Erasmus Universiteit Rotterdam

Faculteit der Economische Wetenschappen

Robust Transportation Systems

Rommert Dekker,

Judith Mulder, Leo Kroon, Michiel Vromans

Erasmus School of Economics, Rotterdam, The Netherlands

Contents

- Some theory
- Robust train networks
- Robust Shipping networks
- Vehicle routing ideas

Transportation Systems

- Ship
- Truck
- Train
- Plane

both passengers and cargo

Robustness of transportation. systems

Being able to provide service under all kind of disturbances, like higher demand, congestion, weather, breakdowns, etc.

A big problem with disturbances is that they can have knock-on effects all over the region / country / world.

Robustness

Two important concepts to create robustness

- Time Buffers (Supplement / slack) and Over capacity

- Recovery actions (flexibility)
  drive /sail faster, take shortcuts, bypass stations etc.

Two types of disturbances:

- Small ones – taken care off by buffers
- Big ones – need recovery actions

Main Problem

Time Buffers – planning in more time then what is technically needed under normal circumstances.

Time buffers and Over capacity are not productive and sometimes even counter productive (idle trains, planes take up platform / gate capacity)!

Every system applies time Buffers, yet the question is how much is needed?

Example: NS Dutch Railways applies a 7% time buffer in her timetable. This number was set some time ago and no scientific justification existed.

Transportation Models for Robustness

Typical two stages

First stage: make decisions for medium term, e.g. timetable, equipment and manpower allocation (railways, airlines, shipping lines, express companies)

Second stage: react to the daily disturbances
- longer travel times
- longer loading / unloading times
- more / less demand

Mathematical Approaches to Robustness

Stochastic Programming

Robust optimization

Stochastic dynamic programming /
Markov Decision Programming

Simulation
+ Optimization

Stochastic Programming

The classical two-stage linear stochastic programming problems can be formulated as

$$\begin{aligned} \min_{x\in\mathbb{R}^n} \quad & g(x) = c^T x + E[Q(x,\xi)] \\ \text{subject to} \quad & Ax = b \\ & x \geq 0 \end{aligned}$$

where $Q(x,\xi)$ is the optimal value of the second-stage problem

$$\begin{aligned} \min_{y\in\mathbb{R}^m} \quad & q(\xi)^T y \\ \text{subject to} \quad & T(\xi)x + W(\xi)y = h(\xi) \\ & y \geq 0 \end{aligned}$$

ξ - indicates randomness - disturbances

Stochastic Programming

Assumes probability distributions given.

Probability distributions can be approximated using samples for simulation. Typically few different values make the solution more robust and convergence can be proven.

Second stage outcome should preferably be linear in first stage variables, for each nonlinearity an integer variable needs to be defined.

Robust Optimization

Counter part – not assuming probability distributions, but ranges of variables.

Hence is independent of choice of distributions.

Same issues as with nonlinearity as SP.

Stochastic Dynamic Programming

- (Discrete) State space E = {1,2,...}
- Actions a ϵ A(i): to be taken independent per state
- Transition probabilities, $p_{ij}(a)$, i,j ϵ E
- Expected Costs $c_i(a)$, i i,j ϵ E ,
- Solution methods: policy improvement, successive approximation, linear programming formulation

Stochastic Dynamic Programming

- Solution method complexity; N^3, where N – number of states. Much more easy to model nonlinearity.

- State should express all information needed to make decisions, so transitions should be independent of all other aspects.

- Yet state space: often multi-dimensional e.g. inventory of each product
  Main problem: state space is quickly too large

Approximate Dynamic Programming

Main element in DP: the value function v_i, which expresses the total (discounted) costs when starting in state i.

Idea: do not determine v_i for all states i, but only for some states, assume a function for it and approximate that, or determine it by Neural networks.

Use v_i in the optimality equation to determine optimal action a from

$$v_i = \min_{a \in A(i)} c_i(a) + \sum_j P_{ij}(a) v_j$$

Contents

- Theory
- *Robust train networks*
- Robust Shipping networks
- Vehicle routing ideas

Railway punctuality - introduction

- Punctuality – the % of trains arriving within k minutes after published
  The Netherlands: was k = 3; now and in many other countries k = 5
  (3 min) punctuality 2005: 85%, target 86%, 89% in 2006,
  punctuality (5 min) 2010: 92.5%: ; 2013: 93.6%

- Importance of punctuality:
  official company target in contract Dutch Railways – Government, determines allowed price increase + bonus board.

- Punctuality does not fully match customer perception, yet other measures (weighted punctuality) are much more difficult to calculate: presently under construction

Delays and causes

- Primary delays – caused by outside causes
  some studies: exponentially distributed, but parameter varies.
  Correlation? Rainy days and peak hours do have effect.

- Main causes
  - failure of rolling stock
  - failure of railway infrastructure (switches, safety system, computers)
  - travellers and accidents

- Secondary delays – caused by delays of other trains:
  Netherlands is sensitive to that because of long lines.

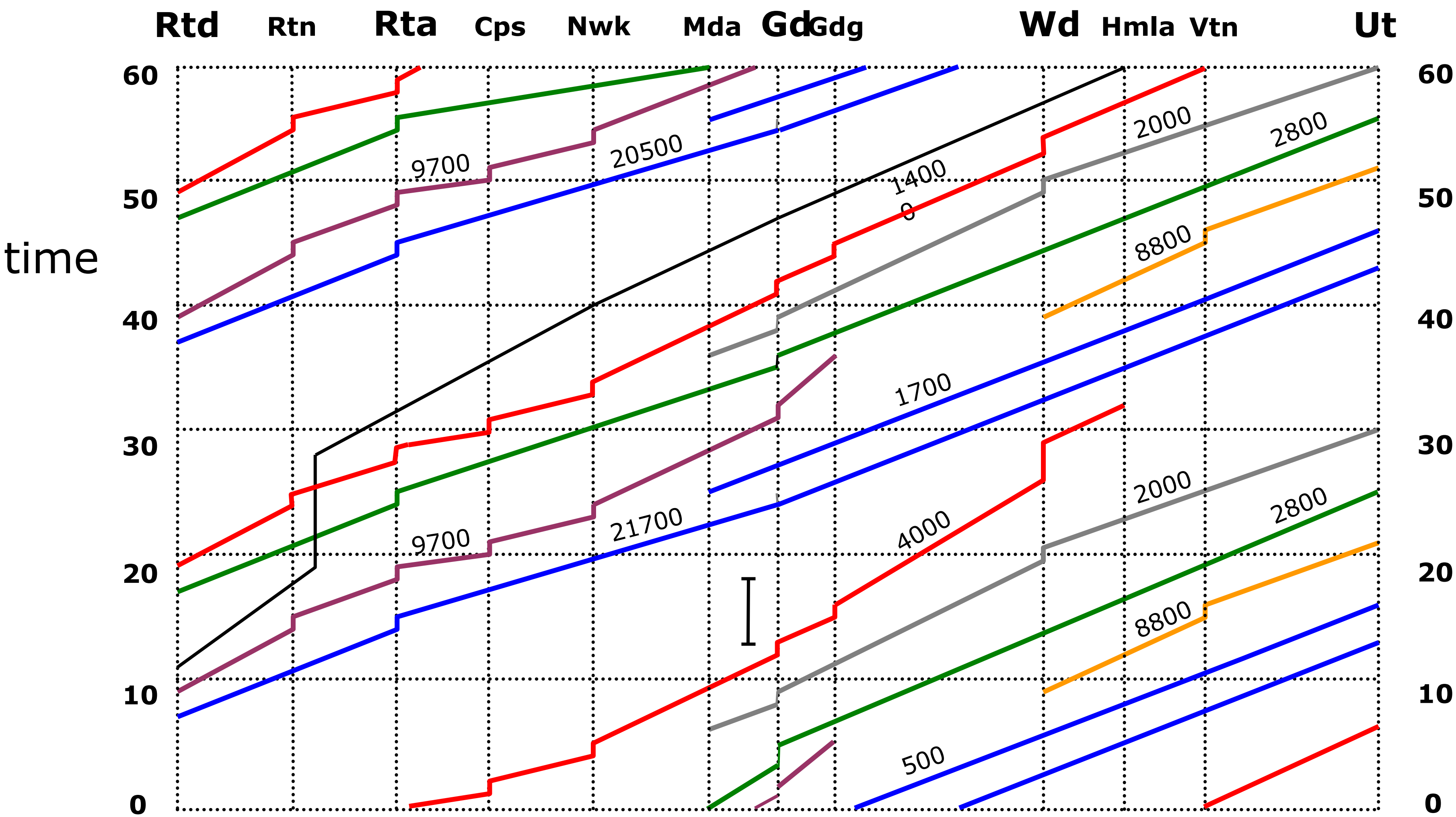
- Problem: only total delays are measured: traffic control has not enough time to register primary delays in much detail


Spoorkaart lange afstand
Rail 1435
©1998
Groningen
Leeuwarden
Assen
Heerenveen
Steenwyk
Emmen
Den Helder
Hoorn
Enkhuizen
Alkmaar
Zwolle
Lelystad
Amsterdam
Almere
Almelo
Hengelo
Hannover
Berlin
Enschede
Deventer
Apeldoorn
Haarlem
zutphen
Leiden
Schiphol
Amersfoort
Den Haag
Gouda
Utrecht
Ede-W
Arnhem
Nijmegen
Emmerich
Düsseldorf
Köln
Oss
Rotterdam
Dordrecht
's-Hertogenbosch
Venray
Roosendaal
Breda
Tilburg
Eindhoven
Goes
Bergen op zoom
Venlo
Helmond
Middelburg
Vlissingen
Weert
Köln
Roermond
Gent
Brugge
Oostende
Mechelen
Brussel
Paris Nord
Sittard
Heerlen
Maastricht
Legenda
IC / IR lijn
AR-lijn (stoptrein)
Station
Duivendrecht

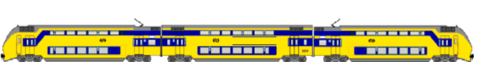
Railway Time table principles

- Basic hourly pattern (with some extra trains)
- Time Symmetry at some stations
- Timetable published in integer minutes

Rotterdam (Rtd) – Utrecht (Ut): the hourly pattern

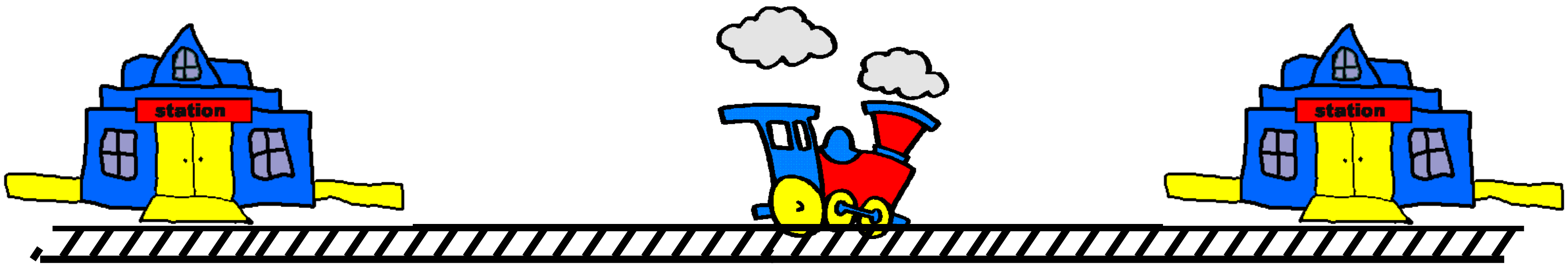


Vertical line: trains wait at station



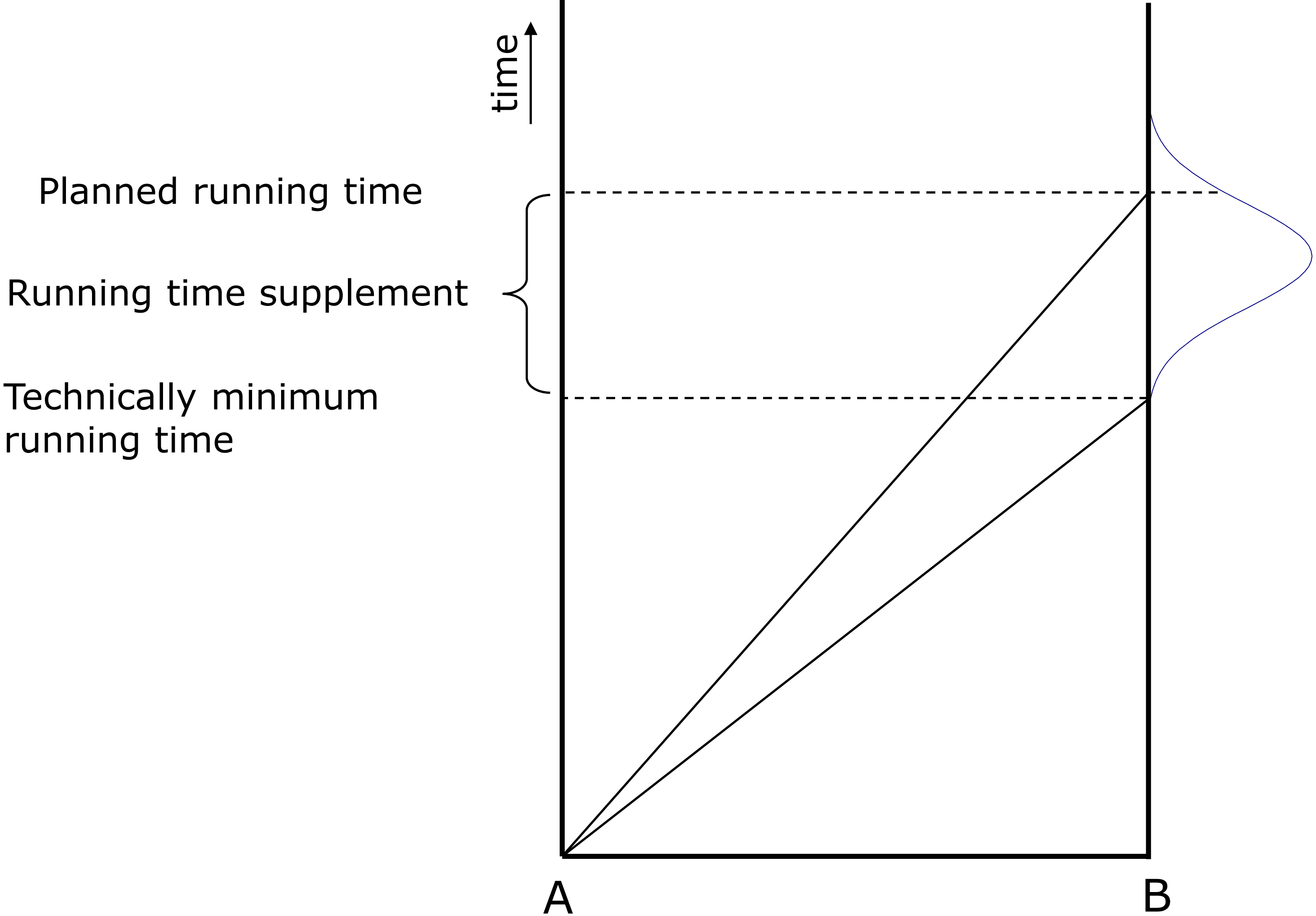
Processes and Process times

- Processes: running, halting, connecting, headways, etc.
- Planned process times are deterministic values
- Actual process times are subject to stochastic disturbances



- Planned process time =

Technically minimum process time + Supplement



Dutch practice: (2005) : Running time supplement = 7% of technically minimum running time

Origin 7%: unclear! Presently (2011): 5% is used after better calculation methods and incorporating station halting time

Relevant questions:

- How much supplement is to be used?
- How are the supplements to be allocated?

Tools:

- Timetable generation
  - ✓ Periodic Event Scheduling Problem (Serafini & Ukovich (1989)
- Timetable evaluation
  - ✓ Simulation tools (SIMONE tool – Incontrol)
  - ✓ Max-Plus Algebra –identifies which circuit has lowest slack (Heidergott et al)
  - ✓Lagrangian based methods to improve robustness (Fischetti)
- Timetable generation and evaluation combined
  - ✓ Stochastic Optimization (Kroon, Vromans, Dekker et al.)

Robust railway timetable research

Simulation:

Bergmark 1996, Middelkoop & Bouwman (2000) – DONS, etc

- do evaluation only;
- sophisticated models have direct link with timetable generation programs
- modern packages can simulate whole network, but are dedicated to operators
- no inclusion of crew rotation, platform allocation and limited traffic control

-- no timetable generation

Robust railway timetable research

Analytical models

- Max-plus algebra: Goverde (1998), de Kort (1999)
- Heterogenity measures: Carey (1999), Vromans et al (2005)
- Queueing models at intersections: Huisman & Boucherie (2001)
- Stochastic Models without knock-on effects Catrysse et al. (2011)

Economics – public stated preference ratios have been determined of running time vs waiting time (1 : 2.37 (Rietveld et al. 2001))

Robust railway timetable research

Two cases:

- Small disruptions: basic train order is preserved, trains delay other trains, but no change of order

- Large disruptions: trains dispatchers implement changes to plan -> rescheduling research
make quickly new plans given existing situation and restrictions

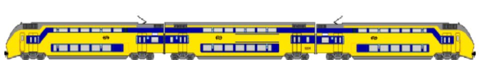
It is very difficult to make timetables while taking these rescheduling into account.

Possible modelling approaches

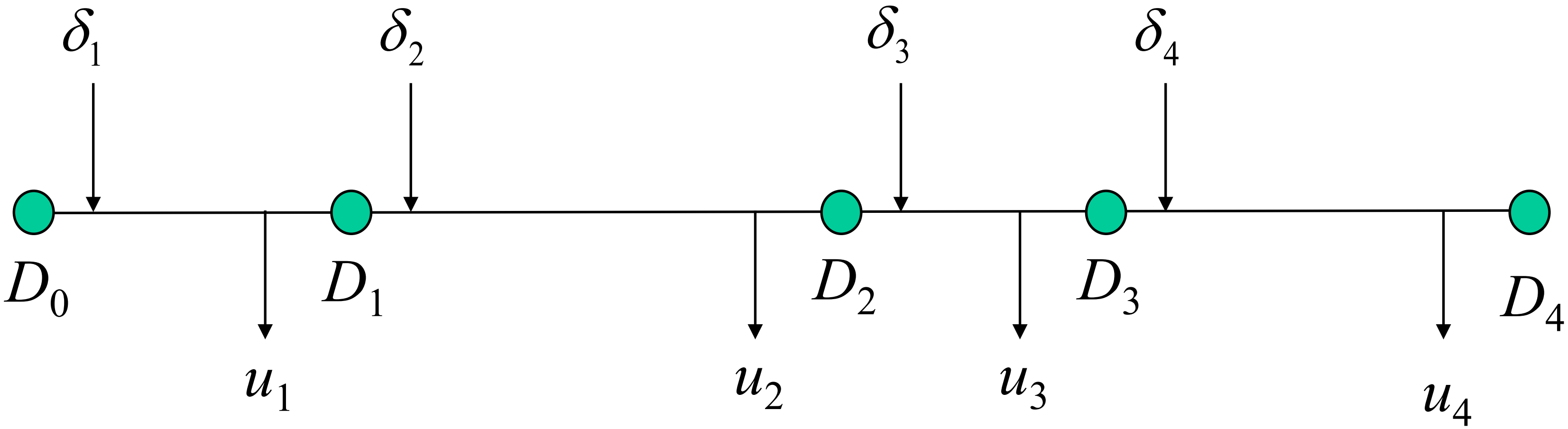
- Simulation – little optimisation possibilities
- Markov decision models – problem is that time supplements have to be fixed before hand; optimal policy in MDP is state dependent, however, state = amount of delay. Action – use x minutes of supplement, but the supplement determination has to happen outside the MDP.

- Stochastic programming: first stage (fix supplement) and second (recourse stage – recover delays).. Problem is that delays propagate and have a long lasting effect.

- Approximative methods ? How? The method should be used in conjunction with IP models for timetabling.
- Light Robust optimization (Fischetti 2007) Penalize deviations in a Lagrangian way to ensure robustness.



Single train delay model



D_t = delay after trip t

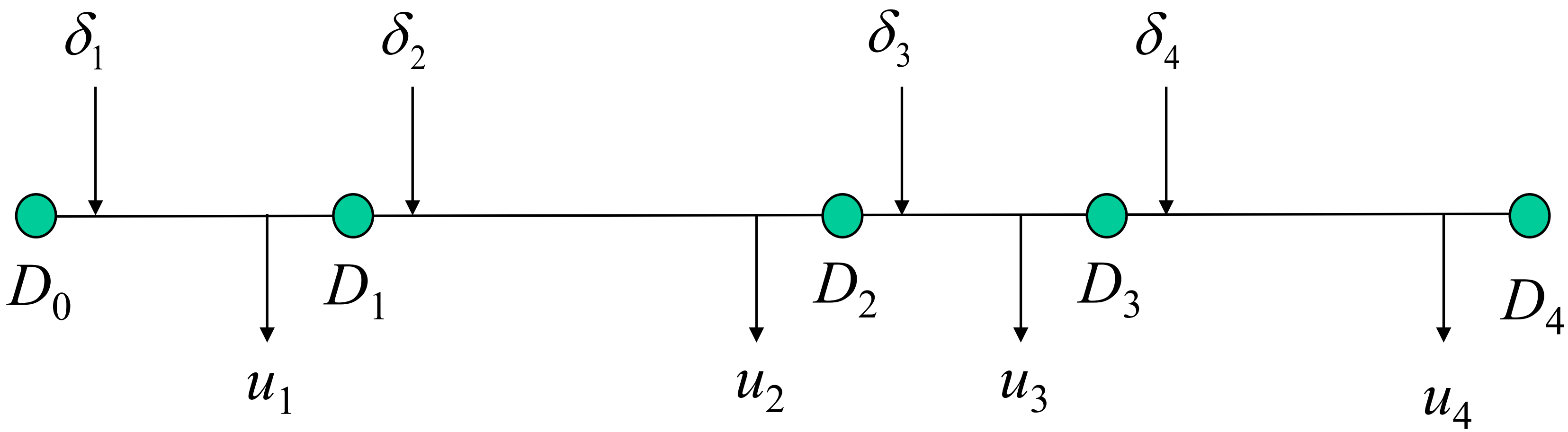
s_t = available supplement on trip t

δ_t = disturbance on trip t

u_t = used supplement on trip t

$$D_t = max\ (0, D_{t-1} + \delta_t - s_t)$$

Single train single line delay model



$D_t = D_{t-1} + \delta_t - u_t$ for $t = 1, \ldots, T$

$u_t \leq s_t$ for $t = 1, \ldots, T$

NB This replaces

$D_t = max\ (0, D_{t-1} + \delta_t - s_t)$

D_t as small as possible

Random variables and time supplements

Main mathematical operation:

- $D_t = max\ (0, D_{t-1} + \delta_t - s_t)$
- D_t – unknown random variable, δ_t – the primary delay random variable, s_t the given supplement

- Problem is that no known distribution families of random variables are preserved under this operation

$$D_1 = D_0 + \delta_1 - u_1$$

$$D_2 = D_1 + \delta_2 - u_2$$

$$D_3 = D_2 + \delta_3 - u_3$$

$$D_4 = D_3 + \delta_4 - u_4$$

$$D_4 = D_0 + \sum_{t=1}^{4} \delta_t - \sum_{t=1}^{4} u_t$$

$$4 \times \overline{D} = \sum_{t=1}^{4} D_t = 4 \times D_0 + \sum_{t=1}^{4} (5-t) \times \delta_t - \sum_{t=1}^{4} (5-t) \times u_t$$

Minimize $\overline{D}$

subject to

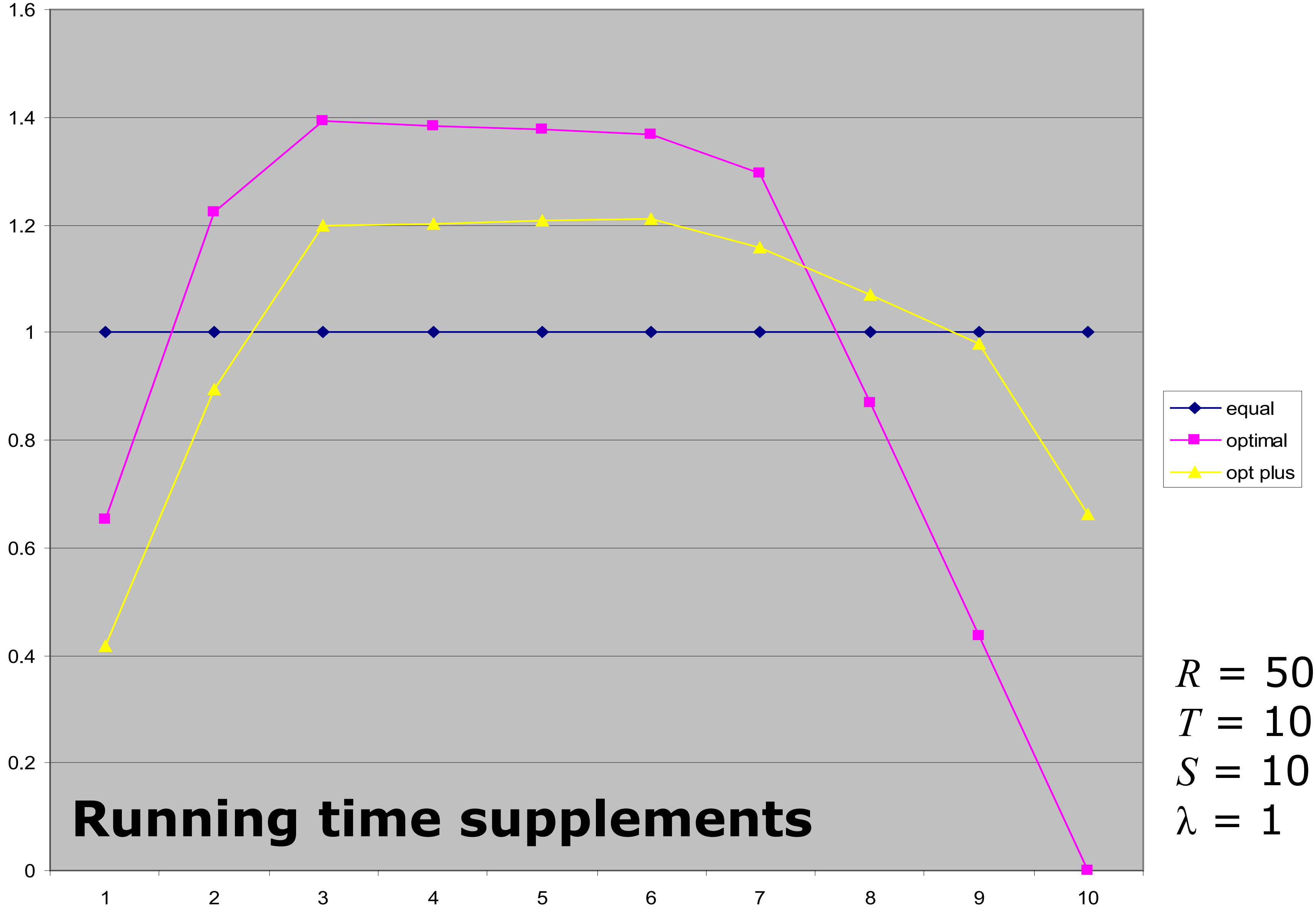
$$D_{t,r} = D_{t-1,r} + \delta_{t,r} - u_{t,r}$$ for $t = 1,\ldots,T$ and $r = 1,\ldots,R$

$$u_{t,r} \leq s_t$$ for $t = 1,\ldots,T$ and $r = 1,\ldots,R$

$$\sum_{t=1}^{T} s_t = S$$

$$\overline{D} = \sum_{t=1}^{T} \sum_{r=1}^{R} D_{t,r} \Big/ (T \times R)$$

TxR realisations $\delta_{t,r}$ are drawn using stratified sampling of primary delay distributions

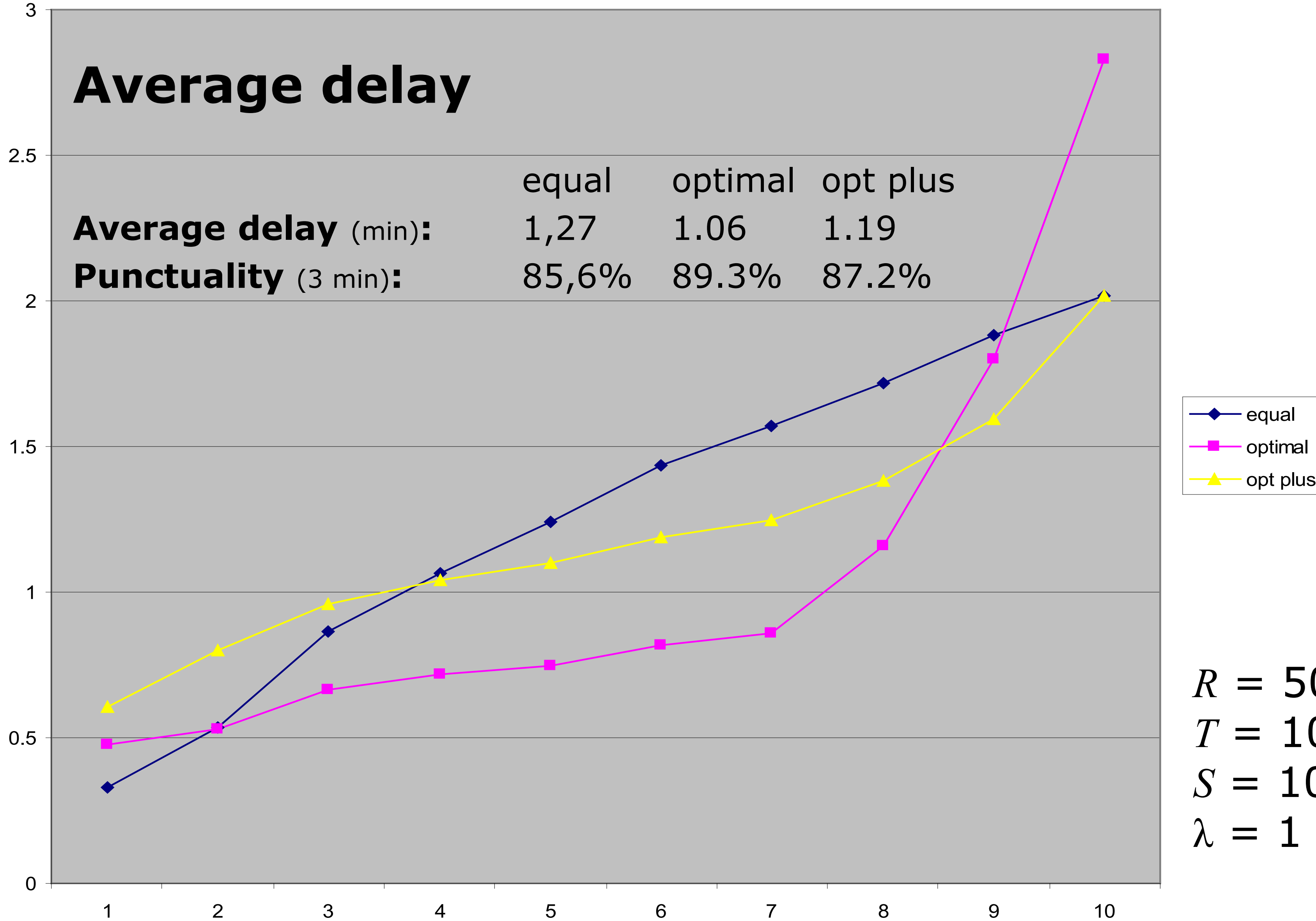


$R = 500$
$T = 10$
$S = 10$
$\lambda = 1$

Delays ~ exp (1) distributed

Average delay

	equal	optimal	opt plus
Average delay (min):	1,27	1.06	1.19
Punctuality (3 min):	85,6%	89.3%	87.2%



$R = 500$
$T = 10$
$S = 10$
$\lambda = 1$

Remarks on the single train model

- Results are empirical only, yet observed in all cases and explainable

- Results also correspond to optimal appointment schemes in hospitals

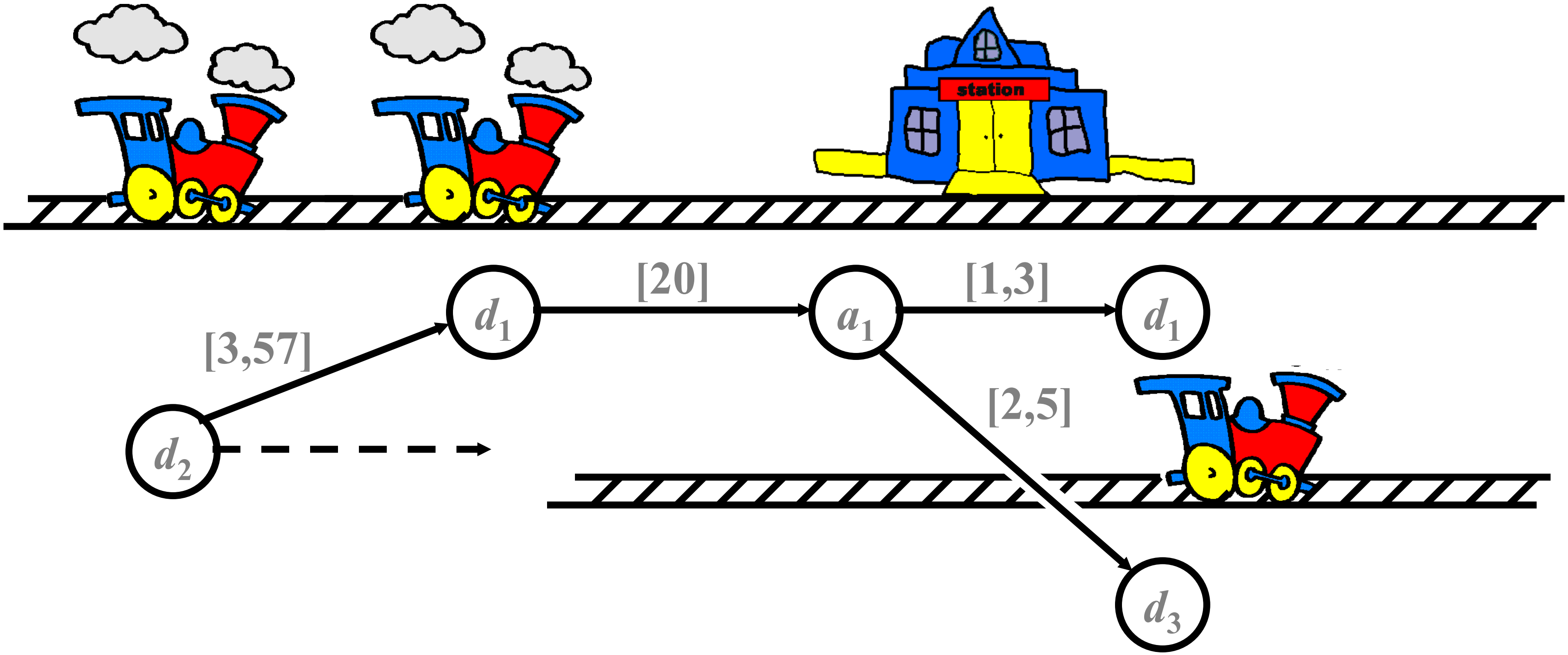
- Open problem: under which conditions can we show that such a U pattern is optimal?

Extensions of the single train model

- Several trains
- Headway times
- Single track sections
- Passenger travel times
- Passenger connections
- Rolling stock circulation

- Fixed or variable cyclic order of events
- In the operations, cyclic order is “the same” as in the plan
- Interaction between successive cycles

Extension: complete timetable



Planning part of the model (thanks to Schrijver)

$a_t = d_t + r_t + s_t$ for $t = 1,\ldots,T$ running time

$d_{n(t)} \geq a_t + \Delta_t$ for $t = 1,\ldots,T$ with $n_t \neq 0$ Min. dwell time

$d_{t'} \geq d_t + h_{t,t'}$ for consecutive $t, t' = 1,\ldots,T$ headway time

$a_{t'} \geq a_t + h_{t,t'}$ for consecutive $t, t' = 1,\ldots,T$ headway time

$d_{n(t)} \geq a_t + A_t - K_{t,t'} \times 60$ if t last trip of a train turn-around time

$\sum_{t=1}^{T} s_t = S$ total supplement

a_t – arrival time, d_t - departure time, t- train index, etcetera

Realization part of the model (1)

$\overline{a}_{t,r} \geq \overline{d}_{t,r} + r_t + \delta_{t,r}$ for $t = 1,\ldots,T$ and $r = 1,\ldots,R$

$\overline{d}_{n(t),r} \geq \overline{a}_{t,r} + \Delta_t$ for $t = 1,\ldots,T$ with $n(t)$ and r = 1,…,R

$\overline{d}_{t',r} \geq \overline{d}_{t,r} + h_{t,t'}$ for consecutive $t, t' = 1,\ldots,T$ and r = 1,…,R

$\overline{a}_{t',r} \geq \overline{a}_{t,r} + h_{t,t'}$ for consecutive $t, t' = 1,\ldots,T$ and r = 1,…,R

$\overline{d}_{n(t),r} \geq \overline{a}_{t,r-K} + A_t$ if t is the last trip of a train, and r = K+1,…,R

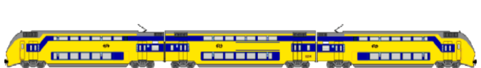
Realization part of the model (2)

$\overline{d}_{t,r} \geq d_t + r \times 60$ for $t = 1,\ldots,T$ and $r=1,...,R$

$D_{t,r} \geq \overline{a}_{t,r} - (a_t + r \times 60)$ for $t = 1,\ldots,T$ and $r=1,...,R$

$\overline{D} = \sum_{t=1}^{T} \sum_{r=1}^{R} D_{t,r} \Big/ (T \times R)$ for $t = 1,\ldots,T$ and $r=1,...,R$

All variables are non-negative; the planned event times are integer

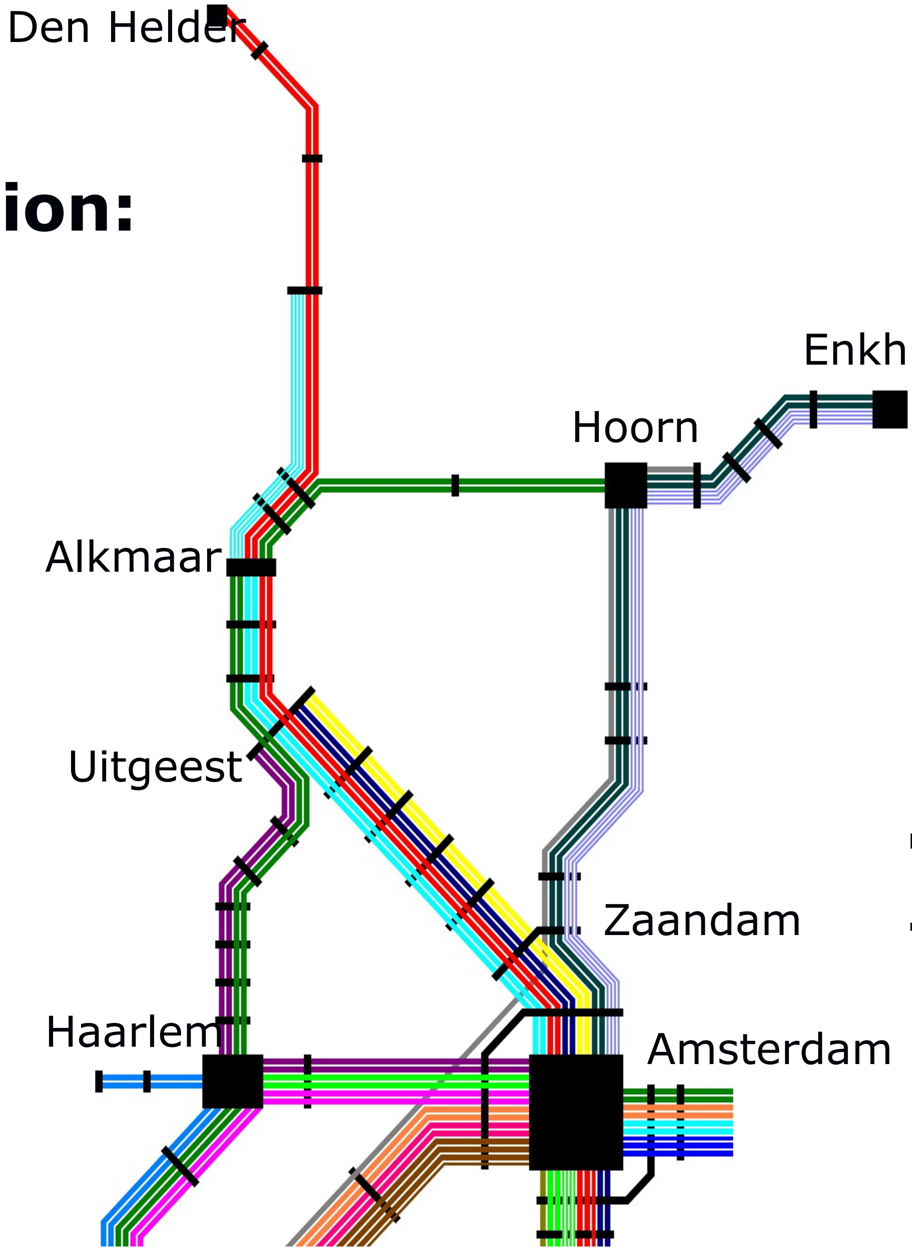


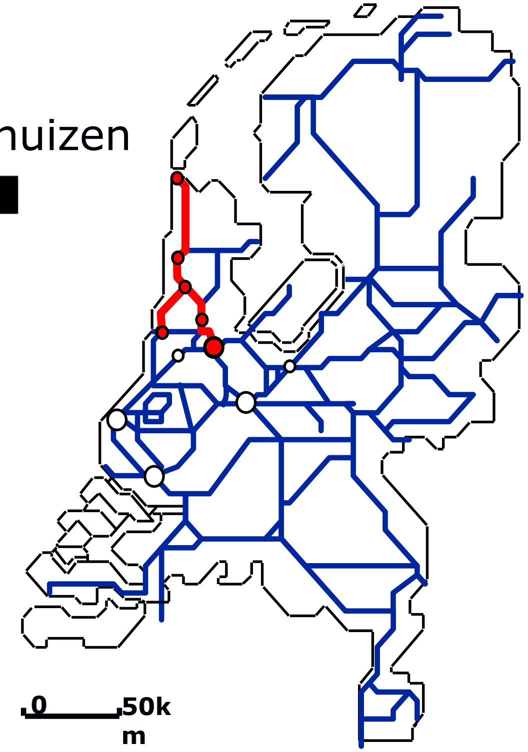
Increasing the number of realizations:

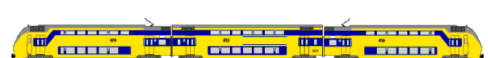
Using a theorem from Kleywegt et al. (2001), we can show that the solution of the models with a limited number of scenarios converges to the optimal one in the model with probabilistic deviations

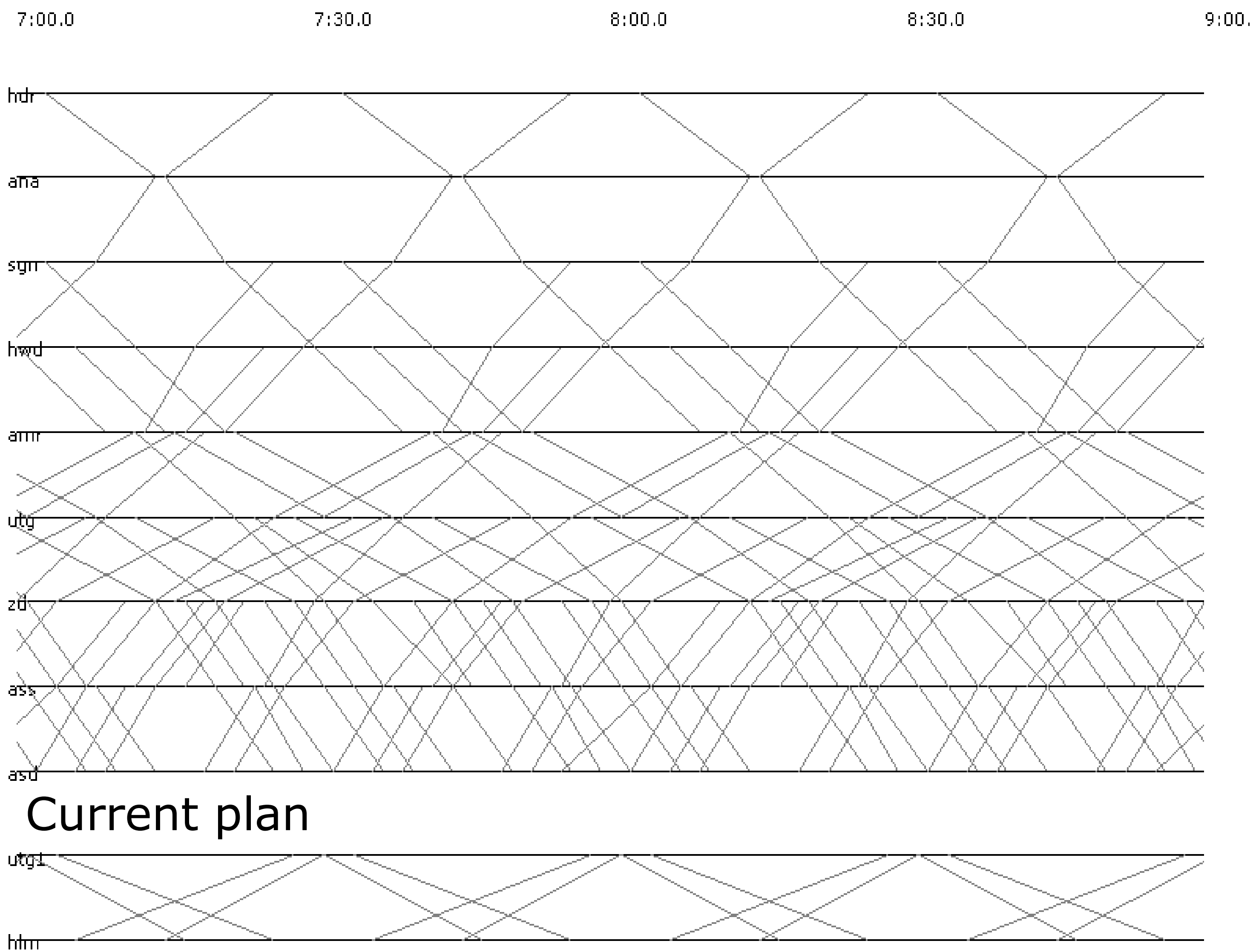
Application:

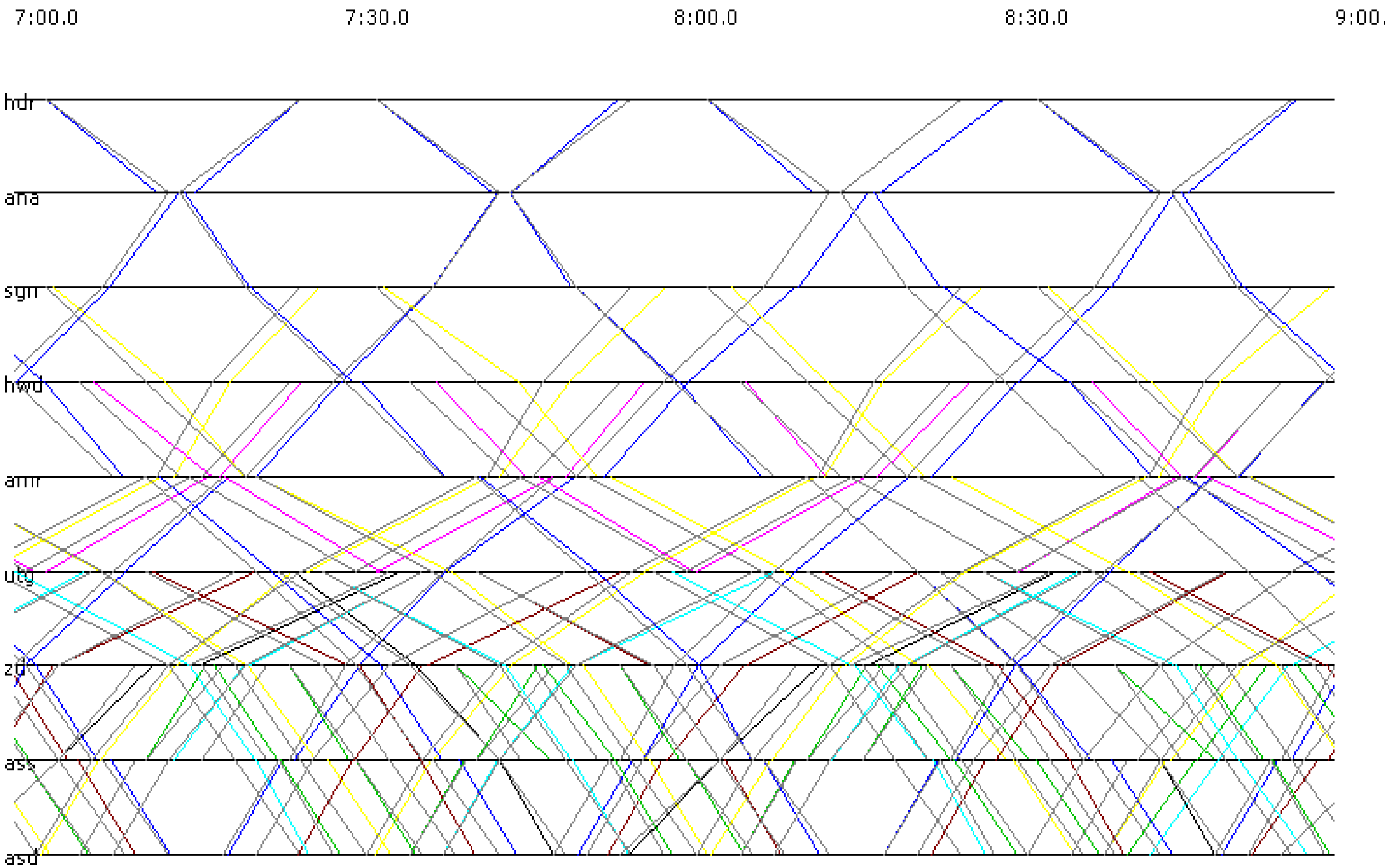
Zaanlijn



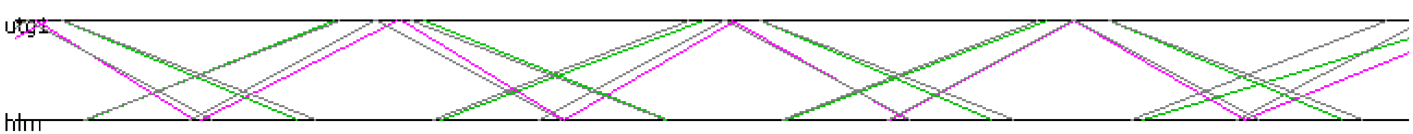


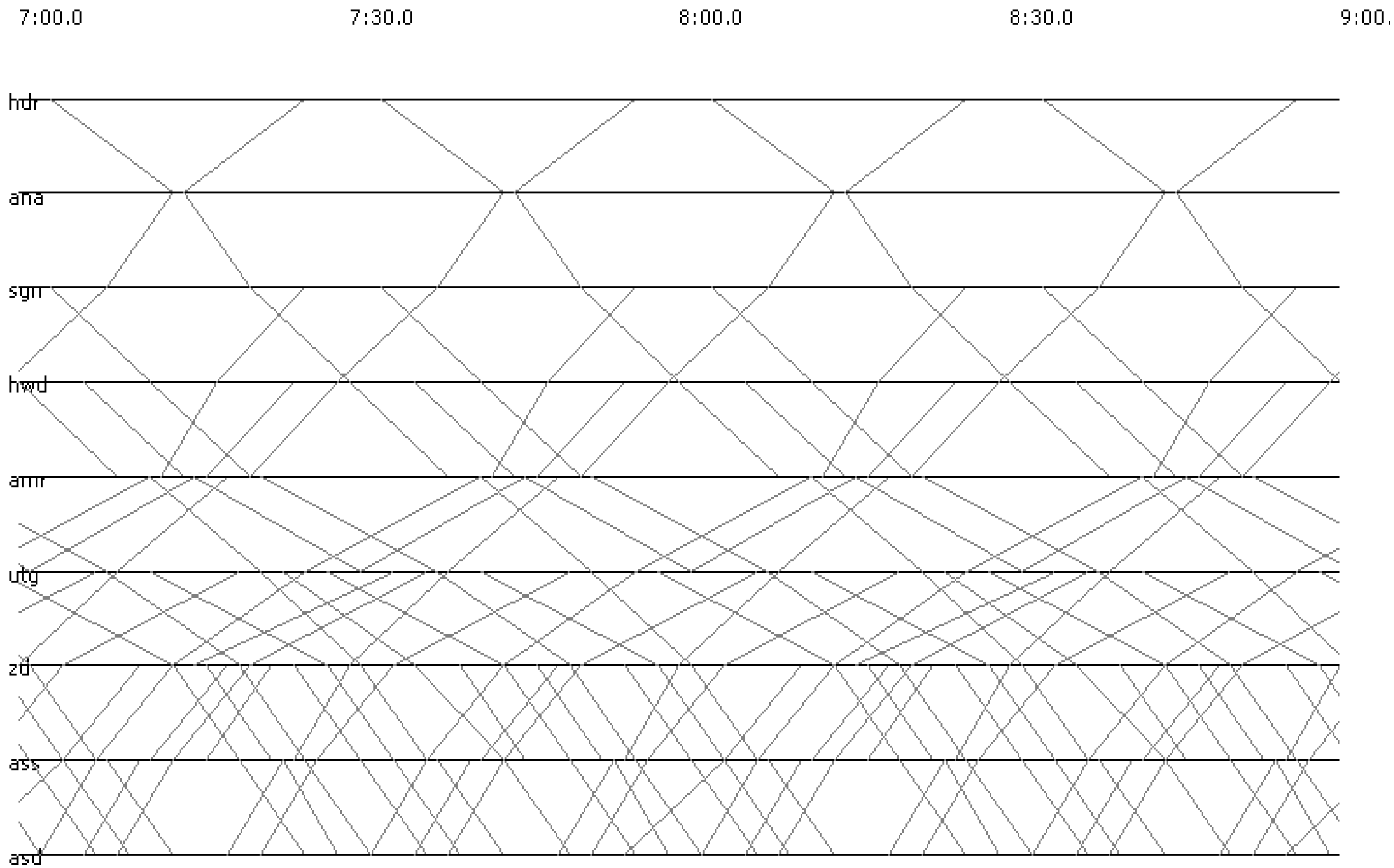



7:00.0
7:30.0
8:00.0
8:30.0
9:00.0
hdr
ana
sgn
hwd
amr
utg
zd
ass
asd
Current plan
utg1
hlm

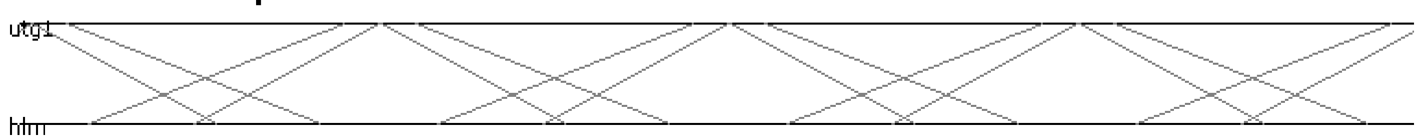


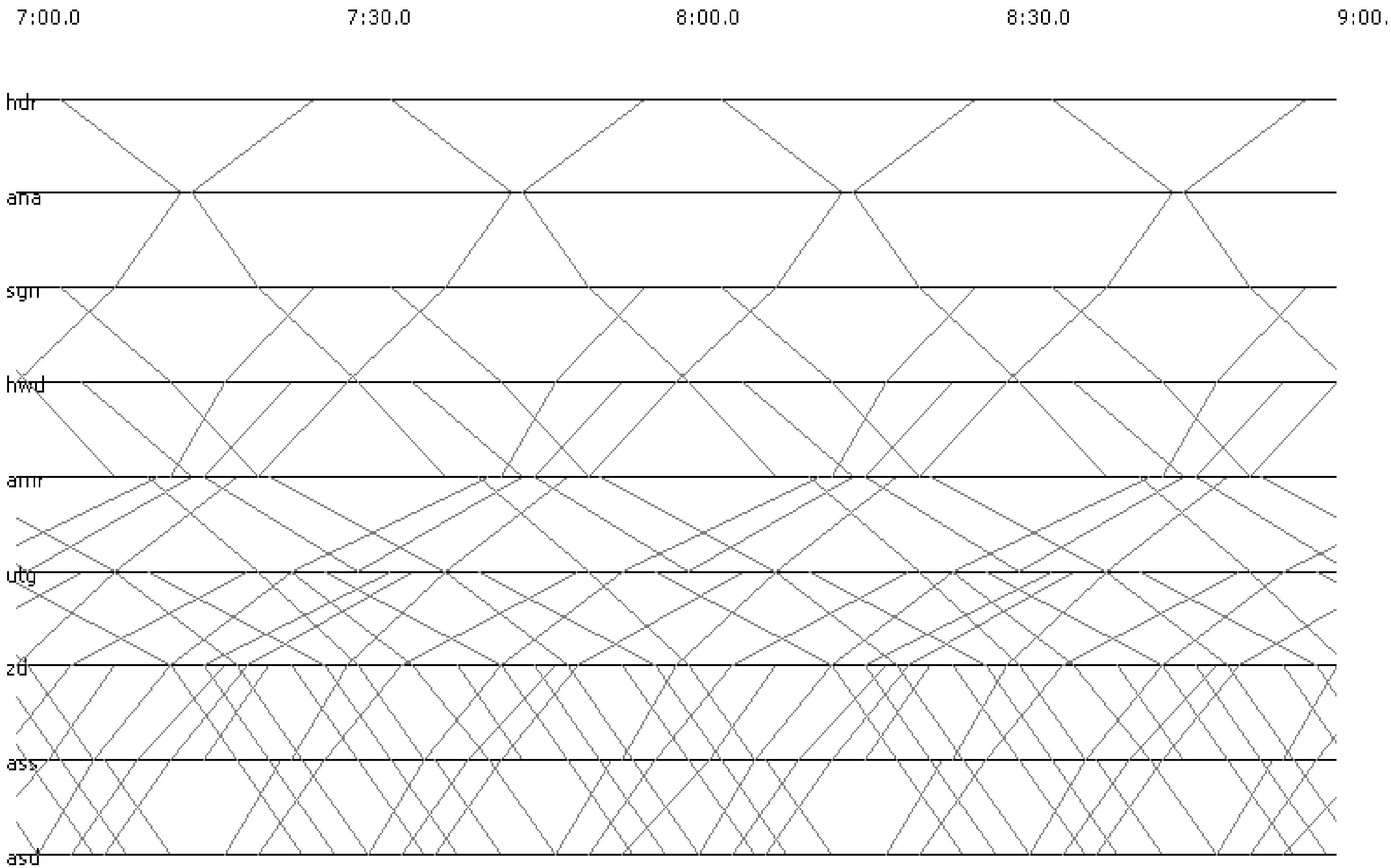
Current plan + realizations: 83.8%; average delay 1.64 min. (R = 200)



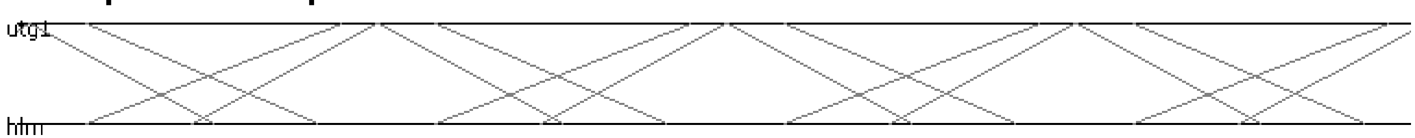


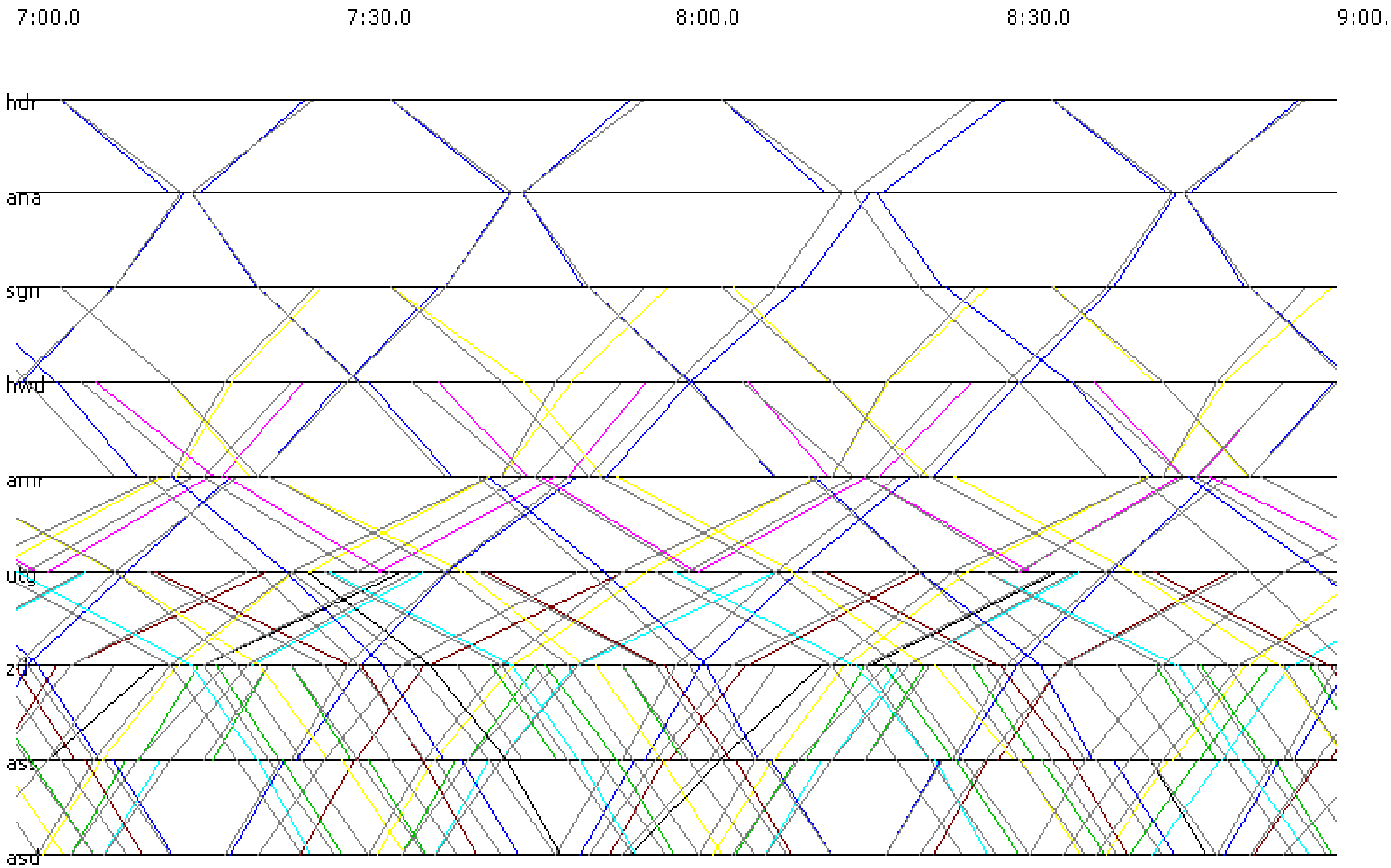
Current plan



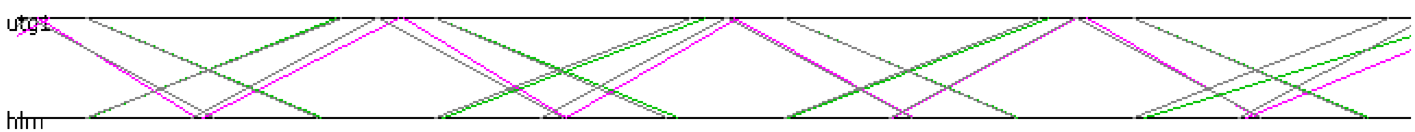


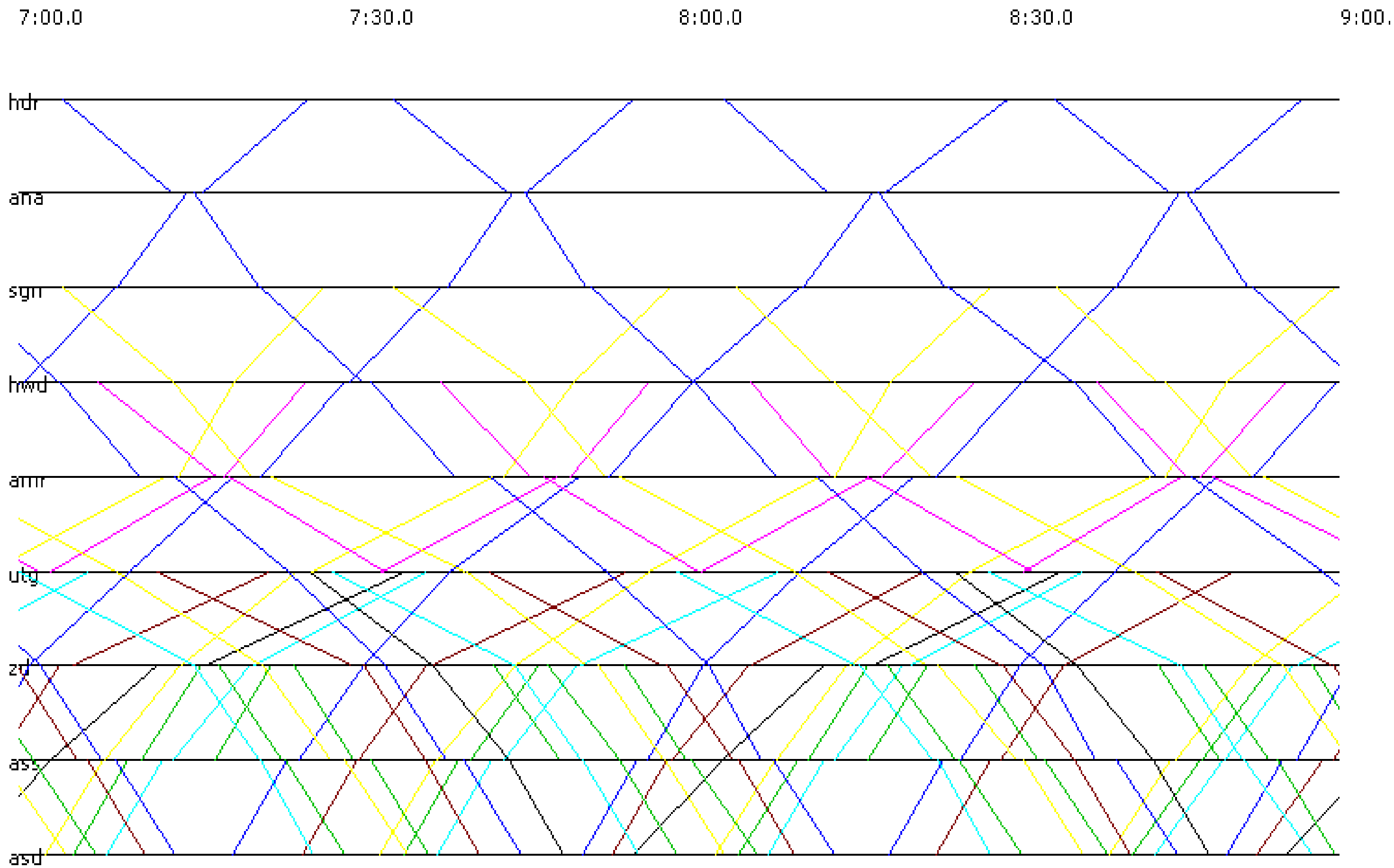
Improved plan



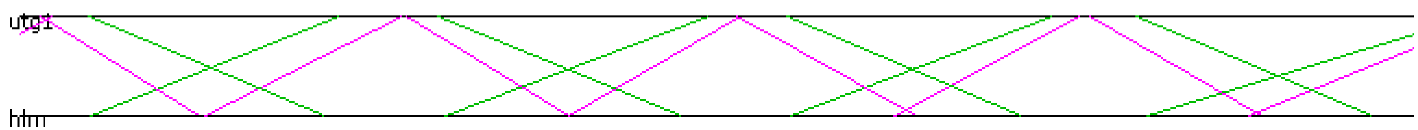


Improved plan + realizations: 86.3%; average delay 1.47 min. (R = 200)

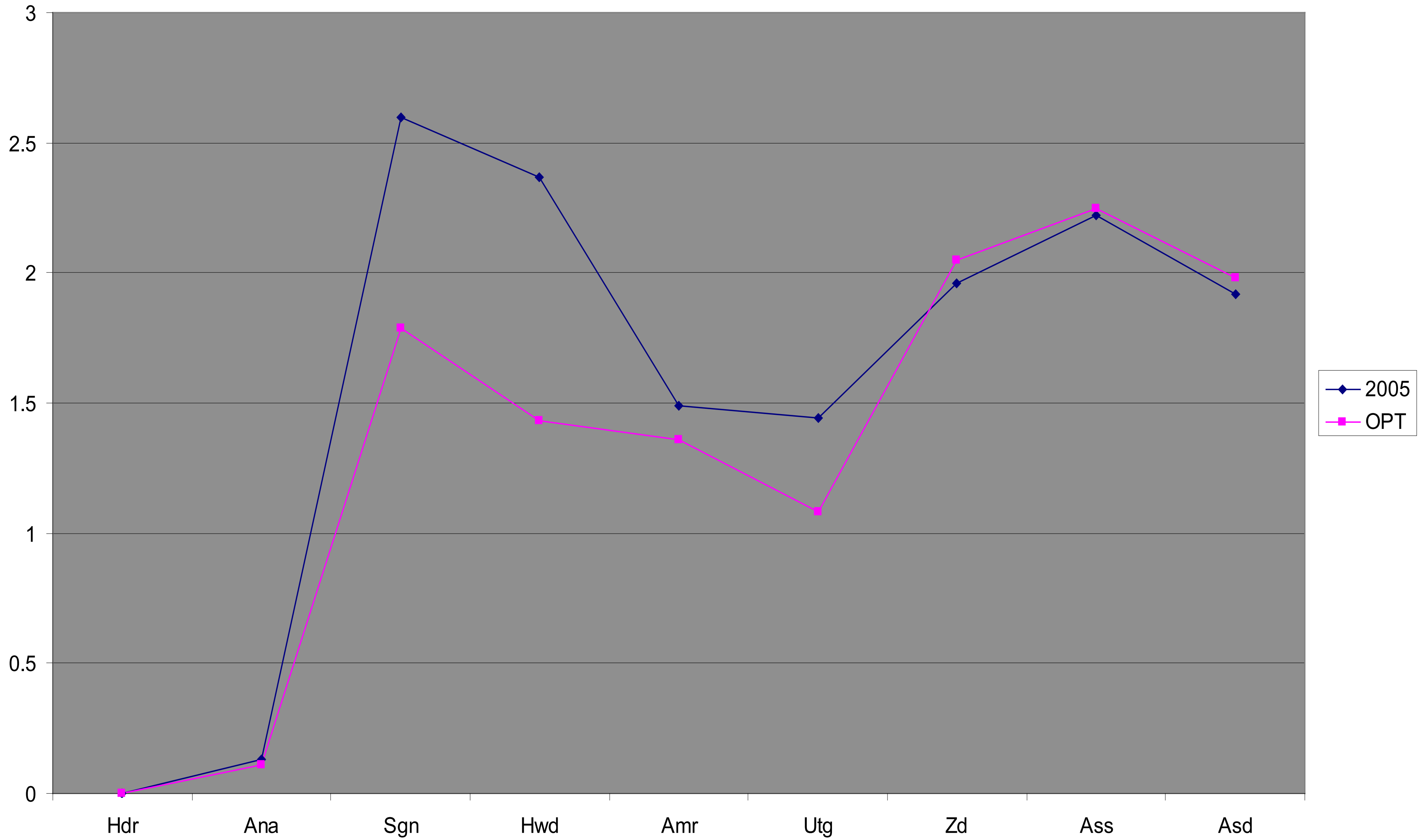




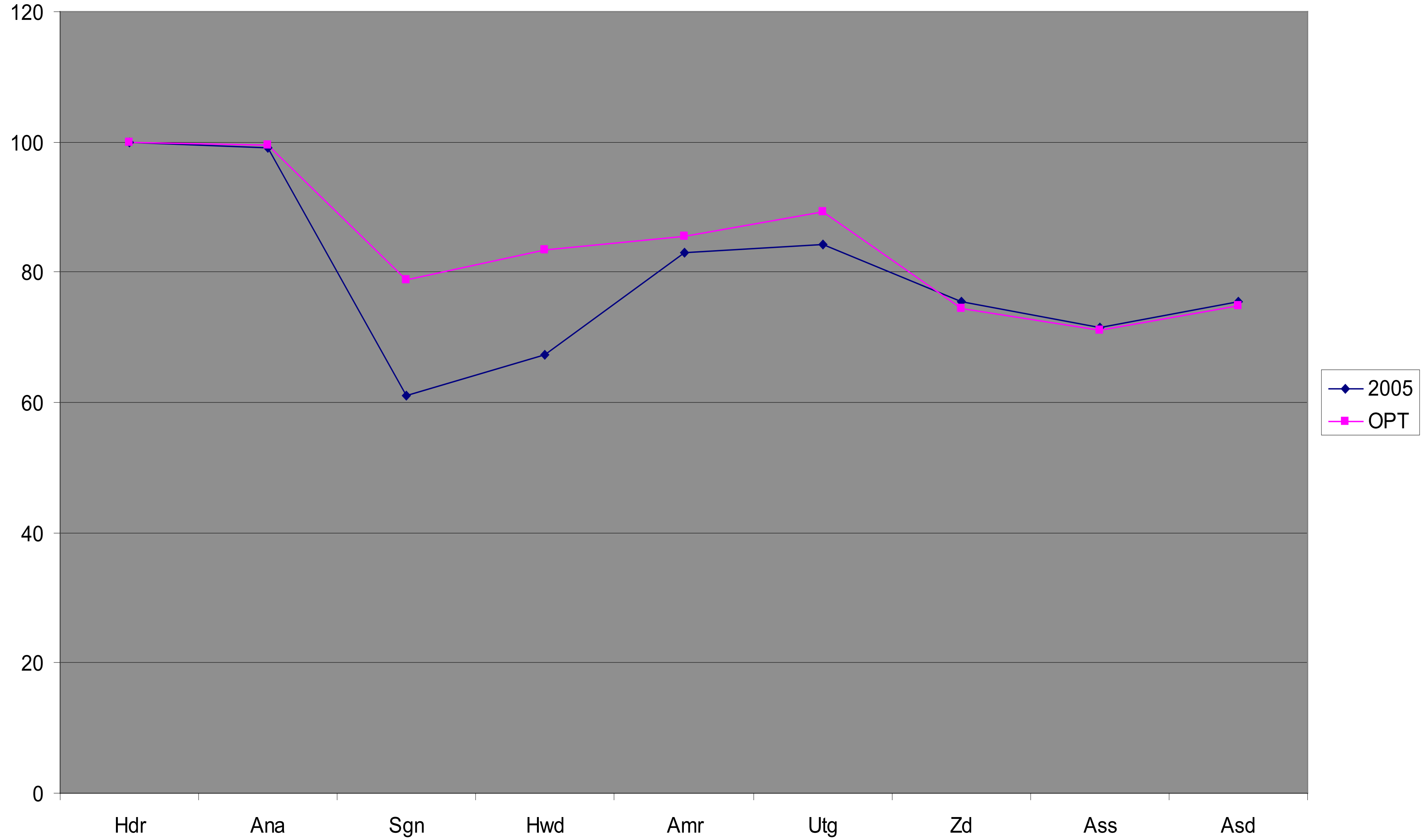
Realizations of improved plan: 86.3%; 1.47 min. (R = 200)



Average delay (North-South)



Punctuality (North-South)



Extensions:

- Solve IP problem by rounding LP solution

  Works reasonably well (Stut 2007)

- Use Dantzig-Wolfe decomposition for solving

  the still large LP

  two ways

  for every realisation problem a

  different subproblem

  for every train series a subproblem

  results

  no real improvement.

Contents

- Theory
- Robust train networks
- *Robust Shipping networks*
- Vehicle routing ideas

Introduction design of cyclic liner shipping networks

Liner shipping maintains a fixed route network with a fixed schedule, contrary to tramp shipping which follows demand.

Objective: develop quantitative tools to determine robust networks for liner shipping networks addressing

- Ship strings (which ports to visit in which order)
- Ship sizes
- Sailing speeds
- Frequency (preferrably once a week)
- Transfer ports

Example ship string NYK line EU2

NEX EU2: North Europe Express

NEX EMX : NEX EU1 : NEX EU2 : NEX EU3 : NEX EU4 : NEX EU5 : NEX EUM

Port Rotation

Origin	ETA/ETD
Kaohsiung	FRI/SAT
Shekou	SAT/SUN
Yantian	SUN/MON
Hong Kong	MON/TUE
Singapore	FRI/SAT
Le Havre	WED/THU
Amsterdam	FRI/FRI
Hamburg	SAT/MON
Antwerp	TUE/WED
Southampton	THU/FRI
Cagliari	TUE/WED
Jeddah	MON/MON
Jebel Ali	SAT/SUN
Singapore	SUN/MON
Kaohsiung	FRI/SAT

Turnaround days : 63

Weekly/Fixed Day Service

Key Transit Table

W/B	LEH	AMS	HAM	ANR	SOU
KHH	25	27	28	31	33
SHK	24	26	27	30	32
YTN	23	25	26	29	31
HKG	22	24	25	28	30
SIN	18	20	21	24	26

E/B	CAG	JED	JEA	SIN	KHH	SHK	YTN	HKG
LEH	12	18	23	31	36	37	38	39
AMS	11	17	22	30	35	36	37	38
HAM	8	14	19	27	32	33	34	35
ANR	6	12	17	25	30	31	32	33
SOU	4	10	15	23	28	29	30	31
CAG	-	5	10	18	23	24	25	26

Robustness of a given a cyclic route

- How much buffer time to insert in sailing time between ports?
  Tactical decision

- Recovery actions:
  - change speed (effective on longer routes)
  - pay for extra terminal handling capacity
  - cut and go (do not load all containers)

  to be decided upon the delay

Markov model

- State: port and the amount of delay
- Action: how much to increase speed
- Transition to: next port and new amount of delay
- Assumption: new delay independent of previous delay

• Notteboom (2006): port handling delays (strikes, problems, ship repairs) are majority of cases.

Markov LP formulation

Formulation

$$\min \sum_{i \in \mathcal{I}} \sum_{k \in \mathcal{K}} C_{ik} \pi_{ik} \qquad \textit{Minimize average cost}$$

$$\begin{aligned} \sum_{i \in \mathcal{I}} \sum_{k \in \mathcal{K}} \pi_{ik} &= 1 && \textit{Probabilities sum up to 1} \\ \sum_{k \in \mathcal{K}} \pi_{jk} - \sum_{i \in \mathcal{I}} \sum_{k \in \mathcal{K}} \pi_{ik} p_{ijk} &= 0 && \textit{Steady state constraint} \\ \pi_{ik} &\geq 0 && \textit{Nonnegative probabilities} \end{aligned}$$

State: i – port p and delay d (discretized)

Action k – extent to reduce delay

Speed optimisation in liner shipping

- So far studies on optimising speed for fixed routes for liner shipping trajectories, independent of the buffer problem.
  E.g. Wang (2012)

- Introducing the buffer time into a Markov decision chain and requiring that in a port always the same buffer is chosen destroys the independent action property of Markov decision chains: policy improvement is no longer optimal and cycling may occur.

- Solution: introduce integer variables in the LP formulation for MDPs, indicating the amount of buffer chosen per port (so a limited number).
  For small problems exact solution, for larger ones heuristics.

Relation speed – fuel consumption

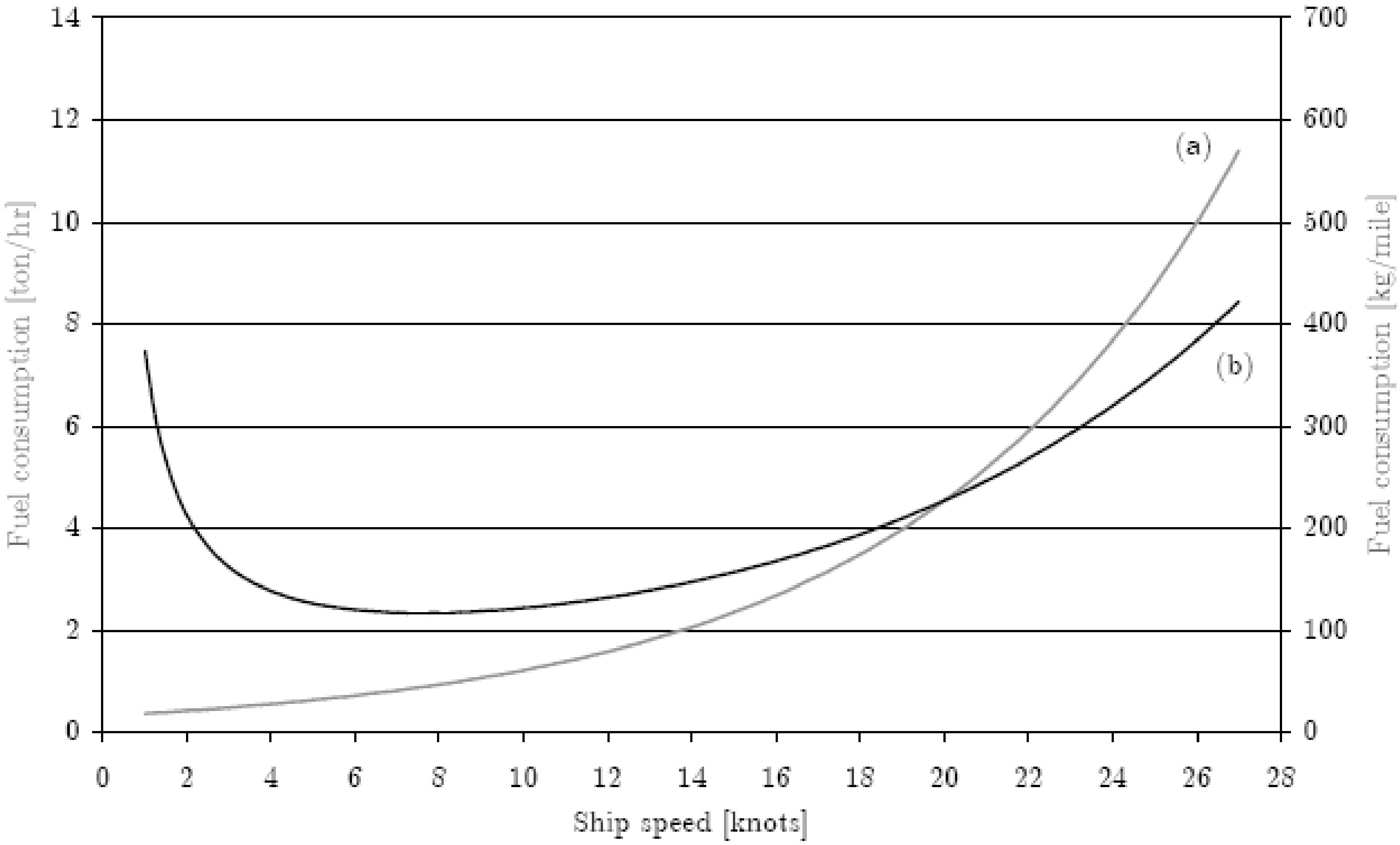


Figure 6.1: Example curve of hourly fuel consumption (a) and fuel consumption per mile (b) as a function of ship speed.

MIP formulation MDP model

Formulation

$$\min \sum_{i \in \mathcal{I}} \sum_{a \in \mathcal{A}} C_{ia} \pi_{ia}$$ *Minimize average cost*

$$\sum_{i \in \mathcal{I}} \sum_{a \in \mathcal{A}} \pi_{ia} = 1$$ *Probabilities sum up to 1*

$$\sum_{a \in \mathcal{A}} \pi_{ja} - \sum_{i \in \mathcal{I}} \sum_{a \in \mathcal{A}} \pi_{ia} p_{ija} = 0$$ *Steady state constraint*

$$\sum_{d \in \mathcal{D}} \sum_{k \in \mathcal{K}} \pi_{(pd),(kb)} \leq y_{pb}$$ *Correct buffer constraint*

$$\sum_{p \in \mathcal{P}} \sum_{b \in \mathcal{B}} B_b y_{pb} \leq M$$ *Total buffer cannot exceed maximum*

$$\sum_{b \in \mathcal{B}} y_{pb} = 1$$ *Unique buffer per port*

$$\pi_{ia} \geq 0$$ $i = (p, d), \; a = (k, b)$

$$y_{pb} \in \{0, 1\}$$

Example optimal policy (recovery action = increase speed)

- In port: take action with 3 hour gain, except in New Orleans, Felixtowe, Rotterdam and Lisbon without delay in which case action with 2 hour gain is chosen
  - Recovery actions are more expensive in these ports
- Between ports:

Delay	0	1	2	3	4	5	6	7	8	9
Charleston	0	0	0	1	2	3	4	5	-	-
Miami	0	0	0	1	2	3	4	-	-	-
Houston	0	0	0	0	1	2	3	-	-	-
New Orleans	0	0	0	1	2	3	4	-	-	-
Antwerp	0	0	0	0	1	2	3	-	-	-
Felixtowe	0	0	0	0	0	0	0	1	2	-
Bremerhaven	0	0	0	0	0	1	2	3	4	5
Rotterdam	0	0	0	0	1	2	3	4	5	5
Lisbon	0	0	0	0	0	1	2	3	-	-

  - Recovery actions most expensive when leaving Antwerp, Felixtowe and Bremerhaven

Application shipping line

- Most uncertainty in port handling, more than in weather
- Uncertainty due to varying cargo loads. Terminal agrees to load /unload x boxes with a certain performance (e.g 100 / hr)
- Large speed variations exist between ports due to all kind of terminal restrictions -> not really optimal
- Output of model is very useful in negotiating berthing windows with terminals.
- Mulder et al (2012).

Further research

- Performance is monotonic in buffer: cost decrease as function of a continuous buffer. Yet Markov chain has discrete states.

- Prove convexity in buffer time
combine continuous and discrete model yields that a continuous version of Policy improvement is optimal!

Contents

- Theory
- Robust train networks
- Robust Shipping networks
- *Vehicle routing ideas*

Vehicle routing options

Various approaches

- Every day new routes:
  flexible, yet much variations
  dependence on travel time predictions

- Fixed routes, fixed drivers

  advantage: drivers known to customers,
  drivers more familiar with routes
  yet what to do with demand fluctuations?

Vehicle routing robustness

Robustness against

travel times & handling times

train buffer approach can be used even simpler (no cycles), yet time windows can play a role and total driving time may be limited

Recovery actions:
skip part of route and/or use other drivers

change order of visiting clients: which ones first? Highly or lowly variable? Results lack.

Vehicle routing robustness

Stochastic Vehicle routing with fluctuating demand:

Many papers assume that when truck capacity is used, trucks return to depot, unload and continue trip from that point

Spliet et al (2014) – introduce a penalty function for not serving customers on fixed route and determine new routes beforehand taking actual demand into account.

Consistent Vehicle routing robustness

Make sure the same drivers visit customers Groër (2009), while making sure that customers are delivered at about the same time.

Spliet & Dekker (2014) – consider a case where α% of the customers need to have the same driver. They give an exact formulation and investigate heuristics based upon customer aggregation

Conclusions

- Many different ways of defining robustness
- Important concept in practice
- New models and approaches are able to make statements on the "optimal" amount of buffer in the light of all kind of recovery actions.
- Results can be extended in several ways and several transportation systems.