White Paper

Intelligent Schedulers for Railways

George Rzevski

Subha Fernando

Abstract

Railway operations are large-scale, safety-critical complex systems in which transport demand must be continuously matched with limited and interdependent resources, including tracks, track blocks, stations, sidings, rolling stock, crews, signalling capacity, and maintenance windows. Operational performance is strongly affected by unpredictable disruptive events such as train delays, infrastructure failures, adverse weather, rolling-stock faults, crew unavailability, and human decision errors. Because railway infrastructure is expensive and difficult to modify quickly, the central management challenge is to allocate existing resources dynamically while maintaining service priorities, safety constraints, and passenger and freight commitments.

This paper presents the design of complex adaptive railway schedulers that combine an ontology-based knowledge base with a swarm of negotiating software agents. The schedulers use the official timetable as an initial reference state and then reschedule affected trains in real time when disruptions occur. The approach replaces purely centralised computational search with distributed interaction among demand and resource agents, enabling rapid conflict detection, localised rescheduling, and improved utilisation of railway resources.

Acknowledgement

We are pleased to acknowledge an important contribution by Professor Petr Skobelev to the two case studies described in this White Paper.

Introduction

Railway timetabling and operational scheduling are traditionally treated as planning and control problems. However, modern railway networks increasingly behave as complex adaptive systems because many interacting elements must respond continuously to changing operational conditions. A delay in one train, a blocked track, or an unavailable crew can cascade across the wider network and affect trains that were not directly involved in the original disruption.

Complexity science provides a useful foundation for this problem because it focuses on systems whose global behaviour emerges from many local interactions. In this context, real-time railway scheduling can be understood as a continuous process of negotiation among trains, blocks, stations, sidings, and operational constraints. This perspective supports a shift from static schedule production toward adaptive schedule management.

Railway Scheduling Problem

The railway scheduling problem can be formulated as the allocation of trains to track blocks over time, while satisfying operational priorities, infrastructure constraints, safety rules, maintenance restrictions, and customer-service commitments.

A track block is a section of railway line that can be occupied by only one train at a given time. This capacity constraint makes scheduling highly sensitive to conflicts, especially in dense networks where passenger, suburban, freight, express, service, and maintenance movements share infrastructure.

Standard operational conditions include:

- Different trains operate at different speeds and acceleration/braking profiles.
- Different train categories have different operational and commercial priorities.
- Overtaking may occur at stations, sidings, or available parallel blocks, subject to safety and signalling constraints.
- Station and siding capacities limit the number of trains that can wait while being overtaken.
- Maintenance activities, signalling restrictions, and rolling-stock availability can reduce usable capacity.

Whenever a disruptive event occurs, the schedule must be adjusted quickly to reduce the impact on directly affected trains and minimise the propagation of delays to unaffected services.

Case Study 1: Moscow-St Petersburg Main Line

The first case study concerns the scheduling of trains on the Moscow-St Petersburg main line, one of the busiest railway corridors in Russia. The line includes approximately 700 km of track, 49 stations, 3,700 track blocks, and 48 sections, including four-track, three-track, double-track, and single-track sections.

The operational environment was highly heterogeneous, comprising approximately 810 trains across categories such as slow passenger trains, suburban trains, express trains, direct high-speed services, freight trains, and service trains. The system also required management of approximately 50 different priority classes. High traffic density in the suburban areas of Moscow and St Petersburg created a particular operational challenge.

A critical requirement was that direct high-speed services should not be delayed under any operating condition. This requirement made disruption management especially challenging because track failures could require closures lasting up to eight hours. The official timetable produced by Railway Headquarters acted as the reference plan and initial state for real-time rescheduling.

Case Study 2: Regional Siberian Railways

The second case study concerns the development of a set of interacting real-time schedulers for the Regional Siberian Railways. The network included approximately 5,000 km of tracks, carried around 5 million passengers per year, and transported approximately 70 million tons of freight annually. The operational context was particularly demanding because trains frequently operated under extreme weather conditions.

Table 1 summarises the annual reasons for delay reported in the source material.

Delay reason	Passenger train delays p.a.	Freight train delays p.a.
Locomotive or crew failures	432	15,819
Passenger or freight carriage failures	155	6,659
Track failures	174	10,067*
Signalling failures	174	672
Power supply failures	99	451
Management errors	442	667

Solution: a Network of Real-Time Schedulers

The proposed solution is a network of cooperating real-time schedulers. Each scheduler network must be customised to reflect the client's specific infrastructure, traffic patterns, priority rules, operational policies, and reporting requirements.

The schedulers load the official timetable as the initial state, generate an operational schedule that covers the relevant resources, and then continuously respond to disruptive events. When a delay, failure, or conflict occurs, the scheduler reschedules affected trains in real time while maintaining correspondence between the computed schedule and the actual field situation.

This approach enables railway timetabling to be divided into two complementary layers:

- Master timetable: the planned reference timetable used by railway authorities, operators, and users.
- Real-time timetable: the adaptive timetable is produced by rescheduling the master timetable whenever disruptions occur.

The main innovation is that operational scheduling is not treated as a one-off optimisation problem. Instead, it becomes a continuous adaptive process in which agents detect conflicts, negotiate resource allocation, and, where possible, revise only the affected parts of the schedule.

To clarify the contribution of the proposed scheduler architecture, Table 2 compares conventional centralised optimisation with adaptive Swarm AI scheduling. The comparison shows that while centralised optimisation is effective under relatively stable conditions, adaptive multi-agent scheduling is better suited to complex railway environments where disruptions, delays, and resource failures occur frequently. The proposed approach improves operational resilience by distributing decision-making across intelligent agents and enabling real-time rescheduling through agent negotiation.

Scheduler Architecture

Each scheduler contains three core components:

- Knowledge Base: an ontology containing conceptual knowledge and databases containing factual operational data.
- Swarm of Agents: software agents representing trains, stations, sidetracks, blocks, and other demand/resource entities.
- Real-World Links: interfaces to railway information systems, sensors, train-position data, dispatching systems, and visualisation/reporting tools.

The ontology defines object classes such as Train, Station, Sidetrack, and Block, as well as relationships such as “Train occupies Block”, “Train travels to Station”, and “Sidetrack belongs to Station”. The databases contain instance-level facts, including train locations, block occupancy, maintenance plans, station capacities, and operational constraints.

Scheduling is performed through negotiation between demand agents and resource agents. A typical demand agent is a Train Agent, while a typical resource agent is a Block Agent, Station Agent, or Sidetrack Agent. For a train to travel from point A to point B, all required blocks must be available at the relevant times. When overtaking is required, Station Agents and Sidetrack Agents participate in the negotiation to identify feasible waiting or passing arrangements, see Figure 1.

At the operational scale, more than 40,000 agents may be active simultaneously. Decisions are produced through local interactions among agents and then communicated to the corresponding operational systems or dispatchers.

Table 2: Comparison between Conventional Centralised Optimisation and Adaptive Swarm AI Scheduling

Aspect	Conventional Centralized Optimization	Adaptive Swarm AI Scheduling
Decision-	Decisions are made by a central	Decisions emerge through negotiation

Aspect	Conventional Centralized Optimization	Adaptive Swarm AI Scheduling
making approach	optimiser using predefined models and constraints.	among intelligent agents representing trains, tracks, stations, sidings, and other resources.
Response to disruptions	Requires recalculation of the schedule, which can be slow for large and complex railway networks.	Responds in real time by allowing affected agents to negotiate local changes without disturbing unaffected parts of the schedule.
Scalability	Computational complexity increases significantly as the number of trains, tracks, and constraints grows.	Highly scalable because scheduling responsibility is distributed across many cooperating agents.
Flexibility	Less flexible when unexpected events occur, especially when the disruption is not anticipated in the original model.	Highly flexible and adaptive to failures, delays, changing resource availability, and operational uncertainty.
Knowledge representation	Usually relies on mathematical models, rules, and fixed optimisation constraints.	Uses a knowledge base containing ontology, operational rules, factual data, historical patterns, and learning experience.
Conflict resolution	Conflicts are resolved centrally, often requiring global recalculation.	Conflicts are resolved locally through negotiation between demand agents and resource agents.
Real-time capability	Limited in highly dynamic environments due to computational search time.	Strong real-time capability because computational search is replaced by interaction among agents.
Best suited for	Stable or moderately predictable railway environments.	Complex, uncertain, and disruption-prone railway operations.

Schedule Construction Process

The schedule is constructed in three main stages.

1. Initial schedule generation: A rough schedule is produced from the official timetable. This first version allows all train movements to be represented and exposes conflicts where multiple trains require the same block or resource.
2. Conflict resolution: Train Agents and Block Agents negotiate access to contested blocks. Where overtaking is required, Station Agents and Sidetrack Agents join the negotiation. Train priorities are respected where possible, subject to safety and feasibility constraints.
3. Schedule balancing: Remaining anomalies, such as excessive delays imposed on low-priority trains, are adjusted to produce a balanced operational schedule.

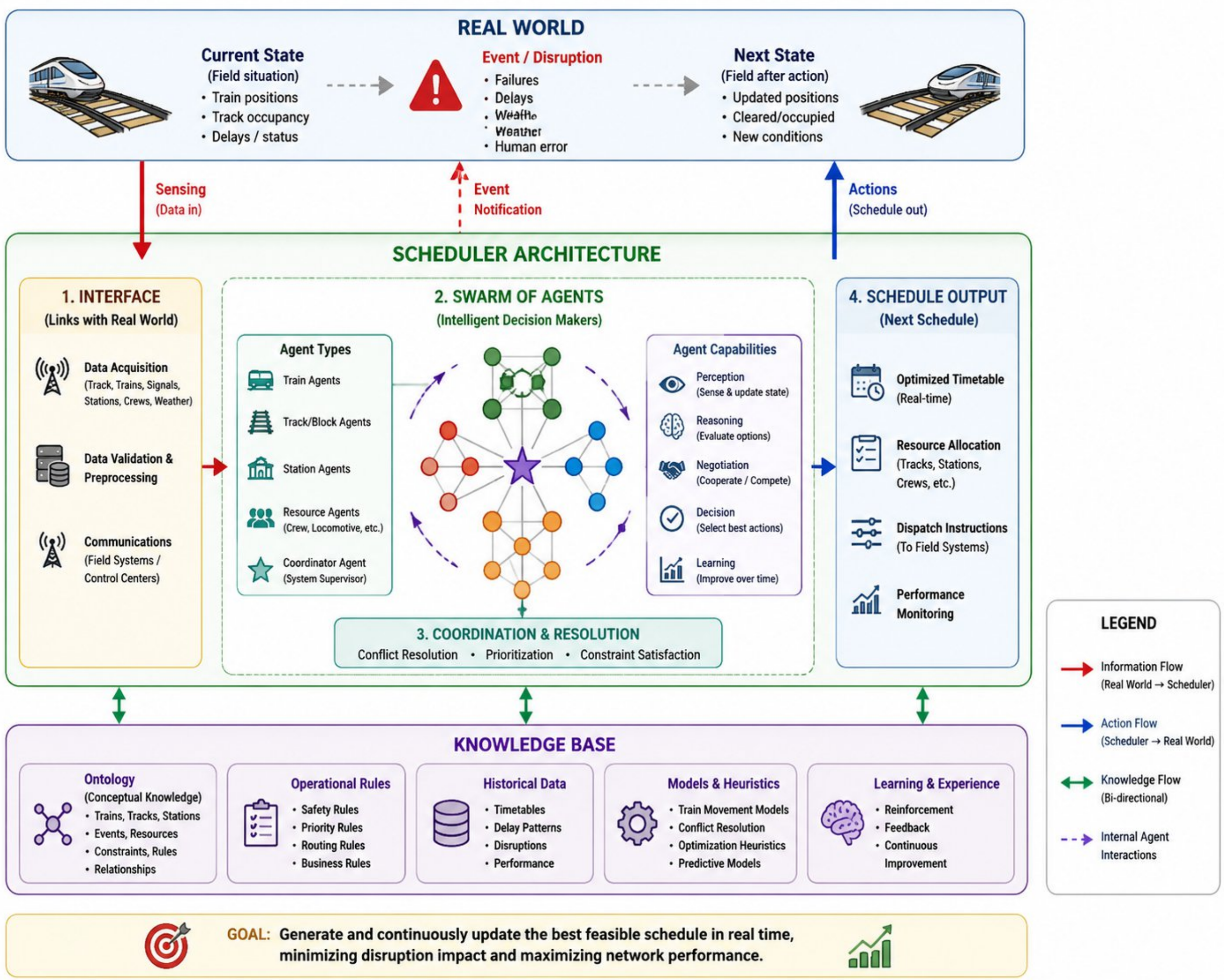

Figure 1: Architecture of an Adaptive Swarm AI Railway Scheduler

Negotiations are conducted to maximise Enterprise Value, a composite objective function that balances competing goals such as train priorities, delay minimisation, reduced use of reverse-direction blocks, safety constraints, and special customer or operational requirements.

If a selected scheduling strategy produces unsatisfactory Enterprise Value, the agents may discard the inadequate schedule and restart using alternative strategies. This mechanism, described as constructive destruction, allows the system to escape poor local solutions and search for more effective operational arrangements.

Enterprise Value (EV) can be expressed as a weighted objective function:

$$EV = w_1 P + w_2 D + w_3 R + w_4 C + w_5 S$$

where Prepresents priority satisfaction, Ddelay minimisation, Rresource utilisation, Cconflict resolution, Sschedule stability, and w_1–w_5 are weights assigned according to operational priorities.

Salient Features

The proposed scheduler network provides the following operational capabilities:

- Loading of master timetables and maintenance plans.
- Integration of infrastructure sensor data and train-position data.
- Real-time processing of train location and track-occupancy information.
- Consideration of more than 80 scheduling criteria, including train length, weight, acceleration, braking time, track layout, safety rules, and security constraints.
- Real-time reaction to unpredictable disruptive events such as track failures, train failures, and delays.
- Rapid resolution of unforeseen conflicts, including competing requests for the same siding or block.
- Visualisation of outputs to support dispatcher decision-making.
- Creation and storage of operational reports.

A key strength of the Swarm AI approach is that computational search is partly replaced by search through agent interaction. Instead of recalculating the entire schedule centrally, the system identifies affected agents and resources, negotiates locally, and updates the schedule before the next disruption occurs.

Results and Operational Impact

The early version of the adaptive Swarm AI scheduling system substantially reduced the time required to reschedule trains after disruptive events and improved railway resource utilisation. The reported improvements include:

- High-speed train delays were practically eliminated.
- Average train delays were reduced to below 8%.
- Trains recovered time lost due to disruptive events approximately 1.5 times faster.
- Dispatcher productivity doubled.
- Losses due to delays were reduced by approximately 50%.
- The system could reschedule approximately 800 trains within one minute.
- Customer satisfaction improved.
- The approach created opportunities to increase traffic density.

These results indicate that adaptive Smart AI scheduling can provide value in operational environments where disruptions are frequent, infrastructure capacity is constrained, and manual dispatcher decision-making is insufficient for large-scale real-time coordination.

Conclusion

Railway timetabling is a mature discipline, but the increasing density, variability, and operational interdependence of modern railway networks require adaptive real-time scheduling methods. Complexity science and multi-agent systems provide a strong foundation for this shift, as they enable large-scale railway operations to be modelled as interacting demand and resource entities.

The case studies demonstrate that ontology-based, Swarm AI schedulers can support rapid rescheduling, reduce delays, improve resource utilisation, and assist dispatchers under difficult operating conditions. The approach is particularly relevant for railway networks that face frequent disruptions, dense mixed traffic, strict service priorities, and limited infrastructure expansion options.