Comparing & improving planners for train shunting with servicing

Bozhidar Andonov
B.P.Andonov-1@student.tudelft.nl

Issa Hanou
Sebastijan Dumančić

TU Delft

1. Background

Trains are stored at **shunting yards.**
- Deciding where they are parked is known as the Train Unit Shunting Problem (TUSP).
- The TUSP is NP-hard.
- Rolling stock may need **servicing.**

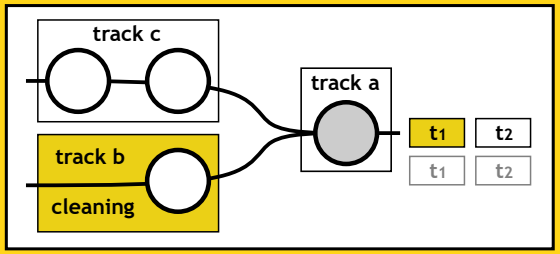

Figure 1. A shunting yard consisting of 3 tracks, one offering cleaning. Trains t_1 (requiring cleaning) and t_2 are parked at this yard.

Algorithmic planners can help:
- Given a domain and instance, produce sequence of actions.
- Mostly tree-like yards and only unit-length trains are considered (Figure 1).

2. Research Aim & Methodology

Goal is to compare and improve planners for the TUSP with servicing domain. For that:

1. Implement TUSP with servicing domain and instances
2. Compare 4 planners against instances
3. Improve best planner

3. Results

1. Domain made in **Planning Domain Definition Language.** Changes from original domain:
- New **`service`** type
- 2 predicates for servicing (Figure 2)
- 2 main actions - `move` and `service-train`

```
(needsService ?train - trainunit ?service - service)
(isServiceTrack ?track - track ?service - service)
```

Figure 2. Two predicates used for servicing trains in the domain.

2. Nine instances created from three shunting yards in the domain (2 of them in Figure 3).

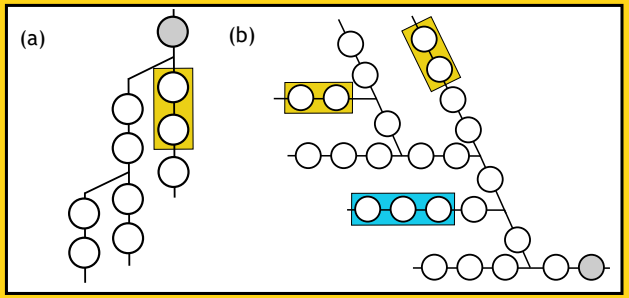

Figure 3. Two of the shunting yards used in the instances: (a) small yard (b) medium yard

3. Four planners from IPC2018 compared against metric based on time and plan size. Planners:
- Baseline (Fast-Downward),
- Team 2 (Fast-Downward),
- Team 4 (Reduces to SAT),
- Team 7 (Fast-Downward)

Results in Figures 4, 5.

4. Team 4 scores best. We **improve** its performance with 31.5% by changing the reduction:
- Removing **redundant clauses**;
- Eliminating **irrelevant step**

Most impact on biggest instances of each yard (Figure 6).

Instance	Before	After	Change
S8	16.0856	10.5742	52.1%
M15	14.0436	17.0402	-18.6%
B30	1660.8350	356.0948	366.4%

Figure 6. Time performance change of improved planner over the 3 biggest instances. First letter signifies shunting yard, digits - the number of trains.

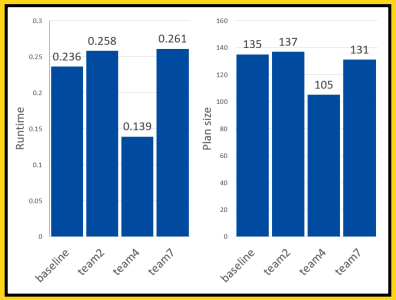

Figure 4. Metrics of 2 instances for the 4 planners.
left: small instance, runtime
right: medium instance, plan size

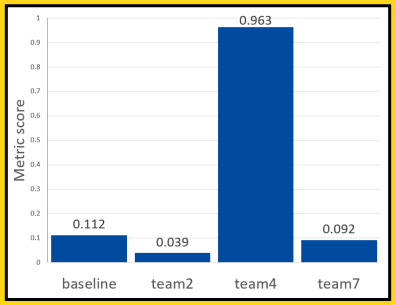

Figure 5. Final metric scores of the 4 planners.

4. Discussion

Regarding **comparison**:
- Fast-Downward is **not** suitable for the domain
- Reducing to SAT turns out to be **very** effective

In terms of **improvement**:
- Redundant clauses are sometimes a hindrance
- Improvement does not always lead to positive results.

5. Conclusions & Future Work

We have compared 4 planners and improved one. Future work:
- Swapping the built-in SAT solver of team 4's planner
- Non-tree-like shunting yard layouts
- Trains not of unit size