Automation possibilities in metro-rail operations planning

Madhu Belur, Electrical Engineering Dept, IIT Bombay

99874 66 279, belur@iitb.ac.in

This talk will soon be available at

www.ee.iitb.ac.in/%7Ebelur/talks

www.ee.iitb.ac.in/%7Ebelur/railways/workshop

www.ee.iitb.ac.in/%7Ebelur/railways

Group: Narayan Rangaraj and others from IIT Bombay

narayan.rangaraj@iitb.ac.in

I-Metro-Workshop, Mar 2023, I-Metro workshop

Outline of the talk

- Need for software tools in operations planning
- Data availability/digitization: machine learning/artificial intelligence
- Tools recently developed by our group in railway operations
- Other tools (non-railway) developed by our group
- FOSS and ‘development-participation & owning’ of tools
- Suggestions about tool-specifications: primarily from ‘shop floor’

Outline of the talk

- Need for software tools in operations planning
- Data availability/digitization: machine learning/artificial intelligence
- Tools recently developed by our group in railway operations
- Other tools (non-railway) developed by our group
- FOSS and 'development-participation & owning' of tools
- Suggestions about tool-specifications: primarily from 'shop floor'

Quantity of service: more throughput, more services, more persons served

Outline of the talk

- Need for software tools in operations planning
- Data availability/digitization: machine learning/artificial intelligence
- Tools recently developed by our group in railway operations
- Other tools (non-railway) developed by our group
- FOSS and 'development-participation & owning' of tools
- Suggestions about tool-specifications: primarily from 'shop floor'

Quantity of service: more throughput, more services, more persons served

Quality of service: high-average speed, low waiting-time, high frequency, medium occupancy within metros)

Outline of the talk

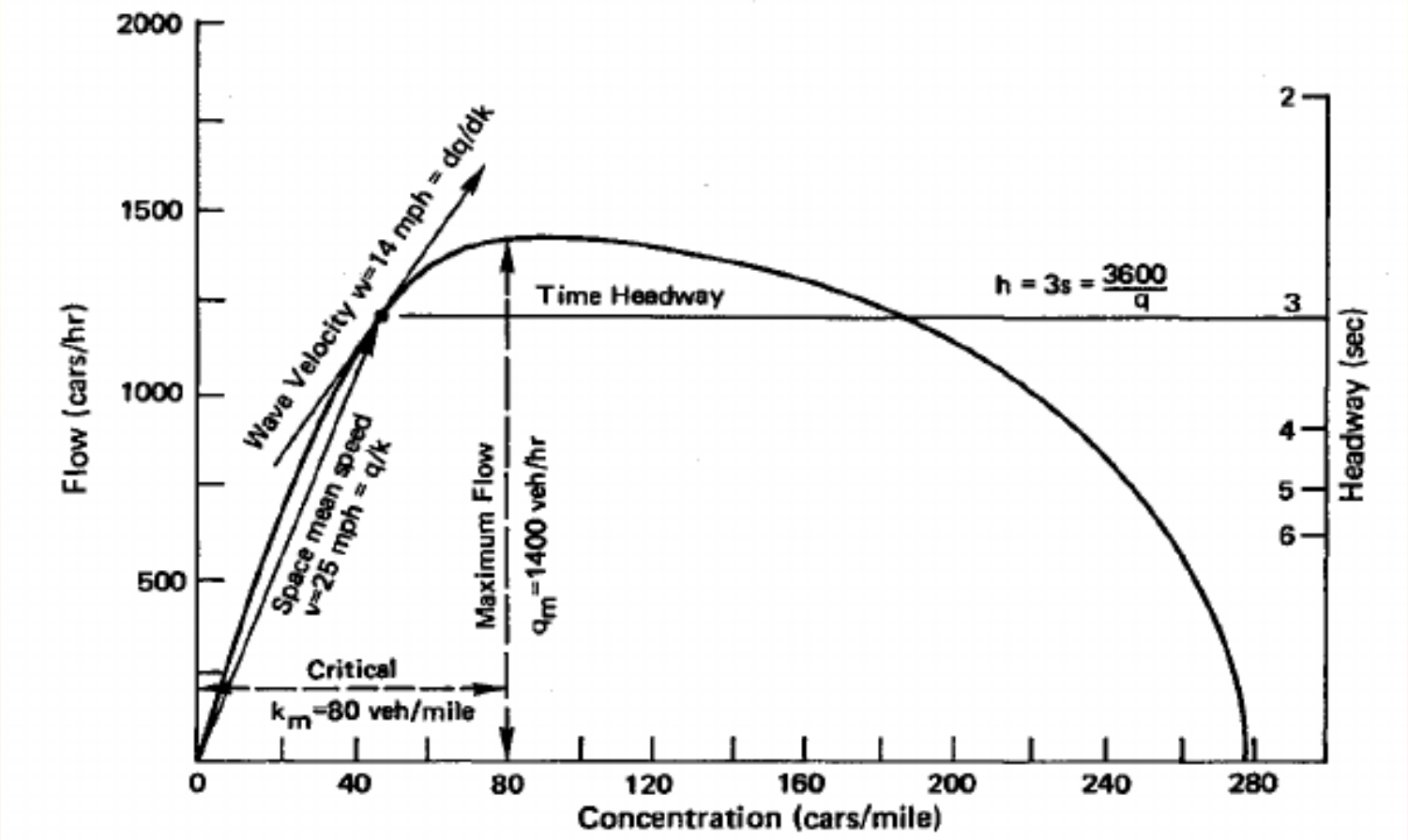
- Need for software tools in operations planning
- Data availability/digitization: machine learning/artificial intelligence
- Tools recently developed by our group in railway operations
- Other tools (non-railway) developed by our group
- FOSS and 'development-participation & owning' of tools
- Suggestions about tool-specifications: primarily from 'shop floor'

Quantity of service: more throughput, more services, more persons served

Quality of service: high-average speed, low waiting-time, high frequency, medium occupancy within metros)
Need to operate at high quantity/quality of service

Vehicles flow models: flow rate (throughput) vs density

(Partly relevant for trains)

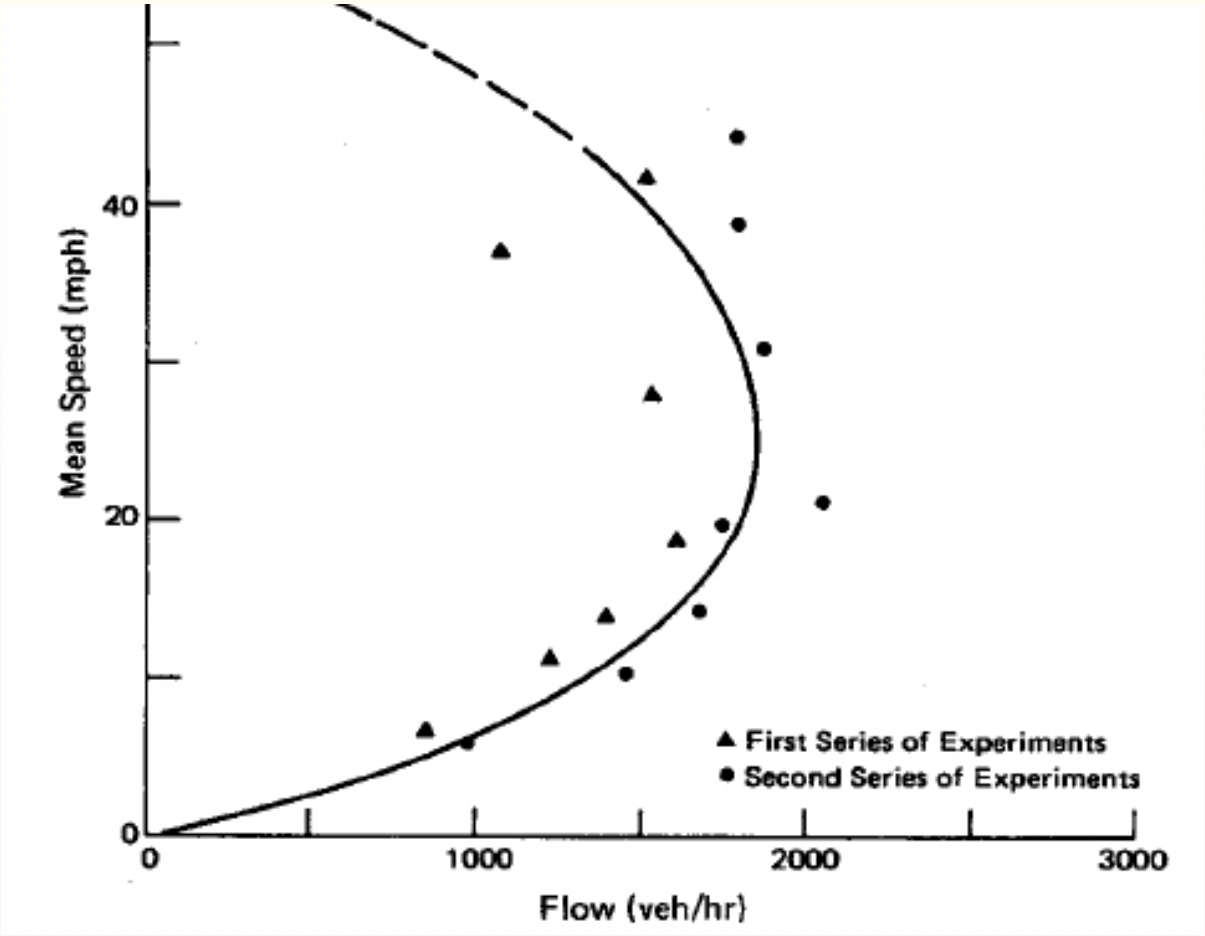


Source: Traffic Flow Theory, D.L. Gerlough and M.J. Huber (1975)

Throughput peaks at an optimum density

Vehicles flow models: average speed vs flow

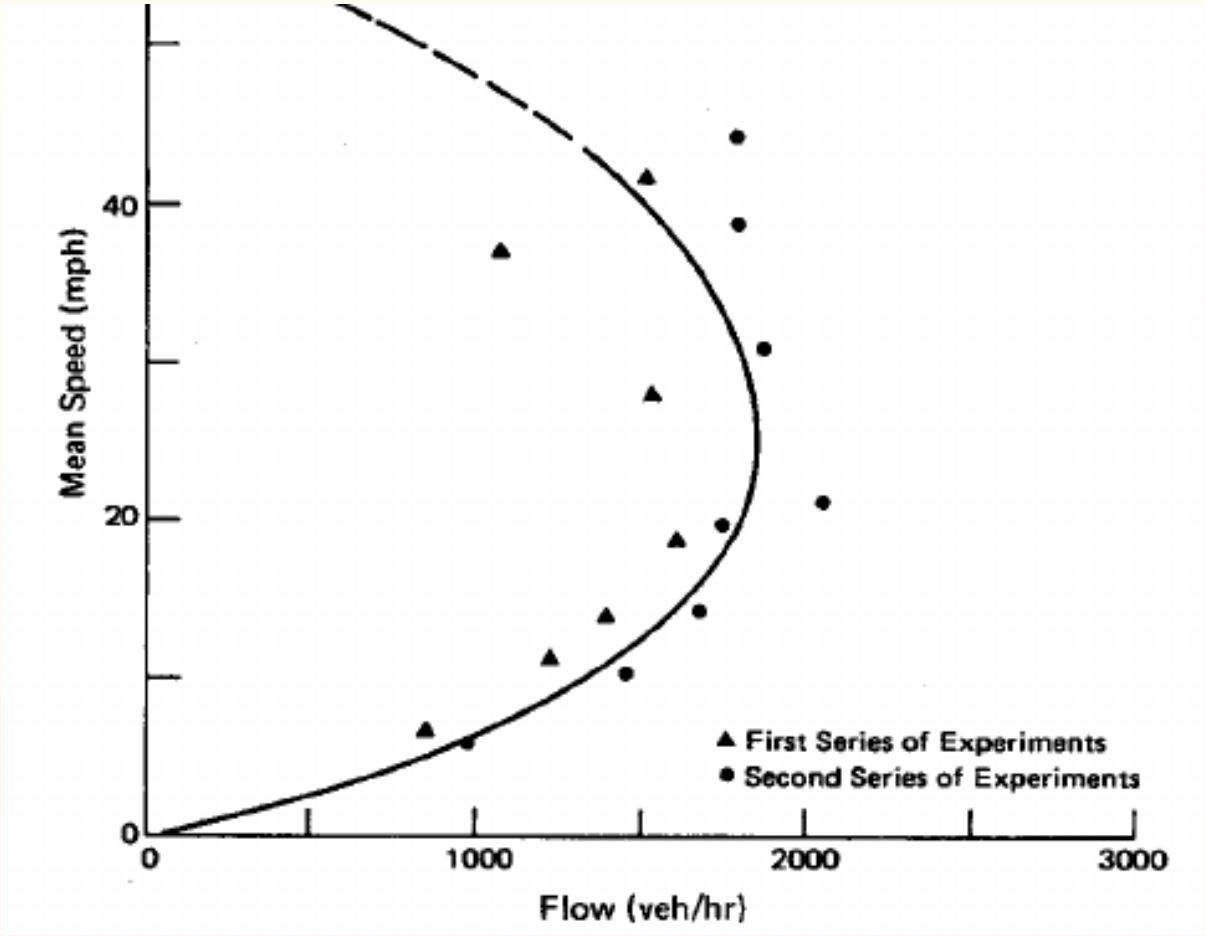
Quality of service vs quantity of service



Source: Traffic Flow Theory, D.L. Gerlough and M.J. Huber (1975)

Vehicles flow models: average speed vs flow

Quality of service vs quantity of service

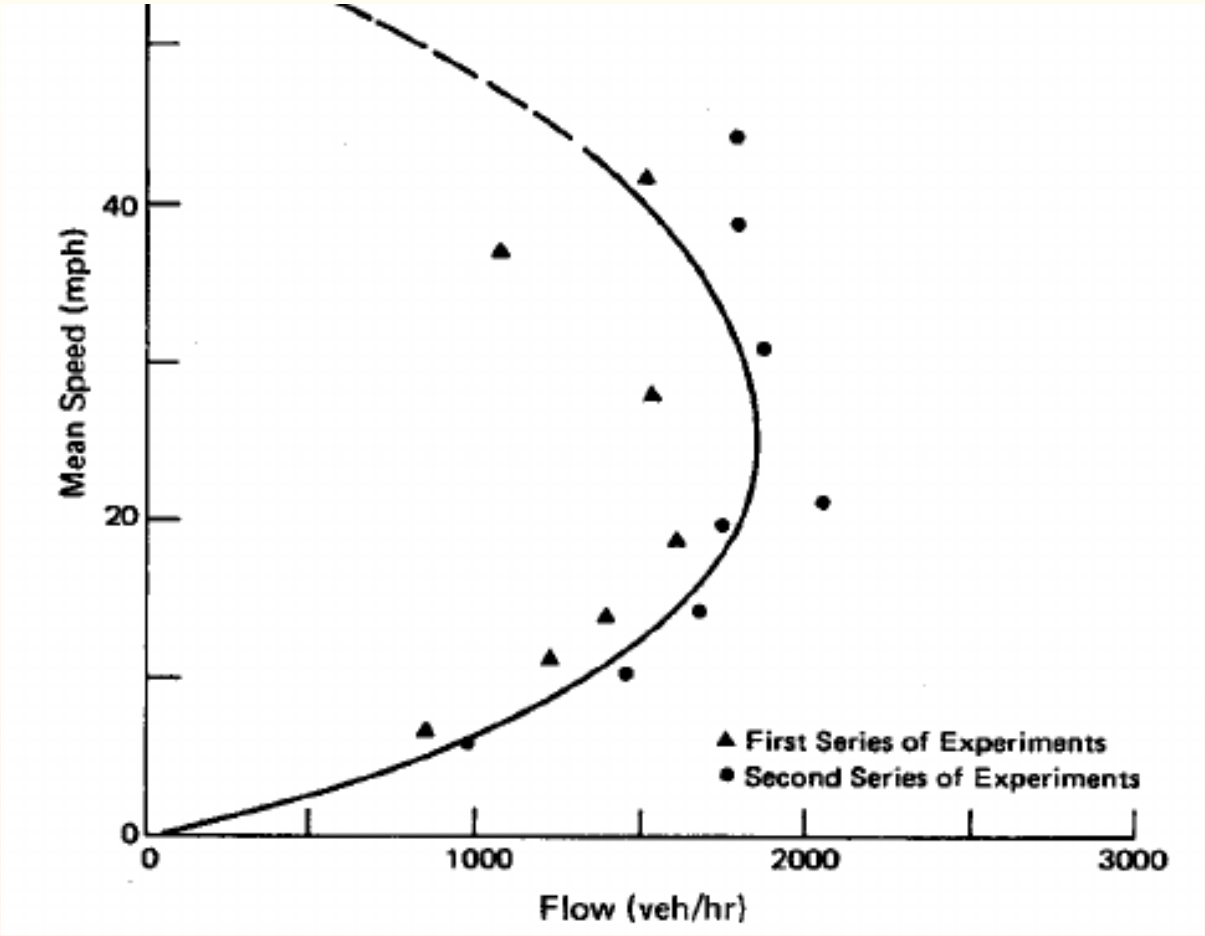


Source: Traffic Flow Theory, D.L. Gerlough and M.J. Huber (1975)

Bad for throughput: too low speed, also

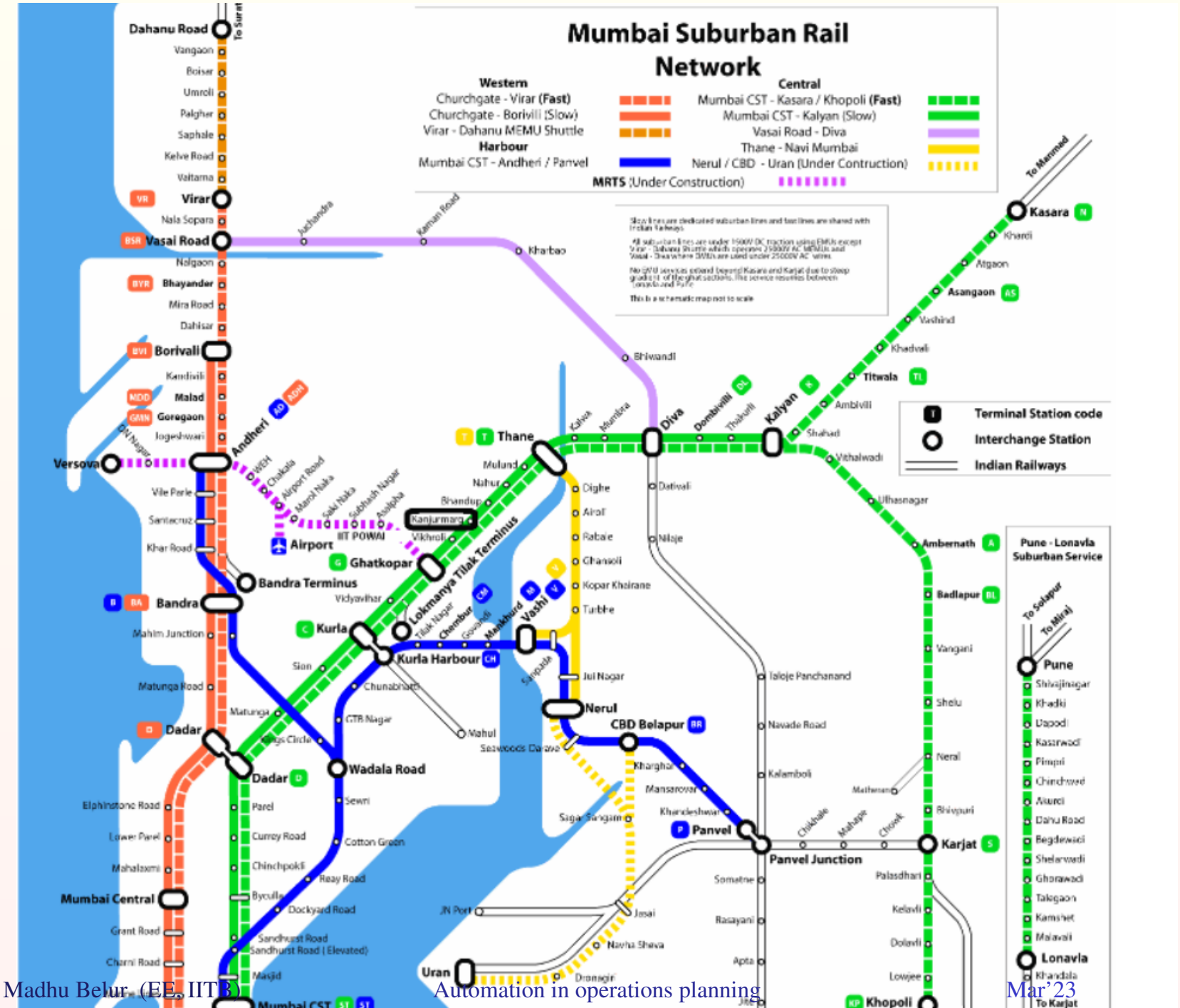
Vehicles flow models: average speed vs flow

Quality of service vs quantity of service



Source: Traffic Flow Theory, D.L. Gerlough and M.J. Huber (1975)

Bad for throughput: too low speed, also too high speed!


Mumbai Suburban Rail Network
Western
Churchgate - Virar (Fast)
Churchgate - Borivili (Slow)
Virar - Dahanu MEMU Shuttle
Harbour
Mumbai CST - Andheri / Panvel
Central
Mumbai CST - Kasara / Khopoli (Fast)
Mumbai CST - Kalyan (Slow)
Vasai Road - Diva
Thane - Navi Mumbai
Nerul / CBD - Uran (Under Contruction)
MRTS (Under Construction)
Terminal Station code
Interchange Station
Indian Railways
Pune - Lonavla Suburban Service
Dahanu Road
To Surat
Vangaon
Boisar
Umroli
Palghar
Saphale
Kelve Road
Vaitarna
Virar
Nala Sopara
Vasai Road
Nalgaon
Bhayander
Mira Road
Dahisar
Borivali
Kandivili
Malad
Goregaon
Jogeshwari
Andheri
DN Nagar
Versova
Vile Parle
Santacruz
Khar Road
Bandra Terminus
Bandra
Mahim Junction
Matunga Road
Dadar
Elphinstone Road
Lower Parel
Mahalaxmi
Mumbai Central
Grant Road
Charni Road
Mumbai CST
Juchandra
Kaman Road
Kharbao
Bhiwandi
Airport
IIT POWAI
Ghatkopar
Kanjurmarg
Vikhroli
Bhandup
Mulund
Nahur
Thane
Kalwa
Mumbra
Diva
Dombivili
Thakurli
Kalyan
Titwala
Ambivli
Shahad
Vithalwadi
Ulhasnagar
Ambernath
Badlapur
Vangani
Shelu
Neral
Bhivpuri
Karjat
Kasara
Khardi
Atgaon
Asangaon
Vashind
Khadvali
To Manmad
Digha
Airoli
Rabale
Ghansoli
Kopar Khairane
Turbhe
Dativali
Nilaje
Lokmanya Tilak Terminus
Vidyavihar
Kurla
Kurla Harbour
Vashi
Sion
Chunabhatti
Tilak Nagar
Chembur
Govandi
Mankhurd
Sanpada
Jui Nagar
Nerul
CBD Belapur
Seawoods Darave
Kharghar
Mansarovar
Khandeshwar
Panvel
Panvel Junction
Taloje Panchanand
Navade Road
Kalamboli
Matheran
Palasdhari
Kelavli
Dolavli
Lowjee
Khopoli
Somatne
Rasayani
Apta
Jasai
Navha Sheva
JN Port
Uran
Dronagiri
Dadar
Parel
Currey Road
Chinchpokli
Reay Road
Cotton Green
Wadala Road
Sewri
Byculla
Dockyard Road
Sandhurst Road
Sandhurst Road (Elevated)
Masjid
Matunga
King's Circle
GTB Nagar
Mahul
Saiwoods Darave
Saga Sangam
Chikhale
Mahape
Chowk
Pune
Shivajinagar
Khadki
Dapodi
Kasarwadi
Pimpri
Chinchwad
Akurdi
Dehu Road
Begdewadi
Shelarwadi
Ghorawadi
Talegaon
Kamshet
Malavali
Lonavla
Khandala
To Karjat
To Solapur
To Miraj

Constraints

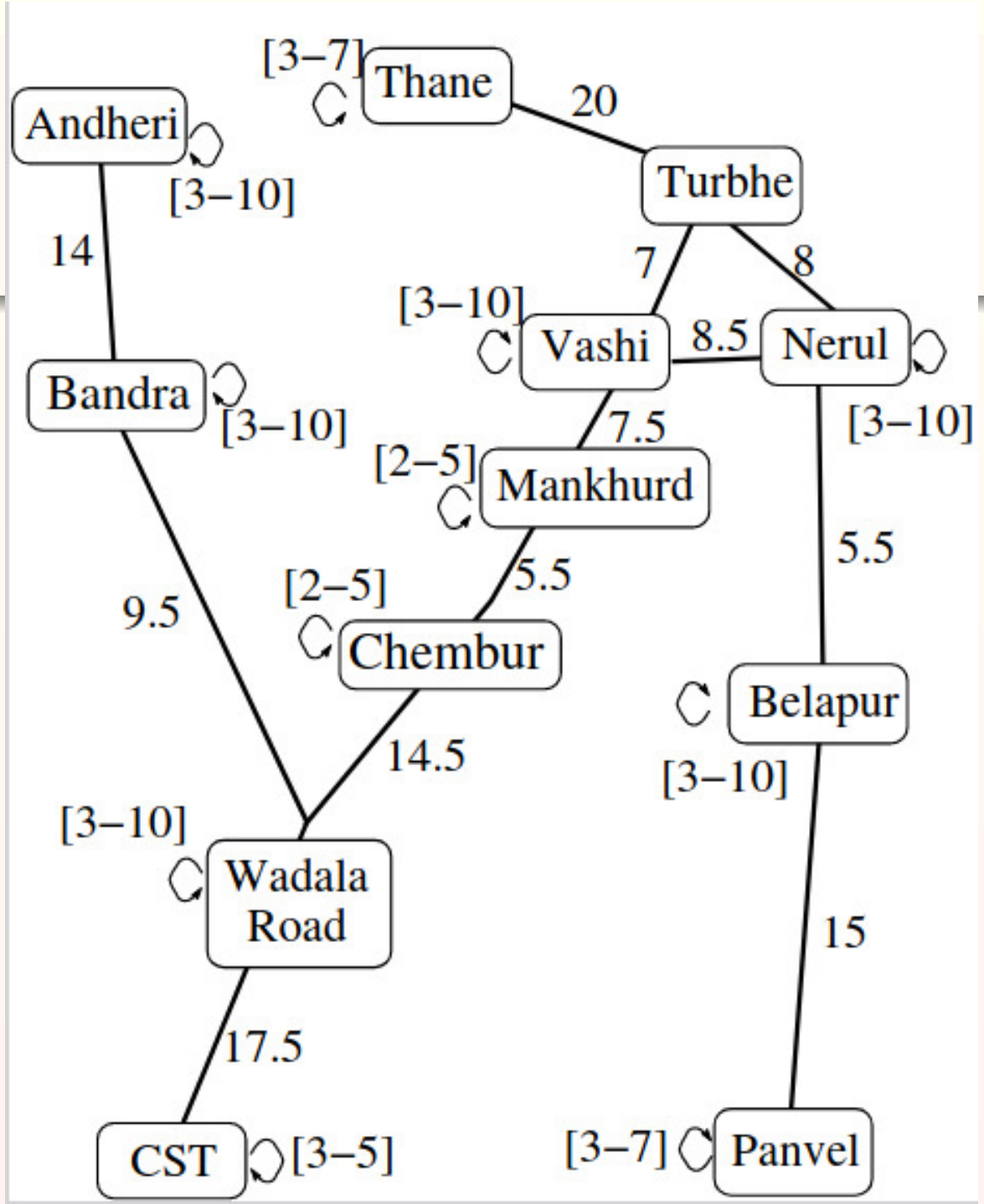
Hard constraints:

- Headway: safety
- Frequency of service: demands: origin/destination services
- Traversal times
- Turn-around constraints: ‘rake linking’
- Platform dwell/occupancy constraints

Soft constraints:

- Spacing between consecutive ‘similar’ services

Mumbai harbour-line

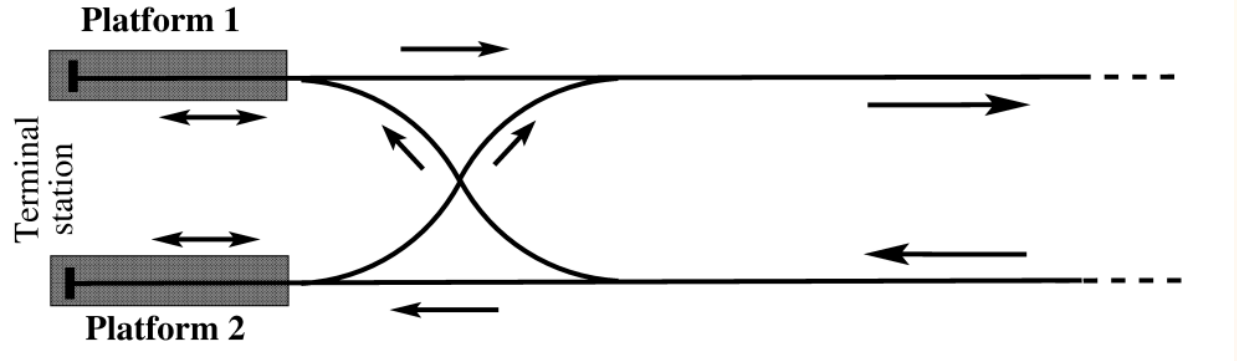


Constraints tightest at CSTM ("VT")

Platform vacating constraints:
An exit (from platform 2) hinders consecutive two entries

Constraints tightest at CSTM ("VT")

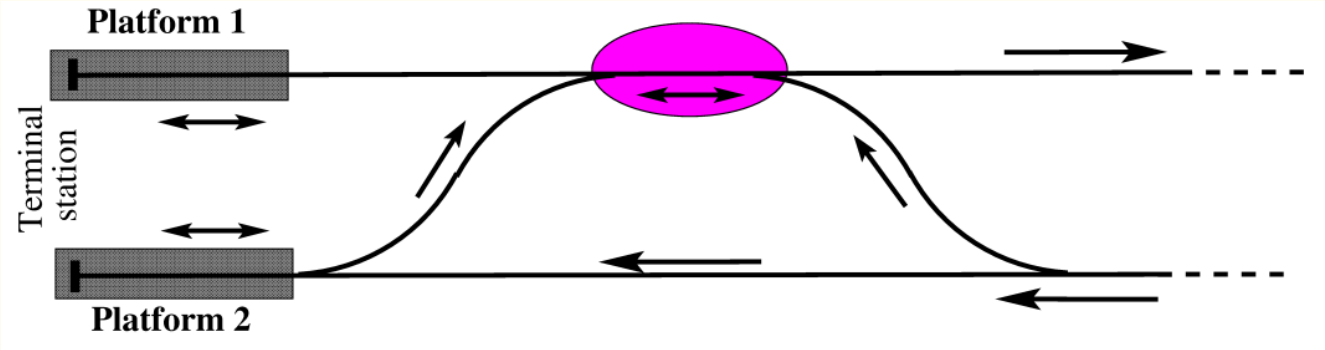
Platform vacating constraints:
An exit (from platform 2) hinders consecutive two entries



- Ideal: 'Scissor' crossing
- Fastest exit and least 'hindrance' to following entries
- Curves are on only a small distance: hence smaller distance of low-speed constraints
- Pretty expensive (but not exorbitant)

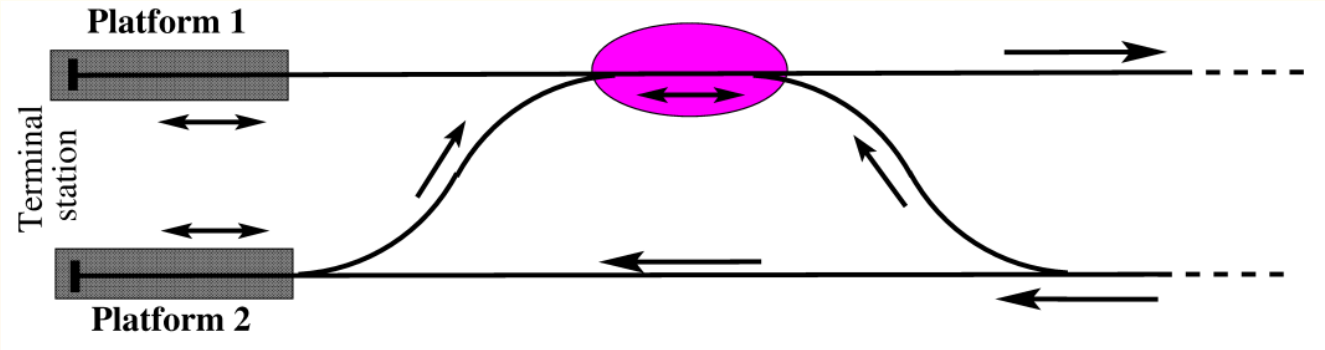
Platform-asymmetric layouts at a terminal

Not-too-bad layout:



Platform-asymmetric layouts at a terminal

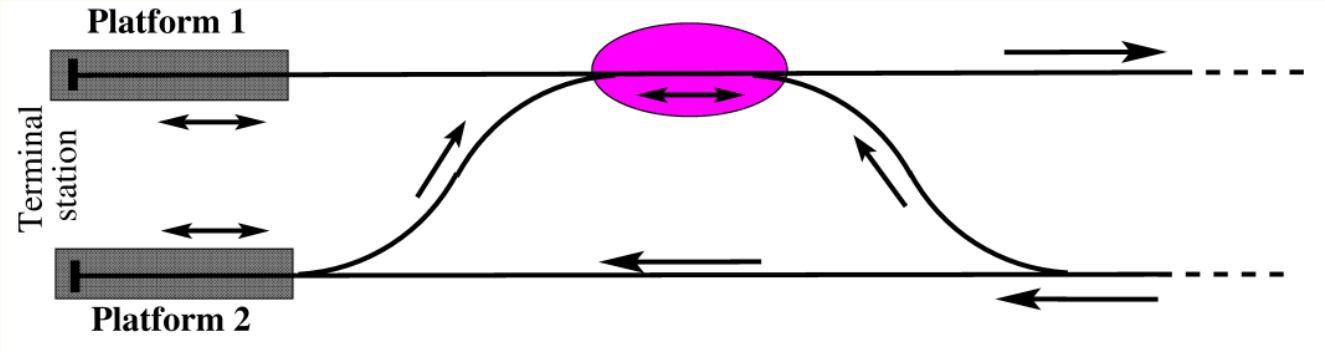
Not-too-bad layout:



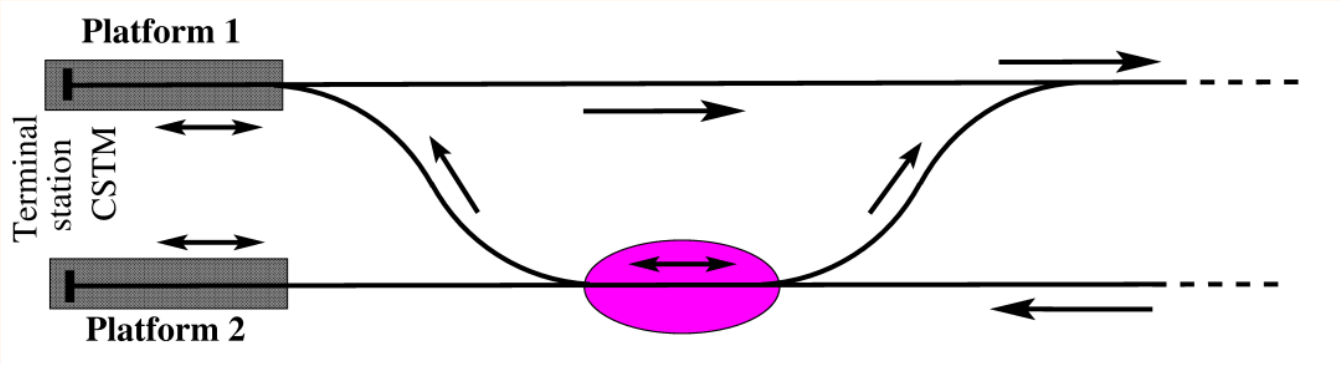
But

Platform-asymmetric layouts at a terminal

Not-too-bad layout:



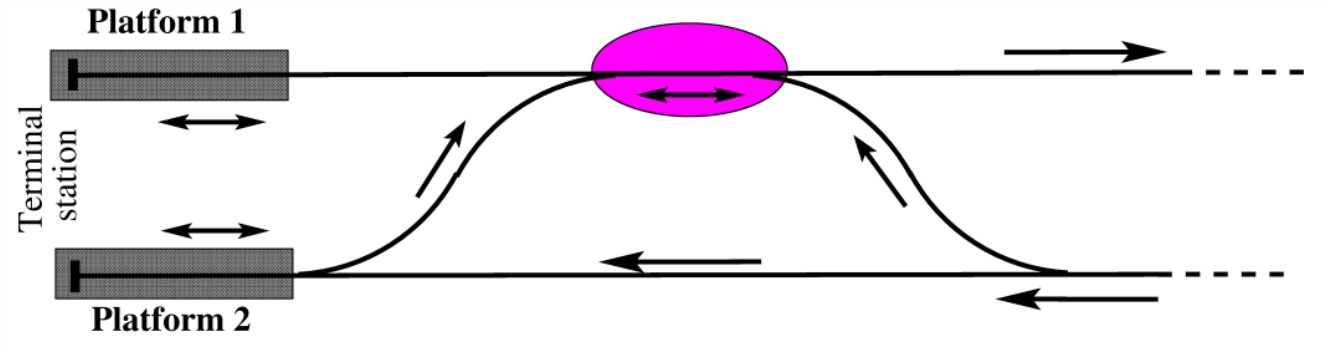
But



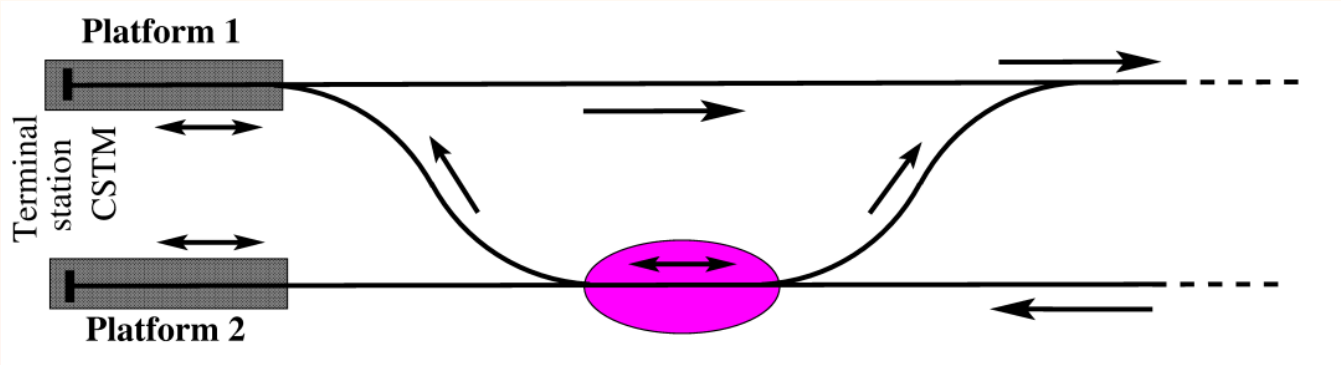
Higher hindrance layout: at CSTM (due to curvature constraints: not displayed)

Platform-asymmetric layouts at a terminal

Not-too-bad layout:



But

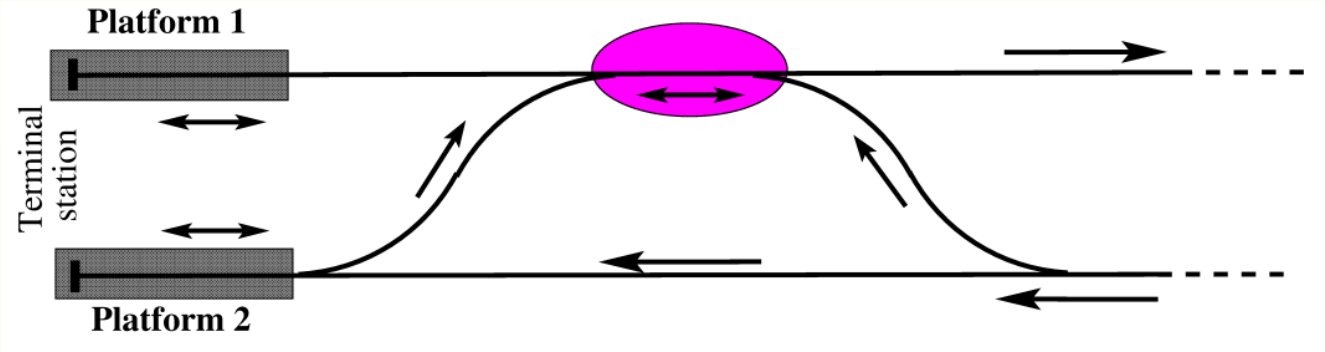


Higher hindrance layout: at CSTM (due to curvature constraints: not displayed)

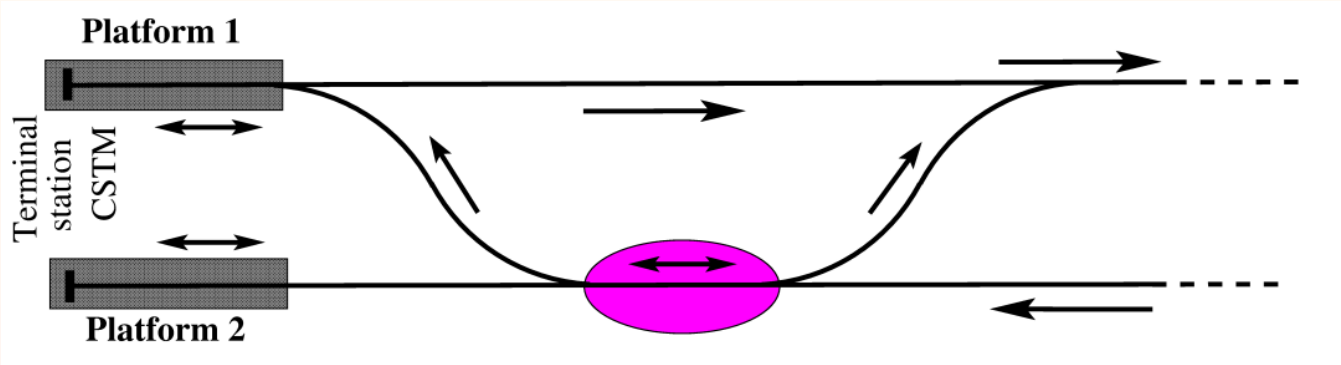
- Exit from platform 2 delays entry into platform 1

Platform-asymmetric layouts at a terminal

Not-too-bad layout:



But

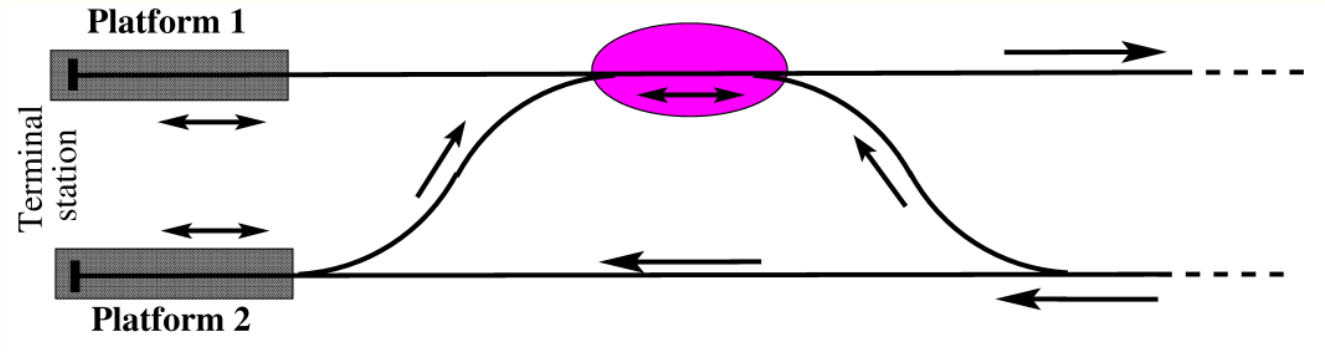


Higher hindrance layout: at CSTM (due to curvature constraints: not displayed)

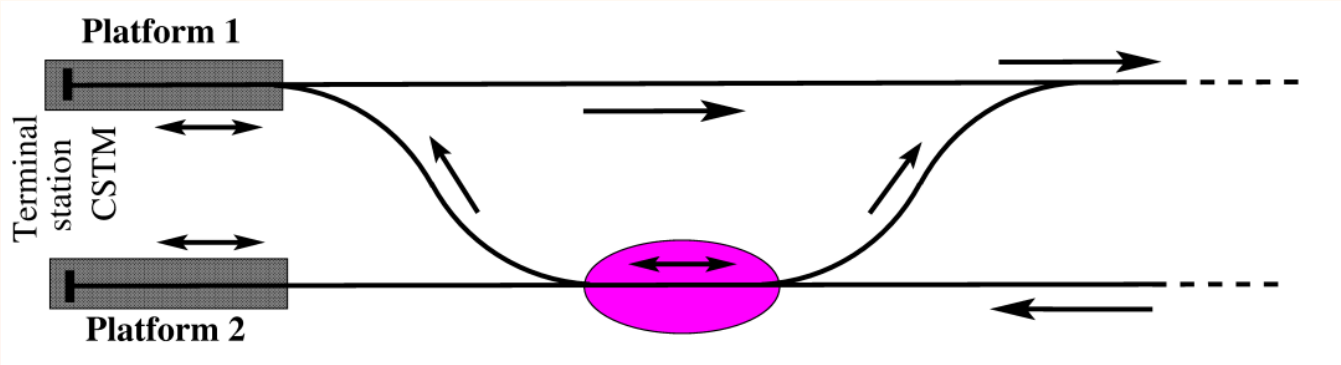
- Exit from platform 2 delays entry into platform 1
- Higher hindrance in the lower layout

Platform-asymmetric layouts at a terminal

Not-too-bad layout:



But



Higher hindrance layout: at CSTM (due to curvature constraints: not displayed)

- Exit from platform 2 delays entry into platform 1
- Higher hindrance in the lower layout
- Lower layout: curve after bottleneck portion: slower exit

Inputs to the tool

Inputs: different infrastructural parameters:

- Stations, network layout
- Passenger demands, traversal times
- Turnaround times at terminals

Output: timetable meeting all constraints with least number of rakes

Case Study: Harbour Line Network

Current Status: Mumbai Harbour line network

- 12 major stations
- 49 rakes in service in peak time - 38 for Harbour and 11 for Trans Harbour

For a 3 hour peak timetable, about:

- 750 Departure-Arrival Events
- 40500 Headway Variables (Integer or Binary)
- 10000 Linkage Variables (Integer or Binary)
- 6500 Platform Variables (Integer or Binary)
- 3200 Precedence Variables for Platform Allocation (Binary)

Case Study: Harbour Line Network

Resources Used:

- Modelling: AMPL to model all the constraints

Case Study: Harbour Line Network

Resources Used:

- Modelling: AMPL to model all the constraints
- Solver: Gurobi

Case Study: Harbour Line Network

Resources Used:

- Modelling: AMPL to model all the constraints
- Solver: Gurobi
- Python & Bash Scripts for pre-processing and post-processing

Case Study: Harbour Line Network

Resources Used:

- Modelling: AMPL to model all the constraints
- Solver: Gurobi
- Python & Bash Scripts for pre-processing and post-processing

Trying Pyomo and PuLP to replace AMPL/Gurobi

Case Study: Harbour Line Network

Resources Used:

- Modelling: AMPL to model all the constraints
- Solver: Gurobi
- Python & Bash Scripts for pre-processing and post-processing

Trying Pyomo and PuLP to replace AMPL/Gurobi
Status: able to save 3 rakes (out of 45 of the current timetable)

Case Study: Harbour Line Network

Resources Used:

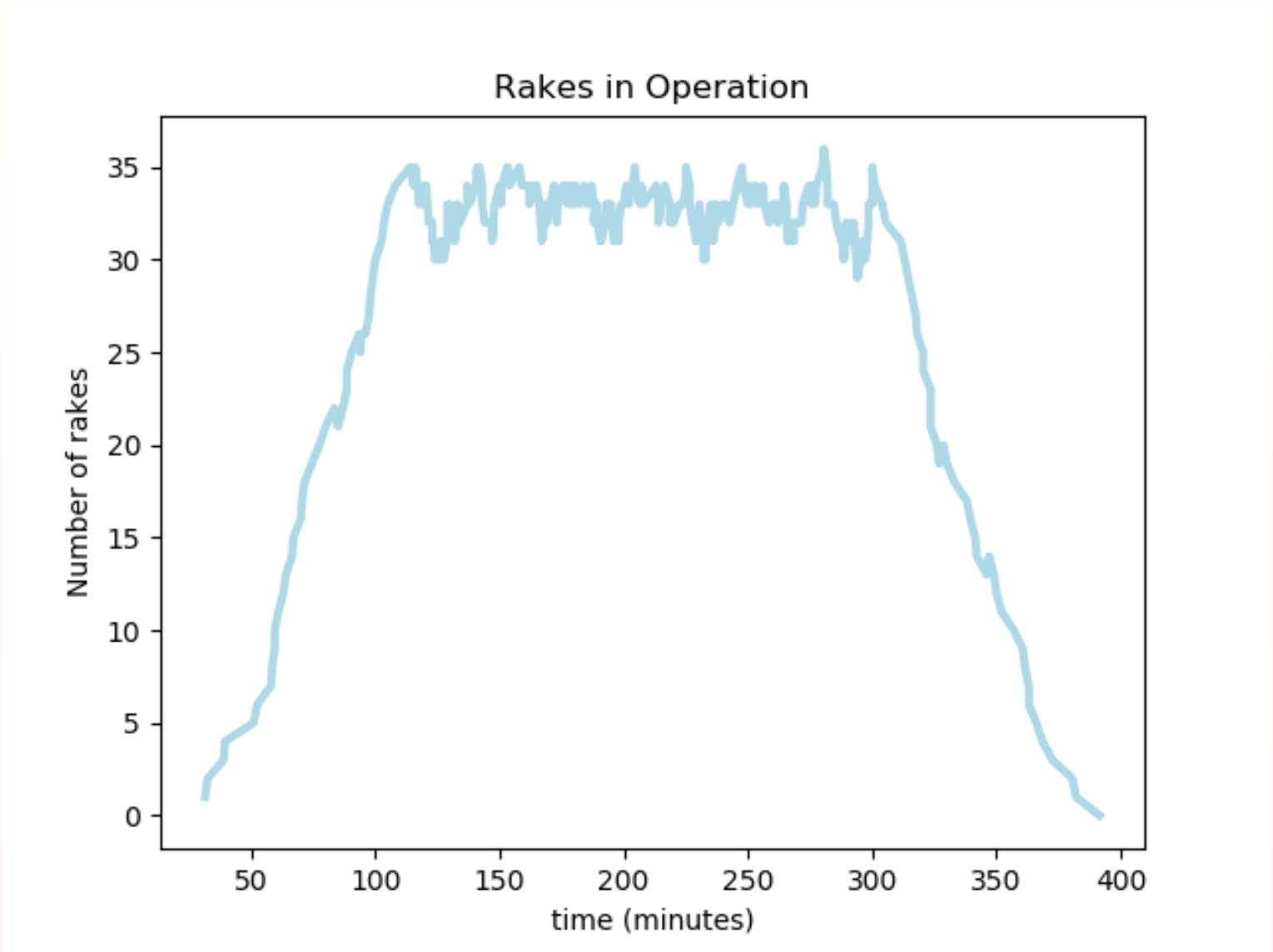
- Modelling: AMPL to model all the constraints
- Solver: Gurobi
- Python & Bash Scripts for pre-processing and post-processing

Trying Pyomo and PuLP to replace AMPL/Gurobi
Status: able to save 3 rakes (out of 45 of the current timetable)

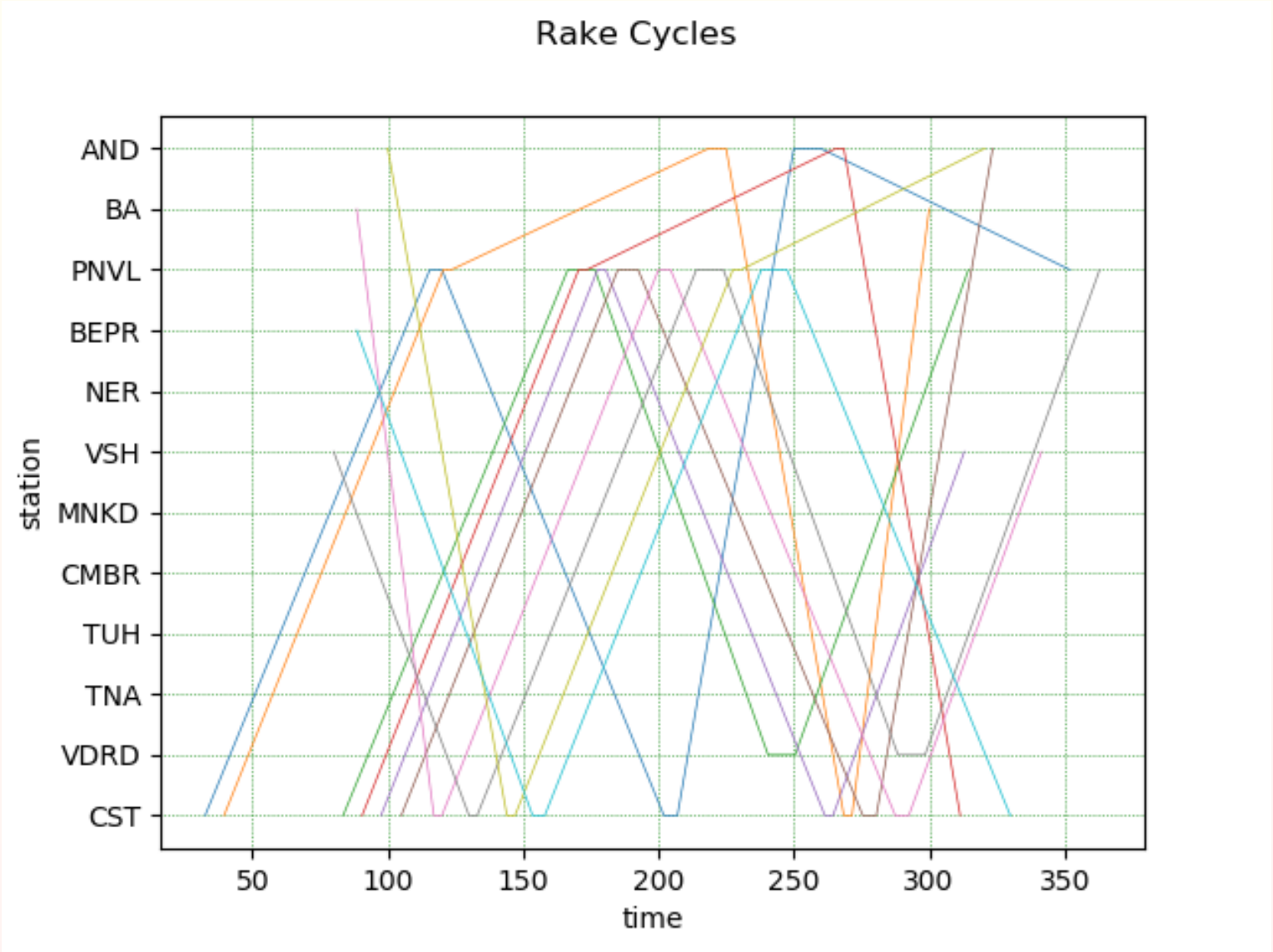
Case Study: Platform Allocation at CST

	ODD-Arrival-dep-events				EVEN-Arrival-dep-events	
Prev-Departure	Next-odd-Arrival	OccupiedFor		Prev-Departure	Next-even-Arrival	OccupiedFor
5:00	5:04	4.0 minutes		5:03	5:07	4.0 minutes
5:07	5:10	3.0 minutes		5:10	5:14	4.0 minutes
5:13	5:18	5.0 minutes		5:17	5:24	7.0 minutes
5:21	5:27	6.0 minutes		5:27	5:33.5	6.5 minutes
5:30	5:39	9.0 minutes		5:38	5:42	4.0 minutes
5:42	5:45	3.0 minutes		5:45	5:49	4.0 minutes
5:48.5	5:53	4.5 minutes		5:52	5:59	7.0 minutes
5:58	6:02	4.0 minutes		6:04	6:08	4.0 minutes
6:07	6:11	4.0 minutes		6:11	6:15	4.0 minutes
6:14	6:18	4.0 minutes		6:18	6:22	4.0 minutes
6:21	6:28	7.0 minutes		6:27	6:34	7.0 minutes
6:33	6:40	7.0 minutes		6:39	6:44	5.0 minutes
6:45	6:48	3.0 minutes		6:48	6:53	5.0 minutes
6:53	6:56	3.0 minutes		6:57	7:01	4.0 minutes

Case Study: Rakes in Operation vs time



Case Study: Rake Cycles vs time



Large data problems: metro/suburban timetabling

- Input: services/frequencies
- Constraints: headway, turn-around, platform occupation

Large data problems: metro/suburban timetabling

- Input: services/frequencies
- Constraints: headway, turn-around, platform occupation & vacating

Large data problems: metro/suburban timetabling

- Input: services/frequencies
- Constraints: headway, turn-around, platform occupation & vacating
- Output: rake-cycles and the timetable

Large data problems: metro/suburban timetabling

- Input: services/frequencies
- Constraints: headway, turn-around, platform occupation & vacating
- Output: rake-cycles and the timetable
- Number of rakes: need to optimize

Large data problems: metro/suburban timetabling

- Input: services/frequencies
- Constraints: headway, turn-around, platform occupation & vacating
- Output: rake-cycles and the timetable
- Number of rakes: need to optimize
- Constructive timetables: rakes/services are incrementally introduced

Large data problems: metro/suburban timetabling

- Input: services/frequencies
- Constraints: headway, turn-around, platform occupation & vacating
- Output: rake-cycles and the timetable
- Number of rakes: need to optimize
- Constructive timetables: rakes/services are incrementally introduced
- Can handle rake-cycles constructively:

Large data problems: metro/suburban timetabling

- Input: services/frequencies
- Constraints: headway, turn-around, platform occupation & vacating
- Output: rake-cycles and the timetable
- Number of rakes: need to optimize
- Constructive timetables: rakes/services are incrementally introduced
- Can handle rake-cycles constructively: still difficult task

Constrained programming based solvers:

Large data problems: metro/suburban timetabling

- Input: services/frequencies
- Constraints: headway, turn-around, platform occupation & vacating
- Output: rake-cycles and the timetable
- Number of rakes: need to optimize
- Constructive timetables: rakes/services are incrementally introduced
- Can handle rake-cycles constructively: still difficult task

Constrained programming based solvers:

- Gets a good feasible solution

Large data problems: metro/suburban timetabling

- Input: services/frequencies
- Constraints: headway, turn-around, platform occupation & vacating
- Output: rake-cycles and the timetable
- Number of rakes: need to optimize
- Constructive timetables: rakes/services are incrementally introduced
- Can handle rake-cycles constructively: still difficult task

Constrained programming based solvers:

- Gets a good feasible solution
- Helps to have spreadsheet based validators/checkers

(Though solvers have ensured satisfaction of constraints, one can validate by introducing 'test-flaws')

Periodic and Aperiodic Versions

Periodic Version:

- Tough integer programming problem: but fewer variables (just one period)
- Needs identical situations at end of every time period
- Very compact description of timetable
- Makes crew-scheduling easier

Aperiodic Version:

- Allows customization of services (peak 3-hour timetable need not have all service-counts as multiple of 3)
- Many more event arrival/departure variables

Have tried both versions in Mumbai suburban, but detailed case study is the aperiodic version

Delhi Metro Train Frequency

Line 1 (Dilshad Garden to Rithala)
- Peak Hours : 3 min
- Lean Hours : Upto 12 min

Line 2 (Jahangirpuri to HUDA City Cent)
- Peak Hours : 3 min
- Lean Hours : Upto 12 min

Line 3 (Noida City Cent to Dwarka Sec. 21)
- Peak Hours : 3 min
- Lean Hours : Upto 12 min

Line 4 (Yamuna Bank - Anand Vihar(ISBT))
- Peak Hours : 6 min.
- Lean Hours : Upto 12 min.

(Central Secretariat - Badarpur)
- Peak Hours : 4 min.
- Lean Hours : Upto 12 min.

(Inderlok - Mundka)
- Frequency : Upto 10 min.

Last trains from Inder lok

Towards Dilshad Garden
23:18 hrs

Towards Rithala
23:27 hrs

Towards Jahangirpuri
23:33 hrs

Towards Huda City Centre
22:51 hrs

Towards Anand Vihar
22:27 hrs

Towards Vishwavidyalaya
23:33 hrs

Towards Noida City Centre
22:39 hrs

Towards Mundka
23:30 hrs

Towards Dwarka
22:51 hrs

Towards Sarita Vihar
22:51 hrs

Rangaraj, Suburban Timetabling 1

Delhi metro timetable: example

Computation tools: example: spreadsheet/excel

- A hand-calculator (or computer's calculator) can perform complex calculations

Computation tools: example: spreadsheet/excel

- A hand-calculator (or computer's calculator) can perform complex calculations
- Insufficient for large repetitive calculations on data
- Need spreadsheet/excel's power

Computation tools: example: spreadsheet/excel

- A hand-calculator (or computer's calculator) can perform complex calculations
- Insufficient for large repetitive calculations on data
- Need spreadsheet/excel's power
  =SUM, =IF, =SUMIF, complex formula, etc.
- Can use 'VLOOKUP' to quickly merge data across sheets

Computation tools: example: spreadsheet/excel

- A hand-calculator (or computer's calculator) can perform complex calculations
- Insufficient for large repetitive calculations on data
- Need spreadsheet/excel's power
  =SUM, =IF, =SUMIF, complex formula, etc.
- Can use 'VLOOKUP' to quickly merge data across sheets
- "Advanced and mature usage of spreadsheets for large data: pivot-tables": data analysis (summary)

Computation tools: example: spreadsheet/excel

- A hand-calculator (or computer's calculator) can perform complex calculations
- Insufficient for large repetitive calculations on data
- Need spreadsheet/excel's power
  =SUM, =IF, =SUMIF, complex formula, etc.
- Can use 'VLOOKUP' to quickly merge data across sheets
- "Advanced and mature usage of spreadsheets for large data: pivot-tables": data analysis (summary)
- However, spreadsheet:

Computation tools: example: spreadsheet/excel

- A hand-calculator (or computer's calculator) can perform complex calculations
- Insufficient for large repetitive calculations on data
- Need spreadsheet/excel's power
  =SUM, =IF, =SUMIF, complex formula, etc.
- Can use 'VLOOKUP' to quickly merge data across sheets
- "Advanced and mature usage of spreadsheets for large data: pivot-tables": data analysis (summary)
- However, spreadsheet:
  - insufficient for complex allocations/constraint validation
  - has limited ability as a 'solver'
  - no 'while' loop, no jumping from one solution to another
  - not OK for automation, nor for large data
  - semi-automatic, at best

Large data problems: crew allotment

- Complex safety-based rules: Hours of Employment and Period of Rest Rules (HOER)

Large data problems: crew allotment

- Complex safety-based rules: Hours of Employment and Period of Rest Rules (HOER)
- Need to utilize crew members efficiently

Large data problems: crew allotment

- Complex safety-based rules: Hours of Employment and Period of Rest Rules (HOER)
- Need to utilize crew members efficiently
- Efficient utilization of crew: more buffer for:

Large data problems: crew allotment

- Complex safety-based rules: Hours of Employment and Period of Rest Rules (HOER)
- Need to utilize crew members efficiently
- Efficient utilization of crew: more buffer for:
  - robustness

Large data problems: crew allotment

- Complex safety-based rules: Hours of Employment and Period of Rest Rules (HOER)
- Need to utilize crew members efficiently
- Efficient utilization of crew: more buffer for:
  - robustness
  - can allow liberal leave policies

Large data problems: crew allotment

- Complex safety-based rules: Hours of Employment and Period of Rest Rules (HOER)
- Need to utilize crew members efficiently
- Efficient utilization of crew: more buffer for:
  - robustness
  - can allow liberal leave policies
- WR suburban services: $\sim$1400 services:

Large data problems: crew allotment

- Complex safety-based rules: Hours of Employment and Period of Rest Rules (HOER)
- Need to utilize crew members efficiently
- Efficient utilization of crew: more buffer for:
  - robustness
  - can allow liberal leave policies
- WR suburban services: $\sim$1400 services: to manage in $\sim$380 crew members

Large data problems: crew allotment

- Complex safety-based rules: Hours of Employment and Period of Rest Rules (HOER)
- Need to utilize crew members efficiently
- Efficient utilization of crew: more buffer for:
  - robustness
  - can allow liberal leave policies
- WR suburban services: $\sim$1400 services: to manage in $\sim$380 crew members
- Tool helps to:
  - tweak/modify and re-run: takes a minute for each program-run
  - add/delete services and re-run program
  - redecide/change lobby locations and check

Large data problems: crew allotment

- Complex safety-based rules: Hours of Employment and Period of Rest Rules (HOER)
- Need to utilize crew members efficiently
- Efficient utilization of crew: more buffer for:
  - robustness
  - can allow liberal leave policies
- WR suburban services: $\sim$1400 services: to manage in $\sim$380 crew members
- Tool helps to:
  - tweak/modify and re-run: takes a minute for each program-run
  - add/delete services and re-run program
  - redecide/change lobby locations and check
  - optimize required number of lobby locations

Tools: constructive/solvers/simulation/semi-automate

In my understanding

Tools: constructive/solvers/simulation/semi-automate

In my understanding

- Constructive allocation tools: ensure constraints are satisfied:

Tools: constructive/solvers/simulation/semi-automate

In my understanding

- Constructive allocation tools: ensure constraints are satisfied:
- Allocation based on 'solvers': satisfy all constraints (feasible solutions),

Tools: constructive/solvers/simulation/semi-automate

In my understanding

- Constructive allocation tools: ensure constraints are satisfied:
- Allocation based on 'solvers': satisfy all constraints (feasible solutions), search for 'best' feasible solution

Tools: constructive/solvers/simulation/semi-automate

In my understanding

- Constructive allocation tools: ensure constraints are satisfied:
- Allocation based on 'solvers': satisfy all constraints (feasible solutions), search for 'best' feasible solution
- Validation/checker tools: easily checked in spreadsheet

Tools: constructive/solvers/simulation/semi-automate

In my understanding

- Constructive allocation tools: ensure constraints are satisfied:
- Allocation based on 'solvers': satisfy all constraints (feasible solutions), search for 'best' feasible solution
- Validation/checker tools: easily checked in spreadsheet
- Simulation: for complex scenarios/rules/constraints
- Analysis: tool extracts statistics from simulation or other data
- Visualization tools: to ratify data

Tools: constructive/solvers/simulation/semi-automate

In my understanding

- Constructive allocation tools: ensure constraints are satisfied:
- Allocation based on 'solvers': satisfy all constraints (feasible solutions), search for 'best' feasible solution
- Validation/checker tools: easily checked in spreadsheet
- Simulation: for complex scenarios/rules/constraints
- Analysis: tool extracts statistics from simulation or other data
- Visualization tools: to ratify data
  and also view analysis-based statistics,

Tools: constructive/solvers/simulation/semi-automate

In my understanding

- Constructive allocation tools: ensure constraints are satisfied:
- Allocation based on 'solvers': satisfy all constraints (feasible solutions), search for 'best' feasible solution
- Validation/checker tools: easily checked in spreadsheet
- Simulation: for complex scenarios/rules/constraints
- Analysis: tool extracts statistics from simulation or other data
- Visualization tools: to ratify data
  and also view analysis-based statistics, aid further tool-usage

Tools: constructive/solvers/simulation/semi-automate

In my understanding

- Constructive allocation tools: ensure constraints are satisfied:
- Allocation based on 'solvers': satisfy all constraints (feasible solutions), search for 'best' feasible solution
- Validation/checker tools: easily checked in spreadsheet
- Simulation: for complex scenarios/rules/constraints
- Analysis: tool extracts statistics from simulation or other data
- Visualization tools: to ratify data
  and also view analysis-based statistics, aid further tool-usage
- Semi-automate: provide portal/interface to simplify complex problems

Tools: constructive/solvers/simulation/semi-automate

In my understanding

- Constructive allocation tools: ensure constraints are satisfied:
- Allocation based on 'solvers': satisfy all constraints (feasible solutions), search for 'best' feasible solution
- Validation/checker tools: easily checked in spreadsheet
- Simulation: for complex scenarios/rules/constraints
- Analysis: tool extracts statistics from simulation or other data
- Visualization tools: to ratify data
  and also view analysis-based statistics, aid further tool-usage
- Semi-automate: provide portal/interface to simplify complex problems
  portal takes inputs manually,

Tools: constructive/solvers/simulation/semi-automate

In my understanding

- Constructive allocation tools: ensure constraints are satisfied:
- Allocation based on 'solvers': satisfy all constraints (feasible solutions), search for 'best' feasible solution
- Validation/checker tools: easily checked in spreadsheet
- Simulation: for complex scenarios/rules/constraints
- Analysis: tool extracts statistics from simulation or other data
- Visualization tools: to ratify data
  and also view analysis-based statistics, aid further tool-usage
- Semi-automate: provide portal/interface to simplify complex problems
  portal takes inputs manually,
  after showing statistics, awaits further inputs
  Decision-support tools: can be visual

Machine Learning/Artificial-intelligence

Data (digitally) available now for ‘pattern-searching’: can use ML and AI

Machine Learning/Artificial-intelligence

Data (digitally) available now for 'pattern-searching': can use ML and AI
Only humans (natural intelligence) can make use of these AI tools

- Supervised learning: needs training data, validation data, and then "deploy" to find

Machine Learning/Artificial-intelligence

Data (digitally) available now for 'pattern-searching': can use ML and AI
Only humans (natural intelligence) can make use of these AI tools

- Supervised learning: needs training data, validation data, and then "deploy" to find more/better patterns, adapt and predict
  - Used to extrapolate/predict/fit new data into pre-calculated models
  - Find outliers (in timetabling?) too fast/too slow?

Machine Learning/Artificial-intelligence

Data (digitally) available now for 'pattern-searching': can use ML and AI
Only humans (natural intelligence) can make use of these AI tools

- Supervised learning: needs training data, validation data, and then "deploy" to find more/better patterns, adapt and predict
  - Used to extrapolate/predict/fit new data into pre-calculated models
  - Find outliers (in timetabling?) too fast/too slow?
- Unsupervised learning: no initial training: run heuristics to get good solution

Machine Learning/Artificial-intelligence

Data (digitally) available now for 'pattern-searching': can use ML and AI
Only humans (natural intelligence) can make use of these AI tools

- Supervised learning: needs training data, validation data, and then "deploy" to find more/better patterns, adapt and predict
  - Used to extrapolate/predict/fit new data into pre-calculated models
  - Find outliers (in timetabling?) too fast/too slow?
- Unsupervised learning: no initial training: run heuristics to get good solution
  - grouping of non-daily trains into same/almost-same path

Machine Learning/Artificial-intelligence

Data (digitally) available now for 'pattern-searching': can use ML and AI
Only humans (natural intelligence) can make use of these AI tools

- Supervised learning: needs training data, validation data, and then "deploy" to find more/better patterns, adapt and predict
  - Used to extrapolate/predict/fit new data into pre-calculated models
  - Find outliers (in timetabling?) too fast/too slow?
- Unsupervised learning: no initial training: run heuristics to get good solution
  - grouping of non-daily trains into same/almost-same path
- Reinforcement learning: specify a 'reward function' on each solution, and iterate/jump from a solution to another

Machine Learning/Artificial-intelligence

Data (digitally) available now for 'pattern-searching': can use ML and AI
Only humans (natural intelligence) can make use of these AI tools

- Supervised learning: needs training data, validation data, and then "deploy" to find more/better patterns, adapt and predict
  - Used to extrapolate/predict/fit new data into pre-calculated models
  - Find outliers (in timetabling?) too fast/too slow?
- Unsupervised learning: no initial training: run heuristics to get good solution
  - grouping of non-daily trains into same/almost-same path
- Reinforcement learning: specify a 'reward function' on each solution, and iterate/jump from a solution to another
  often computationally easy to jump (rather than exhaustively search)

Machine Learning/Artificial-intelligence

Data (digitally) available now for 'pattern-searching': can use ML and AI
Only humans (natural intelligence) can make use of these AI tools

- Supervised learning: needs training data, validation data, and then "deploy" to find more/better patterns, adapt and predict
  - Used to extrapolate/predict/fit new data into pre-calculated models
  - Find outliers (in timetabling?) too fast/too slow?
- Unsupervised learning: no initial training: run heuristics to get good solution
  - grouping of non-daily trains into same/almost-same path
- Reinforcement learning: specify a 'reward function' on each solution, and iterate/jump from a solution to another
  often computationally easy to jump (rather than exhaustively search)
  - optimize crew allotment
  - choose start-timings to get 'better and better' grouping at congested section

Recent railway projects completed/ongoing

- ZBTT (ongoing): only GQD routes (daily-paths) and freight trains
- CR/WR locals: security personnel assignment for late-night travel in ladies-compartments (9pm to 6am): 2017

Recent railway projects completed/ongoing

- ZBTT (ongoing): only GQD routes (daily-paths) and freight trains
- CR/WR locals: security personnel assignment for late-night travel in ladies-compartments (9pm to 6am): 2017
- RDSO: section simulator: 2018
- Use of section simulator for ALD study: for RITES Ltd: 2018
- Use of section simulator for Niti-Aayog: short study: 2017
- Ahmedabad junction: simulation based congestion study: 2019
- WR/CR: crew allotment and harbour line timetable: 2019 (but ....)
- GQD: (long-distance) timetabling: ZBTT (in 2020)
- Several other usages of this simulator: before 2016: kept in the link

Recent railway projects completed/ongoing

- ZBTT (ongoing): only GQD routes (daily-paths) and freight trains
- CR/WR locals: security personnel assignment for late-night travel in ladies-compartments (9pm to 6am): 2017
- RDSO: section simulator: 2018
- Use of section simulator for ALD study: for RITES Ltd: 2018
- Use of section simulator for Niti-Aayog: short study: 2017
- Ahmedabad junction: simulation based congestion study: 2019
- WR/CR: crew allotment and harbour line timetable: 2019 (but ....)
- GQD: (long-distance) timetabling: ZBTT (in 2020)
- Several other usages of this simulator: before 2016: kept in the link

Our other projects involving development of tools

Together with collaborators, supervised/developed tools for:

- Fault diagnosis in nuclear reactor sensors (140 sensors, complex dynamics)
  BRNS project: implementation in Matlab/Scilab,

Our other projects involving development of tools

Together with collaborators, supervised/developed tools for:

- Fault diagnosis in nuclear reactor sensors (140 sensors, complex dynamics)
  BRNS project: implementation in Matlab/Scilab, in 2nd phase: Python

Our other projects involving development of tools

Together with collaborators, supervised/developed tools for:

- Fault diagnosis in nuclear reactor sensors (140 sensors, complex dynamics)
  BRNS project: implementation in Matlab/Scilab, in 2nd phase: Python
- Joint Admission for M.Sc.: allocation program (ties/reservation/multiple-papers/multiple-rounds/transparency):
  2300 M.Sc. seats across IITs (amongst $\sim$6k JAM-qualified candidates)

Our other projects involving development of tools

Together with collaborators, supervised/developed tools for:

- Fault diagnosis in nuclear reactor sensors (140 sensors, complex dynamics)
  BRNS project: implementation in Matlab/Scilab, in 2nd phase: Python
- Joint Admission for M.Sc.: allocation program (ties/reservation/multiple-papers/multiple-rounds/transparency):
  2300 M.Sc. seats across IITs (amongst $\sim$6k JAM-qualified candidates)
- Final project/position allotment for MTech/PhD candidates: EE, IITB

Our other projects involving development of tools

Together with collaborators, supervised/developed tools for:

- Fault diagnosis in nuclear reactor sensors (140 sensors, complex dynamics)
  BRNS project: implementation in Matlab/Scilab, in 2nd phase: Python
- Joint Admission for M.Sc.: allocation program (ties/reservation/multiple-papers/multiple-rounds/transparency):
  2300 M.Sc. seats across IITs (amongst $\sim$6k JAM-qualified candidates)
- Final project/position allotment for MTech/PhD candidates: EE, IITB
- Admission shortlisting portal: semi-automation (in EE, IITB)

Our other projects involving development of tools

Together with collaborators, supervised/developed tools for:

- Fault diagnosis in nuclear reactor sensors (140 sensors, complex dynamics)
  BRNS project: implementation in Matlab/Scilab, in 2nd phase: Python
- Joint Admission for M.Sc.: allocation program (ties/reservation/multiple-papers/multiple-rounds/transparency):
  2300 M.Sc. seats across IITs (amongst $\sim$6k JAM-qualified candidates)
- Final project/position allotment for MTech/PhD candidates: EE, IITB
- Admission shortlisting portal: semi-automation (in EE, IITB)
  About 400 candidates (and parents) can leave a day early (to home/other-IITs)

Our other projects involving development of tools

Together with collaborators, supervised/developed tools for:

- Fault diagnosis in nuclear reactor sensors (140 sensors, complex dynamics)
  BRNS project: implementation in Matlab/Scilab, in 2nd phase: Python
- Joint Admission for M.Sc.: allocation program (ties/reservation/multiple-papers/multiple-rounds/transparency):
  2300 M.Sc. seats across IITs (amongst $\sim$6k JAM-qualified candidates)
- Final project/position allotment for MTech/PhD candidates: EE, IITB
- Admission shortlisting portal: semi-automation (in EE, IITB)
  About 400 candidates (and parents) can leave a day early (to home/other-IITs)
- Interview committee allotment (in EE, IITB): reduce from 4 hrs to 1 min
  About 150 candidates (and parents) can leave a day early (to home/other-IITs)

Our other projects involving development of tools

Together with collaborators, supervised/developed tools for:

- Fault diagnosis in nuclear reactor sensors (140 sensors, complex dynamics)
  BRNS project: implementation in Matlab/Scilab, in 2nd phase: Python
- Joint Admission for M.Sc.: allocation program (ties/reservation/multiple-papers/multiple-rounds/transparency):
  2300 M.Sc. seats across IITs (amongst $\sim$6k JAM-qualified candidates)
- Final project/position allotment for MTech/PhD candidates: EE, IITB
- Admission shortlisting portal: semi-automation (in EE, IITB)
  About 400 candidates (and parents) can leave a day early (to home/other-IITs)
- Interview committee allotment (in EE, IITB): reduce from 4 hrs to 1 min
  About 150 candidates (and parents) can leave a day early (to home/other-IITs)
- PhD-students portal: database/scheduling: semi-automation (in EE, IITB)

Machinery/tools

- Java and Python: free and open source
- Free and Open-Source (FOSS): ‘independence’

Machinery/tools

- Java and Python: free and open source
- Free and Open-Source (FOSS): ‘independence’
- Helps to use open-source tools: else exorbitant (non-academic) prices

Machinery/tools

- Java and Python: free and open source
- Free and Open-Source (FOSS): ‘independence’
- Helps to use open-source tools: else exorbitant (non-academic) prices (and strings on usage)

Machinery/tools

- Java and Python: free and open source
- Free and Open-Source (FOSS): ‘independence’
- Helps to use open-source tools: else exorbitant (non-academic) prices (and strings on usage)
- Gurobi as the solver: Gurobi’s limited version is free
- IPOPT/COIN-OR (open-source and state of the art solvers): will shift soon to this

Machinery/tools

- Java and Python: free and open source
- Free and Open-Source (FOSS): ‘independence’
- Helps to use open-source tools: else exorbitant (non-academic) prices (and strings on usage)
- Gurobi as the solver: Gurobi’s limited version is free
- IPOPT/COIN-OR (open-source and state of the art solvers): will shift soon to this
- Commercial software: dependence: cannot install on many computers

Machinery/tools

- Java and Python: free and open source
- Free and Open-Source (FOSS): ‘independence’
- Helps to use open-source tools: else exorbitant (non-academic) prices (and strings on usage)
- Gurobi as the solver: Gurobi’s limited version is free
- IPOPT/COIN-OR (open-source and state of the art solvers): will shift soon to this
- Commercial software: dependence: cannot install on many computers
- Software made in FOSS can be modified to yield valuable statistics for analysis

Ready-made/proprietary vs ‘home-grown’ software

- Our metro-rail operations: complex and unique challenges
- Other metro/suburban-railways elsewhere: have their own challenges
- Each software needs significant local customization

Ready-made/proprietary vs ‘home-grown’ software

- Our metro-rail operations: complex and unique challenges
- Other metro/suburban-railways elsewhere: have their own challenges
- Each software needs significant local customization
- For complex requirements: prudent to have our own slowly-grown ‘home-grown’ software

Ready-made/proprietary vs ‘home-grown’ software

- Our metro-rail operations: complex and unique challenges
- Other metro/suburban-railways elsewhere: have their own challenges
- Each software needs significant local customization
- For complex requirements: prudent to have our own slowly-grown ‘home-grown’ software
- Problem specifications: from our own ‘shop-floor’: operations personnel
- Railways/Academia/Software-agency: combination inevitable

Zero-base TT efforts elsewhere: example

The New Dutch Timetable: The OR Revolution, 2009 paper in *Interfaces*

Paper abstract

In December 2006, Netherlands Railways introduced a completely new timetable. Its objective was to facilitate the growth of passenger and freight transport on a highly utilized railway network, and improve the robustness of the timetable resulting in less train delays in the operation. Further adjusting the existing timetable constructed in 1970 was not option anymore, because further growth would then require significant investments in the rail infrastructure.

Constructing a railway timetable from scratch for about 5,500 daily trains was a complex problem. To support this process, we generated several timetables using sophisticated operations research techniques, and finally selected and implemented one of these timetables. Furthermore, because rolling-stock and crew costs are principal components of the cost of a passenger railway operator, we used innovative operations research tools to devise efficient schedules for these two resources.

The new resource schedules and the increased number of passengers resulted

Authors/affiliations (of the famous paper in Interfaces)

The New Dutch Timetable: The OR Revolution, 2009 paper in *Interfaces*
Authors: Kroon, Huisman, Abbink, Fioole, Ybema, Maroti, Schrijver, Steenbeek, Fischetti
Affiliations: railway personnel/software-firm/academia

- Department of Logistics, Netherlands Railways: Kroon, Huisman, Abbink, Fioole, Ybema
- Rotterdam School of Management, Erasmus University: Kroon, Maroti
- Econometric Institute, Erasmus University Rotterdam: Huisman
- CWI and University of Amsterdam: Schrijver
- Safiro Software Solutions: Steenbeek
- University of Padova Italy: Fischetti

Authors/affiliations (of the famous paper in Interfaces)

The New Dutch Timetable: The OR Revolution, 2009 paper in *Interfaces*
Authors: Kroon, Huisman, Abbink, Fioole, Ybema, Maroti, Schrijver, Steenbeek, Fischetti
Affiliations: railway personnel/software-firm/academia

- Department of Logistics, Netherlands Railways: Kroon, Huisman, Abbink, Fioole, Ybema
- Rotterdam School of Management, Erasmus University: Kroon, Maroti
- Econometric Institute, Erasmus University Rotterdam: Huisman
- CWI and University of Amsterdam: Schrijver
- Safiro Software Solutions: Steenbeek
- University of Padova Italy: Fischetti

Similar efforts in Germany: Narayan Rangaraj's collaborators

Summary: need to shift to automated/semi-automated tools

- Need to shift to modern tools for
  - self-growth (and ourselves remaining relevant over next few decades)
  - system productivity/efficiency
- Tools that are 'home-grown' and in FOSS allow complete flexibility/independence and customization
- Describing/formulating the specs of the tool:
  - from on-field/operations personnel

Summary: need to shift to automated/semi-automated tools

- Need to shift to modern tools for
  - self-growth (and ourselves remaining relevant over next few decades)
  - system productivity/efficiency
- Tools that are 'home-grown' and in FOSS allow complete flexibility/independence and customization
- Describing/formulating the specs of the tool:
  - from on-field/operations personnel (primarily)

Summary: need to shift to automated/semi-automated tools

- Need to shift to modern tools for
  - self-growth (and ourselves remaining relevant over next few decades)
  - system productivity/efficiency
- Tools that are 'home-grown' and in FOSS allow complete flexibility/independence and customization
- Describing/formulating the specs of the tool:
  - from on-field/operations personnel (primarily)
  - from anybody interested in the area

Contact details

Thanks to Karthik, Husna, Samay: from EE/IEOR: IITB team
Thanks to CEP office, IITB: Mr. Surendra Gaikwad
Thanks to I-Metro: Ms. Manjari Srivastava, Mr. Sandeep Sharma

Contact details

Thanks to Karthik, Husna, Samay: from EE/IEOR: IITB team
Thanks to CEP office, IITB: Mr. Surendra Gaikwad
Thanks to I-Metro: Ms. Manjari Srivastava, Mr. Sandeep Sharma

- Madhu Belur: 99874 66 279, `belur@iitb.ac.in`
- This talk will soon be available at
  `www.ee.iitb.ac.in/%7Ebelur/talks`
  `www.ee.iitb.ac.in/%7Ebelur/railways/workshop`
  `www.ee.iitb.ac.in/%7Ebelur/railways`
- Narayan Rangaraj
  `narayan.rangaraj@iitb.ac.in`