.3.8, Cplex 12.5

Swapping Heuristic - Results

		BAP Framework			BAC_1		
Instance	Z^*	Z	*Swaps*	t (s)	Z	*Swaps*	t (s)
data0	8	0	12	7.383	0	19	3.39
data1	6	0	10	20.35	0	11	4.31
data2	7	0	11	5.50	0	16	1.33
data3	8	0	12	15.08	3	15	2.90
data4	11	1	21	126.10	1	16	5.12
data5	4	0	4	3.04	0	5	0.21
data6	11	0	22	18.748	1	18	15.58
data7	8	1	16	145.32	0	18	44.17
data8	1	0	1	8.70	0	1	0.14
data9	3	0	4	42.27	0	4	1.23

S-tog Network

S-tog - Test Cases

Name	Stops	Trips	Trips*	Weekday	Lines
Fri	28 719	4 558	886	Friday	A,B,Bx,C,E,F&H
Sat	20 474	1 916	590	Saturday	A,B,C&F
Sun	19 919	1 871	574	Sunday	A,B,C&F
Mon	28 017	4 468	868	Monday	A,B,Bx,C,E,F&H

Up to 13 active depots on any given day

S-tog - Results

Instance	Cost	Couplings	t (s)	RS	Solutions	Depot	
						Feasible	Infeasible
Fri	600	48	128	82%	1	1	0
	700	41	154	78%	1	1	0
	800	34	80	83%	1	1	0
	900	30	94	77%	1	1	0
	1 000	27	79	76%	4	4	0
Sat	600	29	23	71%	1	1	0
	700	25	18	68%	1	1	0
	800	23	23	67%	1	1	0
	900	21	13	64%	1	1	0
	1 000	17	11	61%	7	7	0

Agenda

Conclusions

- Proposed an approach to simultaneously consider rolling stock and depot planning
- Includes a heuristic swapping routine to repair infeasible depots
- Further testing is needed
- Cutting routine is unused at this stage
- Column generation does not seem to be a viable approach (TUP)

Thanks!