TORS: A Train Unit Shunting and Servicing Simulator

Jacobus G.M. van der Linden[1], Jesse Mulderij[1], Bob Huisman[2], Joris W. den Ouden[2], Marjan van den Akker[3], Han Hoogeveen[3], Mathijs M. de Weerdt[1]

[1]Delft University of Technology, [2]NS (Dutch Railways), [3]Utrecht University

Introduction

When trains are finished with their transportation tasks during the day, they are moved to a shunting yard for maintenance. Public transportation by rail is expected to remain growing for the next decades. However, most maintenance hubs are located in dense urban areas where space is very limited, while the pressure to reduce costs increases. The challenge is to enable more transportation volume without extending the rail infrastructure significantly and to offer robust services to passengers. Scheduling the routing, parking, cleaning and maintenance checks of trains is known as the Train Unit Shunting and Servicing (TUSS) problem. It integrates several known individually hard problems while incorporating many real-life details.

Problem Description

Routing the train units requires planners to find a conflict-free path for trains, starting when a train enters the yard, up to its departure in the morning. During this route, the train units should be parked on the yard.

Extra complexity comes from the possibility to couple and decouple trains into combined units that require only one train operator. Additionally, trains are not easily reversed. To reverse a train, the train driver has to walk to the other side of the train and operate the train from there. Furthermore, arriving trains and departing trains often need to be matched.

The servicing sub-problem considers that each train unit has a specified set of tasks, consisting of cleaning (inside and outside), regular maintenance checks and repairs. Such tasks can only be performed at certain locations on the shunting yard, such as a special washing track or a workshop.

The informal goal of the TUSS problem is to specify a schedule for all servicing activities to be completed for each train in the provided interval that it is on the yard, finding collision free routes for each train to and from tracks where trains are parked and serviced, while respecting all kinds of safety "business rules".

The Simulator

We present an event-based simulator called TORS (Dutch acronym for Train Shunting and Servicing Simulator) that encapsulates the most important practical details of the TUSS problem. In TORS, users can choose actions such as routing a train from one track to another or cleaning it. TORS then validates the action and updates the state according to the input.

TORS includes a simple visualization tool, included both for presentation and debugging purposes. A screenshot of the tool can be seen below.

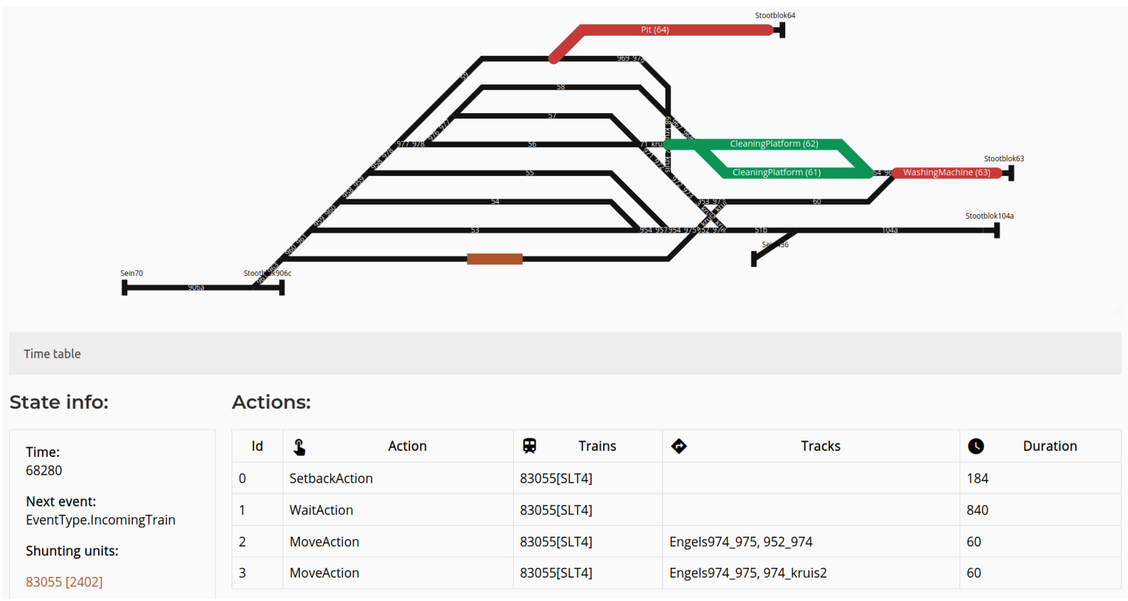

Figure 1: Screenshot of the visualization tool

To enable fast simulation, the core of TORS is written in C++. A python interface is provided to simplify development and testing of new algorithms. Specifically, the OpenAI Gym interface is implemented to comply to the reinforcement learning paradigm. The code is made open-source and available as a git repository: `github.com/AlgTUDelft/cTORS`.

The simulator can test a solution model's quality by running the simulation multiple times for generated scenarios with an increasing number of trains. The quality is determined by the number of trains it can find feasible plans for.

Our Contribution

Our contribution is twofold: First, we provide a translation of a real-world problem to a conceptual model to accommodate researchers the multi-agent community to test their methods in this practical context. Second, with this simulator we facilitate a comparison of different methods and development of new algorithms to solve the train unit shunting and servicing problem.