Improved Local Freight Train Classification

Jens Maue\textsuperscript{1}

Joint Work with Peter Márton\textsuperscript{2} and Marc Nunkesser\textsuperscript{1}

\textsuperscript{1}Institute of Theoretical Computer Science, ETH Zürich, Switzerland

\textsuperscript{2}Department of Transportation Networks, University of Žilina, Slovakia

RailZurich 2009 - 12 February 2009

\textsuperscript{1}Partially supported by the Future and Emerging Technologies Unit of EC (IST priority - 6th FP) under contract no. FP6-021235-2 (project ARRIVAL).

\textsuperscript{2}Partially supported by the Slovak grant foundation under grant no. 1-4057-07 (project “Agent Oriented Models of Service Systems”).
Local freight train
  - multi-destination freight train
  - cars ordered by destinations

Train classification
  - special sorting problem
  - classification yard
Outline

Train Classification in General

Classification Schedules

IP Formulation
  Basic Model
  Real-World Instance
  Real-World Restrictions

Concluding Remarks
Train Classification in General

Example Classification Process

Train Classification

- **goal**: ascendingly **ordered train** on $\theta_{\text{out}}$
- use available tracks $\theta_1$, $\theta_2$, and $\theta_3$
Example Classification Process

Train Classification

- **goal**: ascendingly ordered train on $\theta_{out}$
- use available tracks $\theta_1$, $\theta_2$, and $\theta_3$
Train Classification

- **goal**: ascendingly ordered train on $\theta_{\text{out}}$
- use available tracks $\theta_1$, $\theta_2$, and $\theta_3$
Example Classification Process

Train Classification

- **goal**: ascendingly *ordered train* on $\theta_{\text{out}}$
- use available tracks $\theta_1$, $\theta_2$, and $\theta_3$
Train Classification

- **goal:** ascendingly ordered train on $\theta_{out}$
- use available tracks $\theta_1$, $\theta_2$, and $\theta_3$
Train Classification in General

Example Classification Process

Train Classification
- **goal**: ascendingly ordered train on $\theta_{\text{out}}$
- use available tracks $\theta_1$, $\theta_2$, and $\theta_3$
Train Classification

- **goal**: ascendingly ordered train on $\theta_{out}$
- use available tracks $\theta_1$, $\theta_2$, and $\theta_3$
Train Classification

- **goal**: ascendingly ordered train on $\theta_{\text{out}}$
- use available tracks $\theta_1$, $\theta_2$, and $\theta_3$
Example Classification Process

Train Classification

- **goal**: ascendingly ordered train on $\theta_{\text{out}}$
- use available tracks $\theta_1$, $\theta_2$, and $\theta_3$
Train Classification in General

Example Classification Process

Train Classification

▶ goal: ascendingly ordered train on $\theta_{out}$

▶ use available tracks $\theta_1$, $\theta_2$, and $\theta_3$
Train Classification

- **goal**: ascendingly ordered train on $\theta_{\text{out}}$
- use available tracks $\theta_1$, $\theta_2$, and $\theta_3$
Example Classification Process

Train Classification

- **goal:** ascendingly ordered train on $\theta_{\text{out}}$
- use available tracks $\theta_1$, $\theta_2$, and $\theta_3$
Example Classification Process

Train Classification

- **goal:** ascendingly ordered train on $\theta_{\text{out}}$
- use available tracks $\theta_1$, $\theta_2$, and $\theta_3$
Train Classification

- **goal:** ascendingly ordered train on $\theta_{\text{out}}$
- use available tracks $\theta_1$, $\theta_2$, and $\theta_3$
Train Classification

- **goal**: ascendingly ordered train on $\theta_{\text{out}}$
- use available tracks $\theta_1$, $\theta_2$, and $\theta_3$
Train Classification

- **goal**: ascendingly ordered train on $\theta_{\text{out}}$
- use available tracks $\theta_1$, $\theta_2$, and $\theta_3$

Classification Process

1. initially roll-in input train
2. alternately pull out and roll in
3. finish with ordered train
Train Classification

- **goal**: ascendingly ordered train on $\theta_{\text{out}}$
- use available tracks $\theta_1$, $\theta_2$, and $\theta_3$

Classification Process

1. initially roll-in input train
2. alternately pull out and roll in
3. finish with ordered train

Objective: number $h$ of pull-out steps
Schedule Encoding [JMMN07]
Schedule Encoding [JMMN07]

Schedule representation

- **assignment** of cars to bitstrings of length $h$
- rows: bitstring $b^j$ encodes journey of $j$th car
- columns: bits encode sequence of pull-out steps
- bit $b^j_i = 1$ iff $j$th car visits track pulled in $i$th step
Schedule Encoding [JMMN07]

Schedule representation
- assignment of cars to bitstrings of length \( h \)
- rows: bitstring \( b^j \) encodes journey of \( j \)th car
- columns: bits encode sequence of pull-out steps
- bit \( b^j_i = 1 \) iff \( j \)th car visits track pulled in \( i \)th step

Schedule derivation: two consecutive cars \( \tau \) and \( \tau + 1 \)
- correct order: assign same bitstring
- reversed order: assign bigger bitstring to \( \tau + 1 \)
Schedule Encoding [JMMN07]

Schedule representation

- assignment of cars to bitstrings of length $h$
- rows: bitstring $b^j$ encodes journey of $j$th car
- columns: bits encode sequence of pull-out steps
- bit $b^j_i = 1$ iff $j$th car visits track pulled in $i$th step

Schedule derivation: two consecutive cars $\tau$ and $\tau + 1$

- correct order: assign same bitstring
- reversed order: assign bigger bitstring to $\tau + 1$
- objective: length $h$ of schedule
Basic IP Model [MN09]

\[
\begin{align*}
\text{min} & \quad \sum_{1 \leq i \leq n, \ 0 \leq j < h} b^j_i \\
\text{s.t.} & \quad \sum_{0 \leq j < h} 2^i b^j_i \geq \text{rev}(j-1, j) + \sum_{0 \leq j < h} 2^i b^{j-1}_i \quad \forall j \in \{1, \ldots, n\} \setminus F \quad (1) \\
& \quad \sum_{1 \leq i \leq n} b^j_i \leq C \quad \forall i \in \{0, \ldots, h-1\} \quad (2) \\
& \quad b^j_i \in \{0, 1\} \quad \forall j \in \{1, \ldots, n\}, \ \forall i \in \{0, \ldots, h-1\} \quad (3)
\end{align*}
\]

- \text{rev}(j-1, j) = 1 \text{ iff cars } j-1 \text{ and } j \text{ in reversed order in incoming train}
- \text{F subset of cars that are first in their respective outgoing train}
- \text{classification tracks have capacity } C
Traffic data
- single day in 2005
- volume 328 cars
- 23 outgoing trains

Infrastructure and operation
- two parallel humps
- local freight trains: collect on ten tracks
- time window for pull-out steps
- further tracks for outgoing train formation
Extended IP Model [MN09]

Additional constraints for Lausanne-Triage
- initial roll-in restricted to ten tracks
- assignment of outgoing trains to either hump
- respect departure times

Resulting schedule
- one step shorter
- one track less required

Verification by computer simulation in progress (not finished yet)
Additional constraints for Lausanne-Triage

- initial roll-in restricted to ten tracks
- assignment of outgoing trains to either hump
- respect departure times

Resulting schedule

- one step shorter
- one track less required
- verification by computer simulation in progress (not finished yet)
Concluding Remarks

Conclusion

- encoding yields flexible IP model
- adapts to various real-world restrictions
- Lausanne-Triage: save one step and track
Concluding Remarks

Conclusion

- encoding yields flexible IP model
- adapts to various real-world restrictions
- Lausanne-Triage: save one step and track

Ongoing work

- computer simulation for Lausanne-Triage (Villon)
- evaluation of 2-approximation
- time-dependent input
- robustness questions