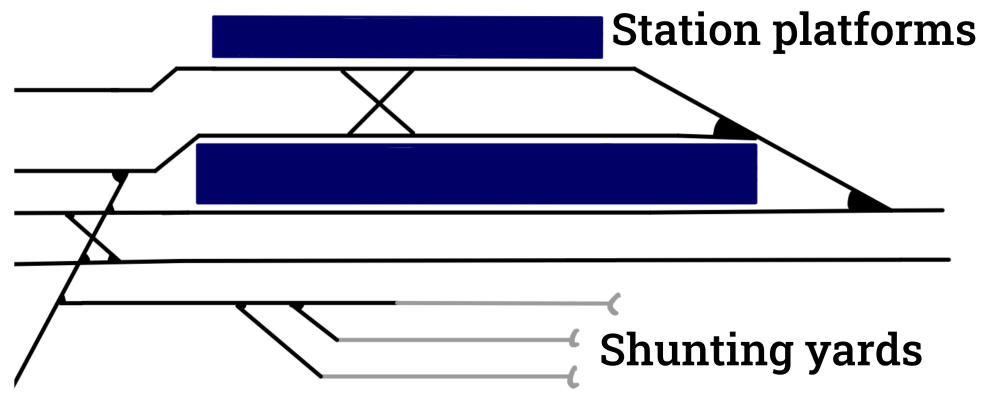
Moving Trains like Pebbles: A Feasibility Study on Tree Yards

July 2023

Issa K. Hanou, Jesse Mulderij, Mathijs M. de Weerdt

INTRODUCTION

- **Train Unit Shunting Problem** (NP-hard)[1]: parking trains at shunting yards during the night. Problem of matching incoming/outcoming trains, parking and routing; with the option of adding servicing actions
- **Pebble Motion**[2]: given a graph and set of pebbles (items), how to move from start to goal locations?
  - Polynomial solving time for tree graphs, though long, impractical solution



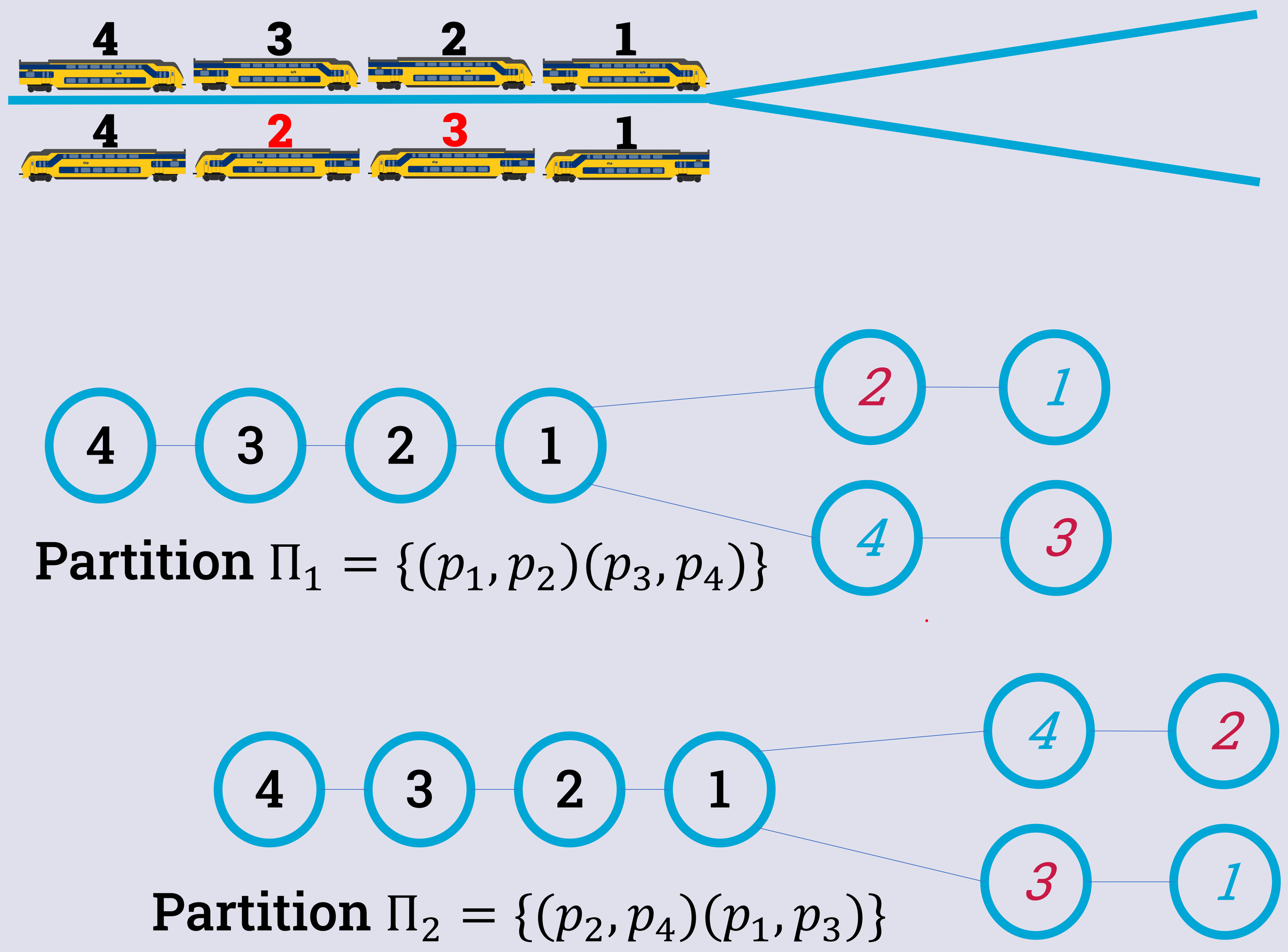
Shunting yard layout

[1] Freling, R.; Lentink, R. M.; Kroon, L. G.; and Huisman, D. 2005. Shunting of passenger train units in a railway station. Transportation Science, 39(2): 261–272.
[2] Auletta, V.; Monti, A.; Parente, M.; and Persiano, P. 1999. A Linear-Time Algorithm for the Feasibility of Pebble Motion on Trees. Algorithmica, 23: 223–245.

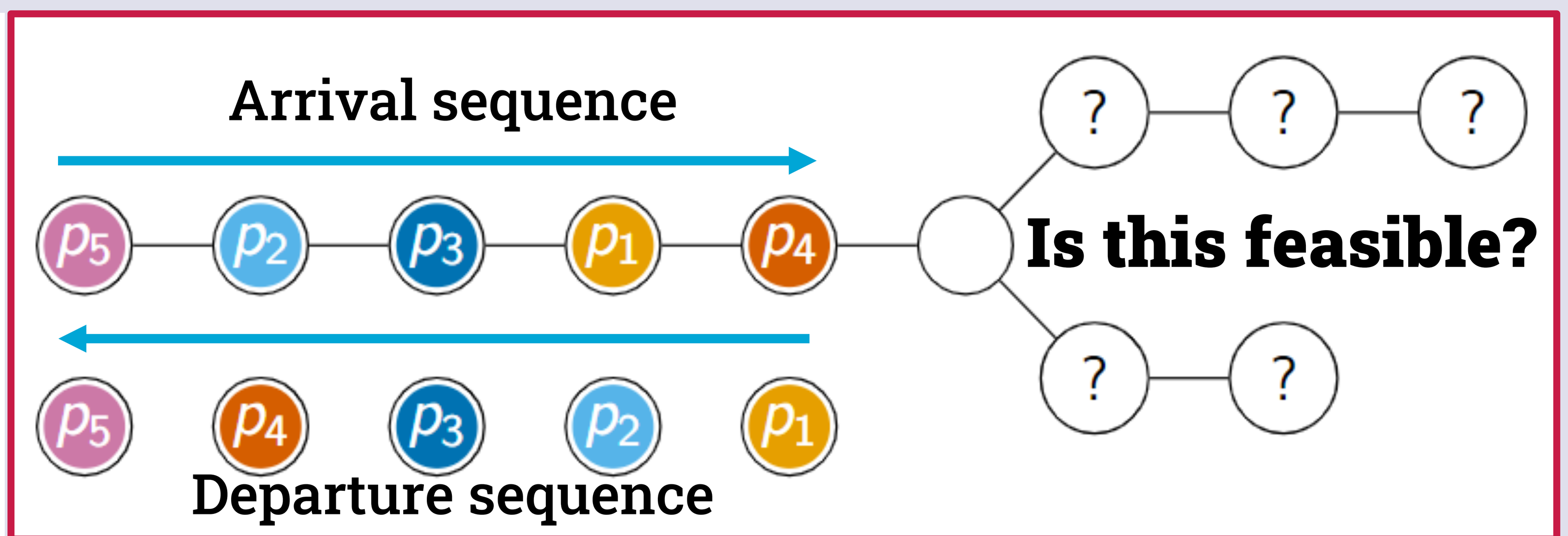
plans

PROBLEM

- In tree graphs (like one-way in/out shunting yards): given arrival sequence of pebbles, and departing sequence, find sequence of moves to park all pebbles, then depart all pebbles, without using intermediate parking locations.
- **Partition**: a set of totally ordered sets of pebbles that can be parked together on a branch

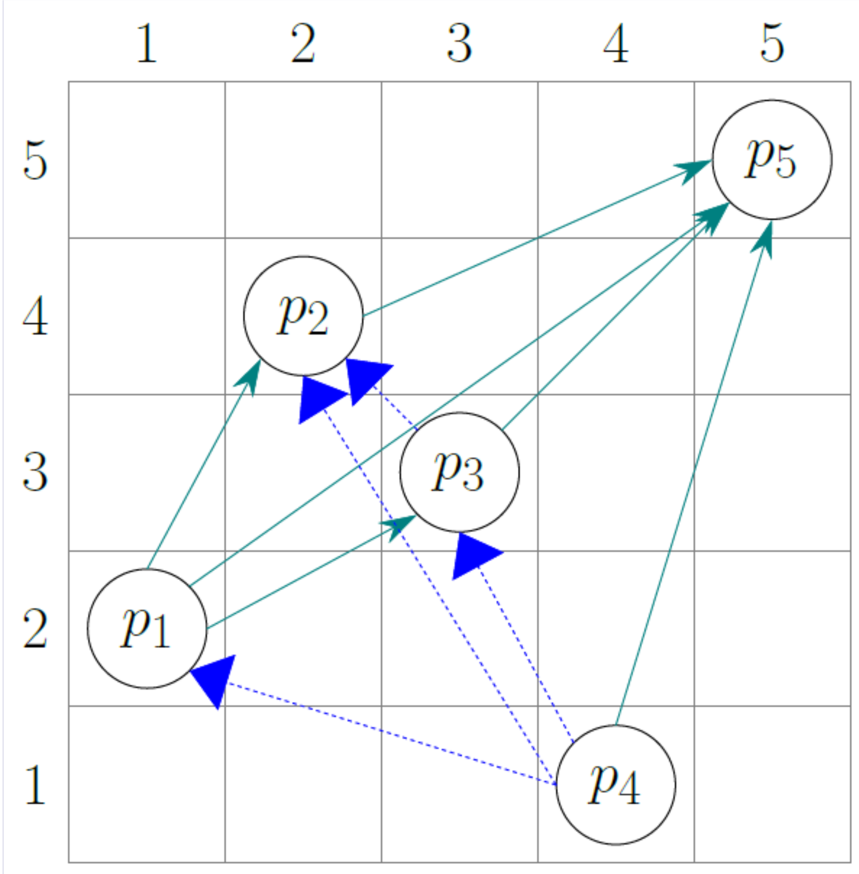


EXAMPLE



FEASIBILITY

- **Directed Acyclic Graph**: link compatible pebbles (in correct order from each other) with green edges (e.g. $p_1 \rightarrow p_2$) and incompatible with blue (e.g. $p_4 \rightarrow p_1$)
- On grid for visibility: vertical arrival horizontal departure
- **Example** arrival $(p_4, p_1, p_3, p_2, p_5)$ and departure $(p_1, p_2, p_3, p_4, p_5)$
  - On the right



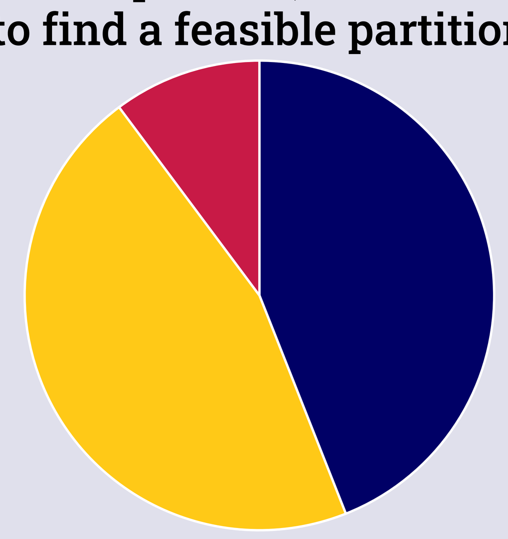
CONCLUSIONS

- With only branches of size two: polynomial solution (bipartite matching[3])
- Introducing branch length and pebble size: NP-hard proof (partition reduction[4])
- **Feasibility approach** using infeasibility constraints and filling branches iteratively
  - Results for proof-of-concept algorithm (shown on the right)
- For branches of 6 or more nodes: NP- hard proof (mutual exclusion scheduling reduction[5])
- **Remaining gap**: branches sizes of three/four/five nodes. Same for double ended tracks[6].

[3] Sedgewick, R. 2004. Algorithms in Java, Part 5 graph algorithm. Addison-Wesley Pearson Education, 3rd edition.
[4] Garey, M. R.; and Johnson, D. S. 1979. Computers and Intractability, A Guide to the Theory of NP-Completeness. United States of America: Bell Telephone Laboratories, Incorporated.
[5] Boge, S.; and Knust, S. 2020. The parallel stack loading problem minimizing the number of reshuffles in the retrieval stage. European Journal of Operational Research, 280(3): 940–952.
[6] Sabine Cornelsen and Gabriele Di Stefano. "Track assignment". In: Journal of Discrete Algorithms 5.2 (July 12, 2006), pp. 250–261. doi: 10.1016/j.jda.2006.05.001.10.

Proof-of-concept implementation

All possible sequences of 4 pebbles, test to find a feasible partition



TU Delft Delft University of Technology

My current PhD research is in collaboration with NS, ProRail and Utrecht University in the RAIL lab, where I look at how we can learn from data to help the planning process of railways hubs to work towards algorithmic tools aiding human planners.