Developing Integrated Tools to Optimise Railway Systems: An Overview

John Preston[a], Attila Kovacs[b], Phillip James[c] and Hongbo Ye[d] on behalf of the DITTO Consortium
[a] Faculty of Engineering and the Environment, University of Southampton,
Building 176, Boldrewood, Burgess Road, Southampton, SO16 7QF, UK
E-mail: jpreston@soton.ac.uk, Phone: +44 (0) 23 8059 4660
[b] Mathematical Sciences, University of Southampton, University Road, Southampton,
Highfield, SO17 1BJ, UK.
[c] Computer Science, Swansea University, Singleton Park, Swansea, SA2 8PP,Wales,
UK.
[d] Institute for Transport Studies, 34-40 University Road, University of Leeds, Leeds,
LS2 9JT, UK.

Abstract

This paper provides an overview of the DITTO (Developing Integrated Tools to Optimise) project and presents some of its initial findings. The aim of DITTO is to increase the practical capacity of railway systems by developing computerised tools that can determine the theoretical capacity of a given infrastructure, optimise timetables in the face of delays and optimise real-time traffic management in situation of both fixed block and moving block signalling. Some prototype tools are developed and applied to the East Coast Main Line (ECML) railway in the UK. In particular, a stochastic version of job shop scheduling is applied to the station of Peterborough. It is found that additional services can be provided on the local network but at the expense of service performance. However, when a wider network area is considered, the number of feasible additional services appears to be much reduced.

Keywords

Capacity, Control, Optimisation, Safety, Simulation.

1. Introduction

In the last 20 years, rail traffic on the national network in Britain has grown by around 100% in terms of passengers and freight and by 50% in terms of train movements, whilst the overall quantum of infrastructure has barely changed (ORR, 2016). To meet the challenges that such growth presents, the UK rail sector has established the Future Traffic Regulation Optimisation (FuTRO) research programme which is examining the ways that advances in technology, including those associated with the digital railway, can improve rail operations. FuTRO is thus developing the control, command and communications theme of the Rail Technical Strategy (RSSB, 2012). One of the projects that has been commissioned by FuTRO is Developing Integrated Tools to Optimise (DITTO) Railway Systems, funded by RSSB (formerly the Rail Safety and Standards Board) for three years from September 2014 – see www.dittorailway.uk. DITTO is a consortium of researchers based at universities in Leeds, Southampton and Swansea. Industrial support has been provided by Arup, Siemens Rail Automation and Tracsis. It builds upon separate projects undertaken by the three Universities for the RSSB/EPSRC Capacity at Nodes programme that ran from 2010 to 2012. The three projects were Challenging Established Rules for Train Control (Leeds), Overcoming Capacity Constraints: A Simulation Integrated with Optimisation of Nodes (OCCASION – Southampton) and SafeCap (Swansea) (- see Goodall et al., 2013).

DITTO will contribute to FuTRO by establishing basic principles and proofs of concept and by developing optimisation formulations, algorithms and processes that will deliver a step change in rail system performance and help to meet future customer needs. This will be done by taking into account developments in human and automatic control on trains and in control centres (particularly related to ERTMS) and by making better use of data, particularly with respect to time and position of trains.

DITTO's objectives are thus to:

1. Develop optimisation activities that maintain safe operating conditions and do not exceed theoretical capacity limits.
2. Develop timetables that optimise capacity utilisation without compromising service reliability.
3. Combine dynamic data on the status of individual trains to produce an optimal system-wide outcome in terms of traffic management.
4. Use Artificial Intelligence to produce tractable solutions to real-time traffic control.

Objective 1 relates to network optimisation. It determines the infrastructure required to provide the theoretical capacity required in a safe manner. Objective 2 relates to plan optimisation. It involves matching trains to the infrastructure so as to maximise the throughput of trains subject to acceptable levels of performance, primarily in terms of punctuality. Objective 3 relates to traffic management optimisation. It involves dynamically controlling trains to minimise the impact of service disruptions. Objective 4 attempts to integrate the three optimisation processes described above by using machine learning tools. The focus of this paper will be particularly on Objective 2, but will demonstrate the inter-relationships with the other three objectives. The approach adopted to optimising the railway life-cycle is illustrated by Figure 1.

Figure 1: Optimising the Rail Life-Cycle.

The DITTO project thus consists of four inter-related and complementary technical strands that are innovative both on their own and in combination with each other.

Safety – this strand allows optimisation activities to proceed in the knowledge that safe operating conditions are being maintained and that theoretical capacity limits are not being exceeded. We discuss our work in this area in Section 2.

Reliability – this strand quantifies the trade-offs between the provision of additional train services and the maintenance of service quality so as to develop timetables that optimise capacity utilisation without compromising service reliability. We discuss our work in this area in Section 3.

Dynamic simulation – micro-level data on the status of individual trains will be combined to produce an optimal, macro-level outcome, transmitting the system-wide needs back to the micro-level, so that individual train movements can be optimised within overall system requirements. We briefly discuss this in Section 4.

Network integration – using artificial intelligence, optimised timetables are produced that can be adjusted in real time through dynamic simulation. Our work in this area has not yet started but we discuss our intentions in Section 5.

The DITTO approach is illustrated by Figures 2 to 4 (see also James et al., 2015a)

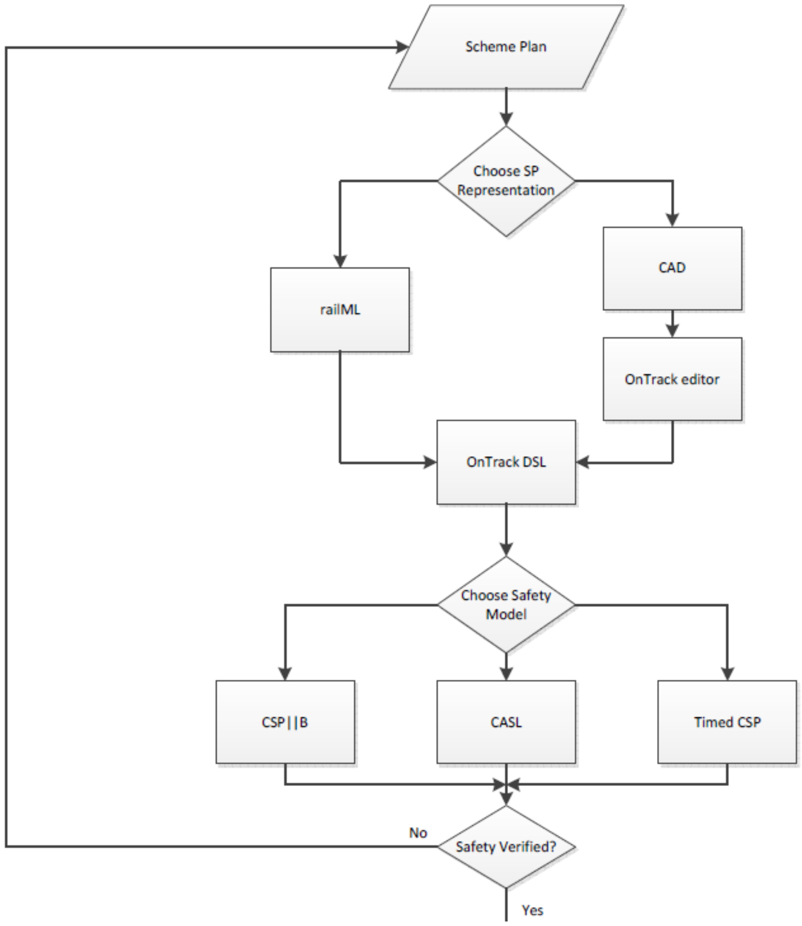


Figure 2: Safety and Capacity Validation.

Figure 2 shows that our starting point is to put a Scheme Plan (SP) through a safety and capacity verification process. This might use either RailML or output from Computer

Assisted Design (CAD) and the OnTrack editor (see James et al., 2015b). These approaches are brought together by the OnTrack Domain Specific Language (DSL) developed by Swansea University. Safety verification can then be performed using a variety of languages such as: CSP, Communicating Sequential Processes; a specification language for concurrent systems defined by Sir Tony Hoare in the early 1980s; CSP Parallel B, a combination of CSP and the specification language B, defined by Swansea's research partners at Surrey University around 2000; and CASL, Common Algebraic Specification Language.

Figure 3 shows that the next stage is to undertake the static optimisation. This takes the existing Timetable (TT) in CIF (Common Interface Format) and the safety limits established by the verification and, using Capacity Utilisation Indices or other related approaches, assesses the likely performance in terms of Congestion Related Reactionary Delay (CRRD). Performance scenarios are then developed to feed into a stochastic optimization based on a variant of job shop scheduling, in which tracks are treated as machine shops and train movements are treated as jobs. This involves a two stage stochastic program. In the first stage, new trains are inserted into the timetable. The second stage involves optimising for reliability for various random scenarios. This is undertaken at the meso-level, for example for a node such as Peterborough on the East Coast Main Line (ECML). The implications are assessed at a macro-level, for example for the ECML between Doncaster and Alexandra Palace using a variant of the Multi-Commodity Network Design Problem (MCNDP). Constraints ensure that the revised timetable is within safety limits. Once the optimisation is confirmed at the meso- and macro-levels, it is fed into the final stage.

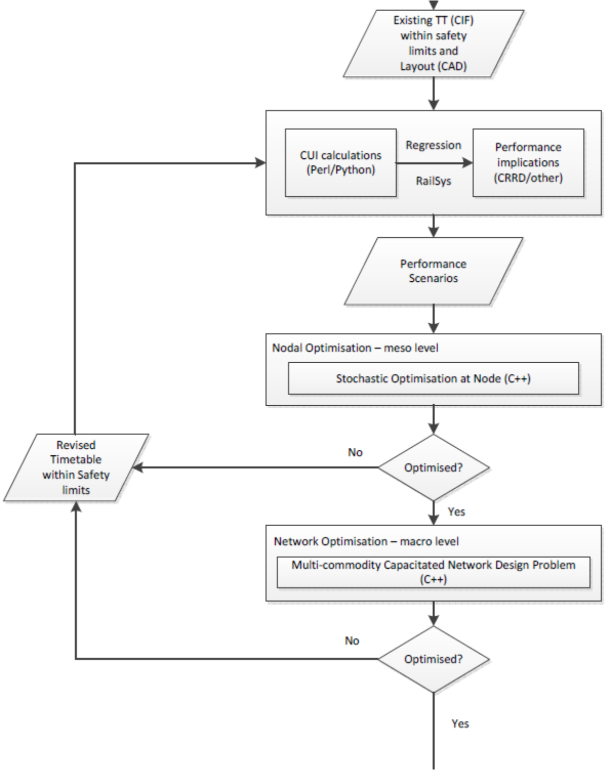


Figure 3: Static Optimisation

The final stage involves examining the scope for dynamic rescheduling and this is done by using the Trackula train simulator, along with examining traditional algorithms, alongside human control and artificial intelligence based on machine learning. This is informed by historic data on performance (in terms of delays) that has also informed the static optimisation but may consider a wider range of scenarios. The final output, as illustrated by Figure 4 is an optimised timetable, along with a series of rescheduling plans, if needed.

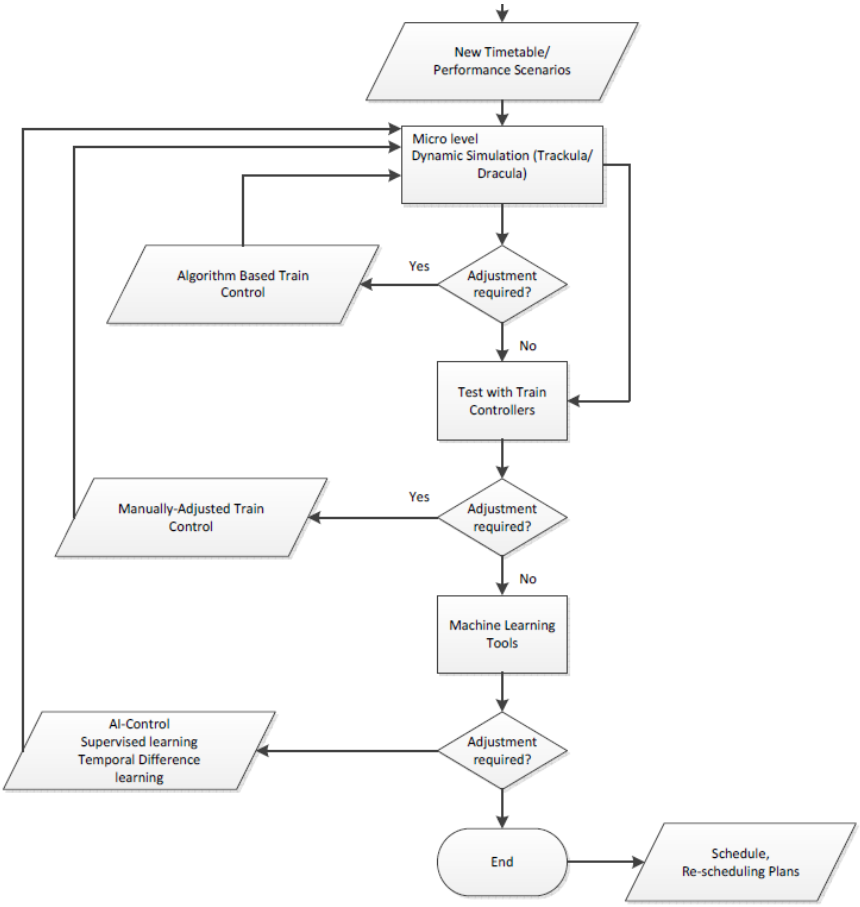


Figure 4: Dynamic Rescheduling

Our work draws on the rich literature in this application domain, with a particular emphasis on rail capacity (for reviews see Abril et al, 2008 and Kontaxi and Ricci, 2012). These reviews have highlighted a number of approaches to rail capacity management, including analytical methods (non-parametric and parametric), simulation, optimisation and integrated assessment. DITTO is attempting to provide an integrated assessment by combining analytical, optimisation and simulation approaches with formal methods for safety and capacity verification.

2. Safety and Capacity Analysis

Work in this area has included safety verification and the development of the digital characterization of rail infrastructure using the OnTrack tool (James et al., 2015b, 2016). From the ECML case study area, a number of nodes have been chosen for safety analysis. As a major objective in the DITTO project is integration with the Birmingham Rail Virtual Environment (BRaVE) tool (see Chen et al., 2016) developed in the parallel DEDOTS project, this data is based on the representations in the BRaVE simulator (see www.bravesim.org) with the nodes analysed covering those presented in Table 1.

Table 1: Details of Five Nodes on the East Coast Main Line

Node	Number of Control Table Entries
Allington	113
Barkston	88
Claypole	154
Grantham	502
Newark	1,190

2.1 Safety Analysis

We carried out our safety analyses using two different modelling approaches: one based on the formal specification language CSP||B, the other based on the language CSP. These approaches follow the line of work undertaken by the European Technical Working Group for Rail Control (ETWG-RC), a group led by the Swansea Railway Verification Group (see www.cs.swansea.ac.uk/rail) which includes similar groups from the Technical University of Denmark, Universita degli Studi di Firenze, and the Universities of Eindhoven, Twente, Bremen and Surrey. Both of the modelling approaches are part of the OnTrack toolset. Here the main motivation for having two independent verification approaches is that there is an increased trust in the correctness results.

Our experience shows (e.g. as reported in Moller and Roggenbach 2016) that CSP gives superior performance over CSP||B, for example, for the most complex node analysed to date (Newark – see also Table 2). This change in performance is due to a different style of modelling that has been undertaken. The CSP||B modelling focuses on how information flows through the various elements of the railway: these have been identified to be Controller, Interlocking, Track Equipment and Trains. Each of these elements sends and receives information to the others: the Controller selects and releases routes; the Interlocking serves as a safety layer between Controller and Track Equipment; and the Track Equipment consists of elements such as signals, points and track units. Some of these elements have states: in particular, a Point can be in normal or reverse positions, and Signals can show proceed or halt. Finally, Trains have drivers who determine their behaviour. The data-rich Interlocking component is modelled by a single B-machine, while the Controller and Trains run independently of each other using the CSP interleaving operator. Thanks to having a generic model, one only has to instantiate the model with the location-specific data.

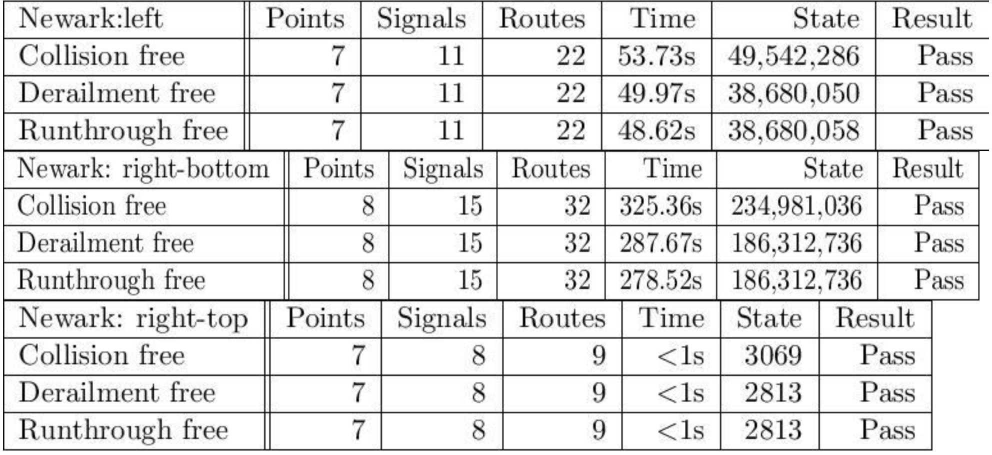
The purpose of our CSP||B models is to verify the correctness of the Control and Release Tables. However the CSP models consist of two parts: the static part and the dynamic part. The static part encodes the topology of track plans as well as the control data associated with it (i.e. control/release tables). The dynamic part of the CSP code encodes a core subset of signalling principles from national or regional authorities supporting features such as: flank protection in route setting; front wheel replacement of signals; train operated route release; sequential release of route locking; simplified version of comprehensive approach locking; and reversing of trains.

Overall, the better performance of CSP can be attributed to the following three factors.

- *Modelling*: Based on previous experience with CSP||B, a systematic approach was taken, building a minimal model that includes only the essential elements, and taking into account a number of well-established modelling ideas (as highlighted above).
- *Technology*: The underlying model-checkers ProB (for CSP||B) and FDR3 (for CSP) are tools from different generations. Whilst ProB was built in the early 2000s and has stayed constant ever since with regards to its model checking algorithm, FDR3 represents a recent (2013) re-implementation of the model-checker FDR2. This re-implementation makes systematic use of algorithmic advances over the last decade. It remains future work to measure the influence of technology by running the ProB model checker on our new CSP models.
- *Granularity*: The CSP||B model is monolithic in the part that represents the interlocking. In contrast to this, the CSP model consists of a large number of small processes. The latter granularity is what makes the CSP model checking faster.

In combination with covering and route decomposition techniques, CSP was able to verify the Newark node (see Table 2), whilst CSP||B was timed out due to a state explosion.

Table 2: Safety Verification of Newark
(Note that the node is decomposed in to three parts)



Newark:left	Points	Signals	Routes	Time	State	Result
Collision free	7	11	22	53.73s	49,542,286	Pass
Derailment free	7	11	22	49.97s	38,680,050	Pass
Runthrough free	7	11	22	48.62s	38,680,058	Pass

Newark: right-bottom	Points	Signals	Routes	Time	State	Result
Collision free	8	15	32	325.36s	234,981,036	Pass
Derailment free	8	15	32	287.67s	186,312,736	Pass
Runthrough free	8	15	32	278.52s	186,312,736	Pass

Newark: right-top	Points	Signals	Routes	Time	State	Result
Collision free	7	8	9	<1s	3069	Pass
Derailment free	7	8	9	<1s	2813	Pass
Runthrough free	7	8	9	<1s	2813	Pass

2.2 Capacity alongside safety

Work has also been undertaken to analyze capacity in a manner that fits industrial design processes. This takes track plans, lists of routes and design rules, and jointly undertakes safety and capacity analyses. This is done by adding time to our CSP models and can illustrate the effects of scheme plan changes on capacity, as shown by Figures 5a and 5b.

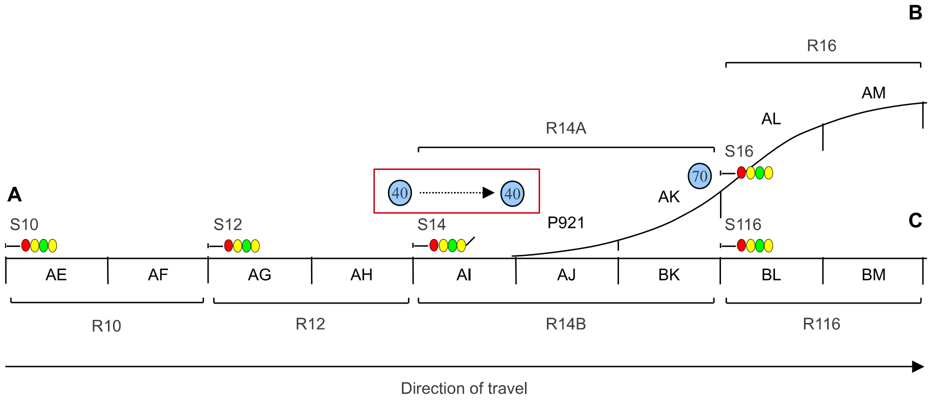


Figure 5a: A single junction network

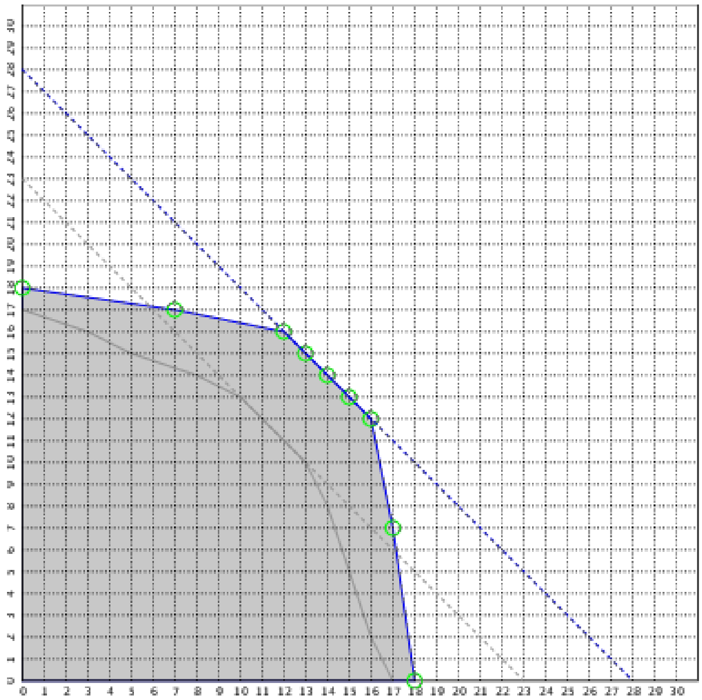


Figure 5b: Theoretical Capacity of the single junction network

Figure 5b shows the theoretical network capacity of the single junction in Figure 5a. Our definition of network capacity is: the number of trains that one can observe during a fixed period of time passing through and being within a given rail network. Here we can differentiate between trains; i.e. in the case of the single junction we differentiate between trains on the main line (travelling from A to C) whose number is shown on the x-axis, and trains on the side line (travelling from A to B) whose number is shown on the y-axis. The network capacity thus consists of a pair of numbers (a,b) which says: within a given period of time, one can observe “a” trains travelling from A to C and “b” trains travelling from A to B. Figure 5b displays the analysis of two different settings concerning a 40 miles per hour speed restriction. In the first setting (grey line), the 40 miles per hour is placed at the end of track AH, in the second setting (blue line), it is placed at the end of track AI.

The visualisation in Figure 5b says: for a point (x,y) above the blue line (grey line), it is impossible to find a schedule to observe x trains travelling from A to C and y trains travelling from A to B; but for any point (x,y) on or below the blue line (grey line), there exists at least one schedule to observe x trains travelling from A to C and y trains travelling from A to B. For instance, (20,18) is a capacity that is impossible, whereas (10,10) is a capacity that is achievable. Observed effects include:

1. Moving the 40 miles per hour sign from AH to AI increases capacity.
2. In order to achieve the best utilization of the infrastructure, i.e., to find a theoretically-possible network capacity (x,y) which maximizes the sum total x+y of trains that can be observed, one has to schedule mixed traffic, i.e., trains both going from A to B and trains going from A to C.

Effect 1 is in line with expectations from signalling engineers from Siemens; and indeed Siemens engineers designed this example as a first litmus test for capacity modelling. Effect 2 is in line with our analyses with pen and paper, where it became clear that in order to maximise network capacity one needs to make use of all tracks. In our example, never having a train on tracks AK, AL, or AM leads to wasted capacity.

The numbers shown in Figure 5b have been produced using the following assumptions. We are observing the junction for a period of 15 minutes, starting with the junction being empty. There is an “infinite supply” of trains entering at AE. Trains enter the system at full speed of 90 miles per hour. Trains travelling from A to C travel at a constant 90 miles per hour. Trains travelling from A to B slow down in time to reach 40 miles per hour at the speed reduction sign, and then speed up to 70 miles per hour after they have passed the 70 miles per hour sign on track AK. Trains have a length of 100 meters; and all tracks are 500 meters long, with the exception of track AJ (the point) which is 50 meters long. Trains accelerate and brake at a rate of 1 metre/second^2. (All speeds, speed changing rates, and lengths have been given to us as realistic data by Siemens.)

It is worthwhile noting the timed CSP modelling used in Figure 5b is “of the same kind” as the one we are using for safety analysis; more concretely: starting from our timed CSP model used for capacity analysis, when we abstract away the timings, we obtain our CSP model used for safety analysis. We see this as a major achievement, as capacity and safety are intertwined. Capacity without safety is pointless; and adding measures to obtain safety usually decreases capacity. The modelling follows the fundamental ideas introduced in Isobe et al. (2012).

The network capacity shown in Figure 5b has been computed using the FDR3 model checker. To this end, we discretise time in our models in timed CSP. Rather than working with continuous time, we sample the system once per second. For analysing capacity, the resulting discretization error is marginal. Note, however, that there is no such error in our analyses for safety. It takes of the order of 15 minutes to “measure” the theoretical line capacity of the single junction with FDR3 on a moderately-powerful computer. It remains future work to optimise the verification time and to try this new method on larger and more interesting rail nodes.

2.3 ERTMS

ERTMS extends classical signalling systems by adding a radio block centre and adding control computers to trains. This allows ERTMS/ETCS Level 2 to take into account speed and braking curves of each individual train. These determine, for each train individually, the train's braking point well in advance of the end of the movement authority that the ERTMS signalling system has granted to the train. This should separate trains by shorter margins (compared to classical signalling systems) and thus increase capacity. Concerning formal safety analyses, for ERTMS it is necessary to develop and analyse timed or hybrid models. Note that – as ERTMS level 2 still includes interlockings – the challenges for formal safety analysis for classical interlocking designs remain, and are extended by new dimensions.

More specifically, an ERTMS/ETCS system consists of a controller, an interlocking (a specialised computer that determines if a request from the controller is “safe”'), a radio block centre, track equipment, and a number of trains. Whilst the ERTMS/ ETCS standard details the interactions between the trains and track equipment (e.g. in order to obtain concise train position information) and the radio block centre and trains (e.g. to hand out movement authorities), the details of how the controller, interlocking and radio block centre interact with each other are left to the suppliers of signalling solutions, such as our industrial partner Siemens. In this paper, we work with the implementation as realised by Siemens. In the following we refer to this system simply as ERTMS.

One development step when building an ERTMS system consists of developing a so-called detailed design. Given geographical data such as a specific track layout and what routes through this track layout shall be used, the detailed design adds a number of tables that determine the location-specific behaviour of the interlocking and radio block centre. To the best of our knowledge, our modelling of ERTMS is the first one comprising all ERTMS subsystems required for the control cycle in ERTMS Level 2.

The objective of our modelling is to provide a formal argument that a given detailed design is safe. Here we focus on collision freedom, though our model is extensible for dealing with further safety properties such as derailment and run-throughs, and potentially with performance analysis.

We base our modelling approach on Real-Time Maude, a language and tool supporting the formal object-oriented specification and analysis of real-time and hybrid systems. In order to obtain a faithful model of ERTMS/ETCS level 2 on the design level, we follow a methodical approach, established by the Swansea Railway Verification Group.

As a first modelling step, we systematically identify the entities of ERTMS, describe their abstract behaviour; and determine the abstract information flow between them, all in line with the design by Siemens (see Figure 6)

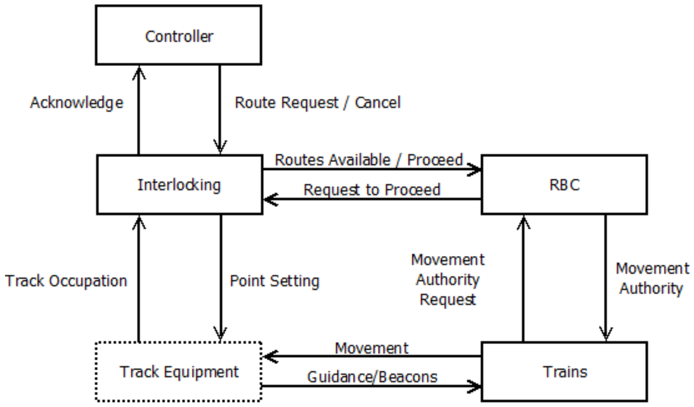


Figure 6: Representation of ERTMS/ETCS Level 2

The (manual or automated) controller is responsible for controlling the flow of trains through the railway network; it completes this task by sending “route request” messages to the interlocking. These route requests are dependent on elements such as the current timetable to be adhered to and details on congestion within the network. For simplicity, we abstract from “route cancel” and “acknowledgement” messages.

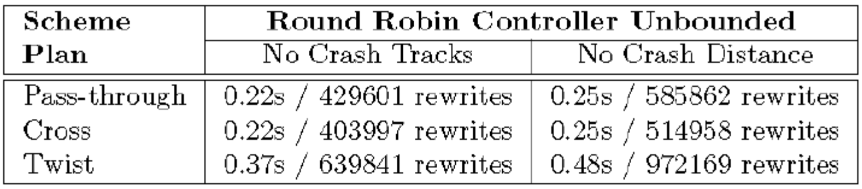
The interlocking is responsible for setting and granting requested routes. Once the controller has requested a route, the interlocking will use information on current track occupation and point settings (from the track equipment) to determine if it is safe for the requested route to be set. Whether a route can be set or not is computed in a process based on the conditions stipulated by the control table. Once the interlocking has checked that all points on the route are free to move or already in the right position, it will send a “route available” message to the Radio Block Centre (RBC). This informs the RBC that the route is free for use, though it is not yet reserved for a particular train. The RBC initiates the process of locking a route for a train by sending a “request to proceed” message to the interlocking. On receiving this message, the interlocking will then ensure that, based on the control table, all tracks for the route are free and that the points are indeed locked in the required positions. Once this step is completed, the interlocking sends a “proceed” message to the RBC indicating that a train can use the route.

The RBC's main responsibility is to take the route information presented by the interlocking and use it to manage the movement of trains across geographic positions on the railway. To do this, the RBC and trains use the notion of a movement authority. A movement authority is an area of geographical railway that a train is permitted to move within. The furthest point along the railway to which a train is permitted to move is indicated by a point called the end of authority (EoA) which is given to a train by the RBC. The data within such an EoA includes the distance that the train can travel, along with a marker board towards which the train is travelling. As a train moves across the railway network, it uses beacons on the track to continually calculate its position. When it is nearing its EoA, it makes a new “movement authority request” to the RBC indicating that it would like its movement authority to be extended. After receiving this request, the RBC maps the physical location of the train to an available continuation route that has been presented to it by the interlocking. This calculation is performed based on a look-up table designed as part of the RBC for a scheme plan. It will then issue a “request to proceed” message to the interlocking for this route. Once the RBC has received a “proceed” message from the interlocking, it will compute a new EoA for the train based on the route that has been granted; again, this information is provided by a look-up table. This new EoA is then finally sent as a “movement authority” message to the train.

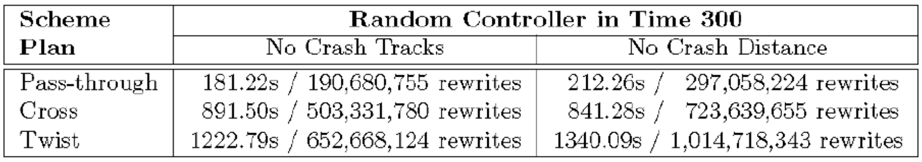
The behaviour of trains is parameterised by maximum speed, acceleration and braking curves. We make a maximum progress assumption for trains, i.e. trains are running as fast and as far as possible. Thus, if a train has a movement authority beyond its current position, it will accelerate towards its maximum speed; and when the maximum speed is reached, the train will continue to travel at this speed. Furthermore, whilst accelerating or travelling at maximum speed, the train will start braking at the last possible time for which it is guaranteed to come to a halt before the EoA. Trains are guided by the track layout, respecting the positions to which the interlocking has set points. As trains move along the track, track equipment sense track occupation and report this to the interlocking.

We assume that track equipment (points, track circuits, beacons etc.) function correctly and that points move instantaneously. This is justified as our verification aim is to establish correctness of the location and train specific design parameters for an ERTMS system for a single geographic region. Therefore, we refrain from modelling track equipment.

Table 3 shows the verification times and the number of rewrite steps for three scheme plans against two different control strategies: a round-robin controller and a random controller (for details see James et al. 2015c).



Scheme Plan	**Round Robin Controller Unbounded**	
	No Crash Tracks	No Crash Distance
Pass-through	0.22s / 429601 rewrites	0.25s / 585862 rewrites
Cross	0.22s / 403997 rewrites	0.25s / 514958 rewrites
Twist	0.37s / 639841 rewrites	0.48s / 972169 rewrites



Scheme Plan	**Random Controller in Time 300**	
	No Crash Tracks	No Crash Distance
Pass-through	181.22s / 190,680,755 rewrites	212.26s / 297,058,224 rewrites
Cross	891.50s / 503,331,780 rewrites	841.28s / 723,639,655 rewrites
Twist	1222.79s / 652,668,124 rewrites	1340.09s / 1,014,718,343 rewrites

Table 3: Verification Results of Model Checking with restricted control strategy (top) and and random control strategy (bottom)

The results show that unbounded model checking is successful when control is restricted, e.g. to our round-robin controller. This is due to the restrictions that such a control strategy puts on train movements through the scheme plan. However, when using our random controller, the state space increases. Moreover, there are infinite traces possible, e.g., by the controller choosing the same route over and over again. Thus, we provide results for up to a given time bound of 300 seconds. Note that this time is enough to ensure that both trains can travel completely through each of the scheme plans.

Another phenomenon is that model checking for the logical safety condition “No Crash Track” requires fewer rewrites (approximately 20%) than for the physical safety condition “No Crash Distance”. This follows one's intuition.

As expected, model checking times increase with the complexity of the scheme plans. One naive complexity measure would be the number of routes available in a scheme plan.

We note that there are five routes in the Pass-through station; six routes in the Cross; and eight routes in the Twist. This again follows intuition, as the random controller has more freedom in more complex track plans. Note that this observation does not necessarily carry over to the round robin controller: here, the order in which the routes are requested plays a role as well and can possibly overshadow this effect.

Finally, it is future work, to consider, more varied rail-yards, and also how the frequency of controller requests affects model checking results.

3. Reliability and Plan Optimisation

Our optimisation problem is based on the job shop scheduling formulation associated with Liu and Kozan, 2009 – see also Bektas et al., 2015. Our initial work in the pre-cursor OCCASION project was with deterministic formulations (Paraskevopoulos et al., 2015).

The associated problem of scheduling trains in a stochastic environment is complex, although a standard method of overcoming this complexity is to use a sample average approximation (SAA) approach (Kleywegt et al., 2002). The stochastic optimisation that we develop has been applied at a meso-level to individual stations (which are themselves assemblages of nodes and links). At a macro-level, we model the network as a Multi-Commodity Network Design Problem (MCNDP) (Bektas et al., 2010). The job shop scheduling and the MCNDP are iterated until an overall feasible and robust timetable is found.

3.1 Stochastic Job Shop Scheduling

Our work on the interrelationships between rail service performance and capacity utilisation is given in an accompanying paper (Armstrong and Preston, 2017) and in Armstrong and Preston (2015). A viable approach to keeping up with increasing numbers of railway passengers is to run more services at peak times; that is, add more services to the timetable. However, more traffic means more conflicts amongst trains; the tighter the capacity constraints, the more conflicts. Without sufficient buffer times to absorb uncertain delays, the delay of one train might propagate over the entire network.

Given this, we address a realistic timetabling problem by considering the number of services offered along with their reliability. The approach adopted is detailed in Kovacs et al. (2016a) but a summary is provided here. A two-stage stochastic programming model has been developed for generating timetables with the required number of services at the tactical level. Different recourse actions to recover from delays are taken into account at the operational level (e.g., speeding up trains). The model considers conflicts among different types of trains (e.g., express and freight trains) at different locations (e.g. points, junctions, and platforms).

Small instances can be solved by commercial solvers; however, for solving large instances, we developed a large neighbourhood search algorithm (LNS). In each iteration, the algorithm executes two phases: in the first phase, a feasible order among trains is determined; given this order, the reliability of the timetable is optimised in the second phase.

Train services are scheduled by a recursive job shop algorithm that is guaranteed to insert a service into a given timetable if a feasible insertion position exists. Appropriate buffer times are incorporated into the timetable by a greedy algorithm and linear programming in order to absorb uncertain delays.

More complicated recourse actions have been tested which include changing the platform assignments if a platform is blocked, and allowing trains to overtake if an express train is stuck after a regular train. However, our results suggest that considering complicated recourse actions can be avoided in the timetabling phase. This result remains to be verified on railway systems with large-scale delays.

The LNS has been tested extensively on benchmark instances. The results show that the algorithm is able to generate feasible timetables even when capacity constraints are tight. The solution quality increases with a larger number of iterations. The generated results are on average 6.6% worse than the best known solution; the average computation time is 4.1 hours. (Kovacs et al., 2016b)

The results of a case study appear to indicate, at a first-cut, that there is room for increasing the operational capacity at Peterborough. However, as the availability of rolling stock and staff, as well as shunting movements within the stations, have not been considered here, the results should be interpreted as a best case situation. Nevertheless, they suggest that it is possible to increase the capacity utilisation of the existing infrastructure by using state-of-the-art optimisation techniques, as opposed to alternative strategies that are significantly more expensive and involve reducing headway times (e.g., by updating the signalling system and/or improving the braking performance of trains) or laying new tracks.

For our timetable optimization modelling, our model consists of a network layout, a set of trains, and a set of delay scenarios. An example of a network layout is given in Figure 7. We consider several stations with different numbers of platforms, points, junctions, double track lines, quadruple track lines with fast and slow tracks in each direction and single track lines that are traversed in both directions.

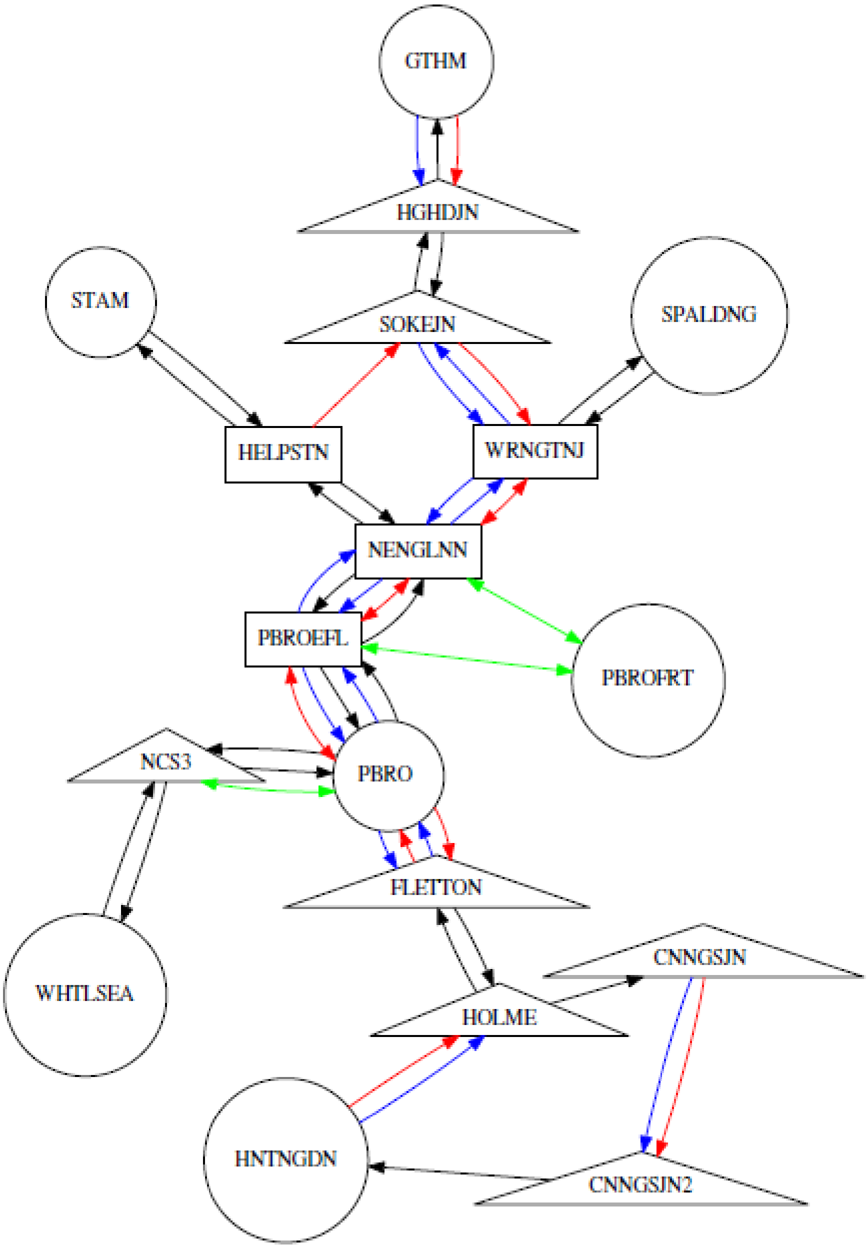


Figure 7: Example of the Peterborough network layout.

The set of trains travelling along the network is provided in the form of a timetable. For each train, we are given the route (i.e., a sequence of stations, junction, and points); preferred arrival and departure times at different locations; and the type of train (e.g., freight, express, or regular). Furthermore, a delay scenarios list is provided, where for each train, the duration and location of the delay is specified.

Our case study focuses at the rail network surrounding Peterborough station. The layout involves seven stations, four junctions, and seven points. The network comprises of 47 arcs, each arc representing a track segment that can either be a fast, slow, main, or freight track. Freight trains can be assigned to slow, main, and freight tracks; regular trains to slow and main tracks; and express trains to fast, slow and main tracks.

The set of trains is selected from a representative weekday (4/11/2015). From the national timetable, we select all passenger and freight trains (including empty locomotives) that visit Peterborough between 7am and 9am. In total, we consider 55 services in the reference timetable. The average speed of express, regular, and freight trains is assumed to be 125, 100, and 75mph, respectively. The time required for acceleration and deceleration is considered by decreasing the average speed by 7% if a given train has to stop once in our model, by 14% with two stops, and by 21% with at least three stops.

Delay information is gathered from historical delay data provided by Network Rail. More than 6 million delays were recorded between 1/12/2013 and 18/04/2015 (i.e., over 503 days). As primary delays are the model input, we filter out irrelevant information on secondary delays. Almost 800,000 delay records remain. The efficiency of the model algorithm is measured by its ability to mitigate delay propagation by incorporating proper buffer times into the timetable. The smaller the secondary delays, the better the objective value, and the better the solution.

In a second step, we match filtered trains (T) with trains in the delay data (D). There is no unique identifier that unambiguously links trains in the two sources of data. Therefore, we apply the following strategy. We take the set of relevant trains, the delay data, and a time margin TM. We then match T with D if: (i) T is a passenger train (delays of freight trains are not recorded); (ii) T and D have the same origin and destination; and (iii) T departs within the departure time of D +/- TM.

Delay scenarios are sampled in a Monte-Carlo fashion. In each scenario, and for each train, we decide by Bernoulli trial whether or not it is delayed; if yes, we associate the location and duration of the delay. The length of the delay is modelled by a Gamma distribution.

Our first-cut results are shown by Figure 8. Out of 200 timetables tested, 184 were feasible and this suggests that an additional 40 trains in the morning peak hour at Peterborough is feasible, although not necessarily desirable. This preliminary result would represent an increase in service of around 73% but an increase in an index of delays (as measured by the objective function (OV), which is a combination of maximum and mean delays) of around 144%. This represents a 3.64% increase in delays with each train.

Of these 40 additional trains, 18 will run to/from Grantham. Grantham is modelled as having 46 trains in the morning peak, so this would represent an increase in service of 30% at this node. This Figure involved over half a year of computing time (ran in parallel on the University of Southampton’s Iridis 4 supercomputer). The pattern of delay increases appears to follow a linear rather than an exponential function.

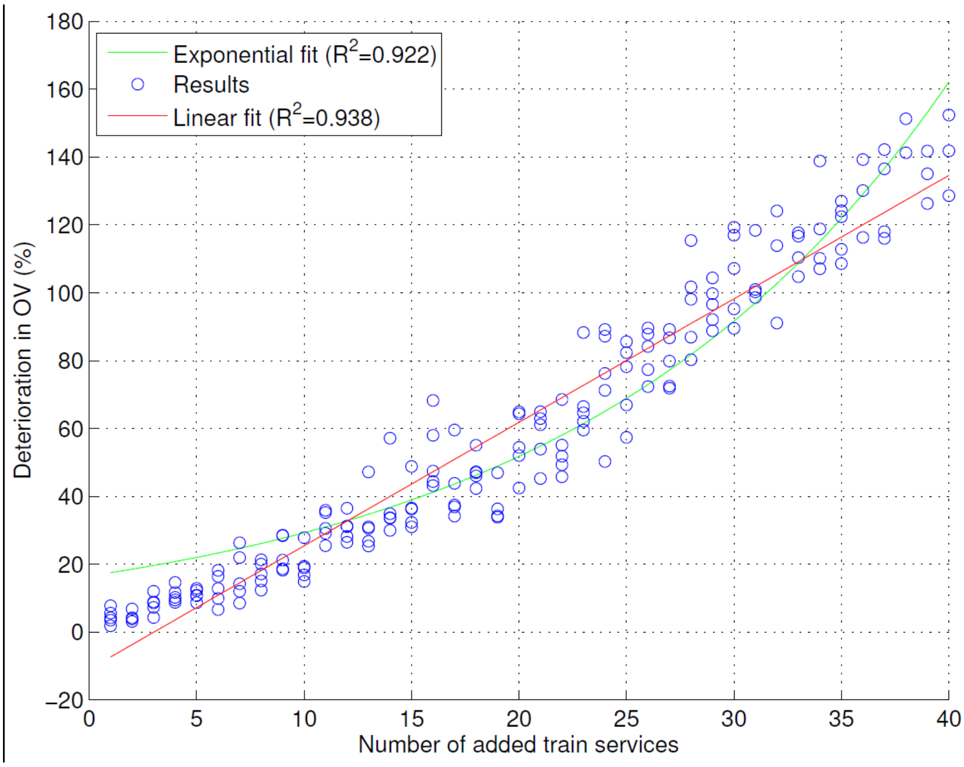


Figure 8: The Relationship between Additional Train Services and the Delay-based Objective Function at Peterborough station.

3.2 A Multi-Commodity Network Design Problem Application

Figure 8 suggests that Peterborough station (which was remodelled in 2014) is not a bottleneck. However, to confirm that these services are feasible we need to examine a larger area but a larger network will lead to higher complexity. Our first attempt to examine this issue was to treat it as a Multi-Commodity Network Design Problem (MCNDP), with the commodity being an individual service and the network being a time-space diagram where space refers to rail network layout. The design is then the trajectory of each service. This approach is widely used in logistics to make strategic decisions (Goetschalckx et al., 2002).

Complexity is reduced by aggregating data and simplifying constraints. However, some important aspects of railway operations have not yet been considered in the MCNDP such as no overtaking is permitted on the same track, there might be more than one track and the need for consistent headways.

In a second attempt a mathematical programming approach was adopted based on column generation (see, for example, Desrosiers and Lübbecke, 2005). Column generation algorithms are convenient when the number of variables in the linear programme is great. In our model, the variables are the possible trajectories that a service can be assigned to where each trajectory is a path through the time-space network. The goal is to schedule as many services as possible by assigning at most one trajectory per service. A prototype has been developed but the algorithm didn't work well enough to tackle the desired scope of the network and as a result the column generation approach, which has some novelty, was abandoned.

Our third and final attempt used a variant of the job shop algorithm as described above but the algorithm was simplified in order to solve a larger case study. In particular, uncertain delays and spread constraints are ignored and conflicts at large junctions are simplified. The focus is exclusively on generating a feasible timetable and delay propagation is ignored.

The network involves the main trunk of the East Coast Main Line between Alexandra Palace and Doncaster via Peterborough, but also includes branch lines to, for example, Cambridge, Leicester, Lincoln and Nottingham. In total, the modelled network consists of 35 stations, 17 junctions, 21 points, and 215 arcs and covers all trains from the national timetable on an average weekday between 7am and 9am. The travel times of the trains are taken from the timetable without any modification (e.g. they include travel time margins and scheduled waiting times). Trains that do not pass through Peterborough adhere to the timetable. Trains passing through Peterborough may be shifted by +/- 5 minutes. Headways are set to 3 minutes and reduced where needed to make the original timetable feasible. Trains to be added to the timetable are selected randomly, but have to pass through Peterborough. The number of added services is between 1 and 10. For each number between 1 and 10, we generate 5 timetables. Some preliminary results are shown by Table 4.

Table 4: Preliminary Results from the Wider Network Analysis

No. of Services Added	1	2	3	4	5	6	7	8	9	10
No. of Services Feasible	5	5	4	1	2	0	1	1	0	1

Table 4 indicate a high utilisation of the network, with only a relatively modest number of additional services feasible. However, there might be room for improvement by allowing more flexibility in the travel times and track occupation times. Although Peterborough is not a bottleneck in the classic sense, there may be bottlenecks elsewhere on the network, such as the Digswell (Welwyn) Viaduct.

4. Dynamic Simulation

ERTMS is expected to significantly increase railway capacity, train speed, reliability and punctuality (Wendler, 2007). The increased capacity means shorter train headways and more severe interference among trains using the same infrastructure, which may lead to longer delay and higher energy consumption. Therefore, it is important to investigate, in a practical train operating environment, whether the claimed performance improvement can be achieved. A railway traffic simulation platform can provide convenience for such an investigation before the ERTMS is widely implemented in practice. As a part of the DITTO project, the simulation platform could help to generate detailed train operation data for the capacity analysis of the railway network, and the analysis of time- and energy-efficiency of timetables generated by other scheduling tools.

ETCS (European Train Control System) is the component of ERTMS for signalling, train control and train protection. ETCS Levels 2 and 3 are the two most advanced application levels under the existing categorisation of ETCS, and they are distinguished from other lower levels in the way that they are radio-based. The most obvious difference between ETCS Level 2 and Level 3 is that, the former is a fixed-block system while the latter is a moving-block system.

We provided microsimulation platforms for simulating the key functions of ETCS Level 2/3, which works based on the interaction of the train, the radio block centre, the control centre, and other trackside equipment such as interlocking. The key factors represented in the simulation model include network characteristics, train characteristics, train timetable, traffic behaviour and control strategies.

Figure 9 illustrates a four station simulator of ETCS Level 2 developed for both a single line (left) and a line with passing loops at intermediate stations (right). The distance between stations is set at 30 km and is an approximation of the Retford-Newark-Grantham-Peterborough section of the East Coast Main Line (100 km). For both simulations a mix of fast trains (blue solid) and slower trains (red dash) are operated. A variant of this simulator for ETCS Level 3 has also been developed.

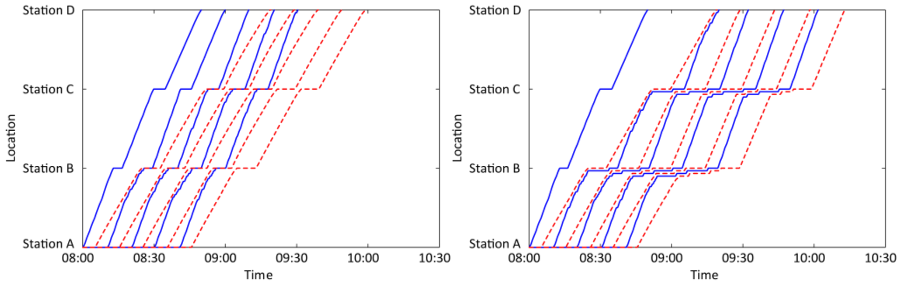


Figure 9: Four Station Simulator for a Single Track (Left) and a Single Track with Passing Loops (Right)

The capacity increase of ERTMS relies on not only the new radio systems for communication between trains and radio block centres, but also the intelligent traffic management and control strategies for, e.g., train following, train trajectory optimisation, moving authority generation and rescheduling. Also, with the real-time and detailed train running information, the movement authorities and schedules can be adjusted in a more sophisticated way, which introduces challenges for the real-time scheduling and train control algorithms for time and energy saving. Besides the establishment of the simulation platforms for ETCS Level 2/3, we have made progress in development of advanced train control under ETCS Level 2/3. Chen et al. (2016) developed a controlled train-following model to maintain the optimal speed and desired following headway in a train platoon, and discussed the impact of the parameters of such train-following model on the stability of the train following, where such impact would determine how a disturbance propagates along the train platoon. We also proposed new control models and algorithms for energy-efficient train driving, in both fixed-block system like ETCS Level 2 and moving-block system like ETCS Level 3, by taking into account speed limits, track gradients, maximum tractive/braking forces, and resistance (Ye and Liu, 2016, 2017). We considered multiple trains running simultaneously on a same railway track segment and allowed possible rescheduling at the intermediate stations on their journeys.

5. Conclusions

We have detailed work on the safety and capacity analysis of scheme plans, which may be used to optimise the network, on the application of a stochastic version of job-shop scheduling to optimise train plans and on the development of dynamic simulation models to optimise train management. We have shown how safety and capacity analysis can be undertaken using CSP and timed CSP respectively and also how safety analysis can be extended to ETCS Level 2. We have demonstrated that stochastic job shop optimisation can be applied to practical problems. We have shown how dynamic simulation and train control can be developed for ETCS Levels 2 and 3.

Our approach to integrating this work is given in Preston et al. (2016). In particular, our methods will be integrated through the use of state-of-the-art developments in computer science with respect to safety and capacity verification (and the resultant OnTrack tool) and in the use of artificial intelligence to provide real time train control. These self-learning tools will also be analogous to methods developed for road traffic (Box and Waterson, 2012).

From our case study of the ECML, it is found that additional services can be provided on the local network around Peterborough but at the expense of service performance. However, when a wider network area is considered, the number of feasible additional services appears to be much reduced. Our work is also developing to illustrate the gains that ERTMS level 2 may bring in terms of theoretical capacity and how that theoretical capacity can be used in practice. Further work will examine these issues in more detail, in particular to examine whether any of the feasible services are desirable from either a commercial or a social perspective building on work we presented at the previous conference (Preston et al., 2015).

Acknowledgements
This research was funded by RSSB and undertaken as project T1071-02**,** as part of the FuTRO (Future Traffic Regulation Optimisation) programme. Other investigators in DITTO include: Ronghui Liu and Tony Whiteing (Leeds); Tolga Bektas and Chris Potts (Southampton); and Markus Roggenbach and Faron Moller (Swansea)

References

Abril M., Barber, F., Ingolotti, L., Salido, M.A., Tormos, P. and Lova, A. (2008). An assessment of railway capacity, *Transportation Research Part E,* 44 774–806.

Armstrong, J. and Preston, J. (2015) *Node-based Capacity and Congestion Estimator Documentation.* DITTO Milestone 4. September.

Armstrong, J. and Preston, J (2017). Capacity Utilisation and Performance at Railway Stations. *RailLille2017*, Lille, France.

Bektas, T., Chouman, M. and Crainic, T.G. (2010) Lagrangean-based decomposition algorithms for multicommodity network design problems with penalized constraints. *Networks*, 55, (3), 171-180.

Bektas, T., Kovacs, A. and Potts, C. (2015) *Scenario Generation for Train Timetabling and Train Scheduling under Uncertainty.* DITTO Milestone 5. November.

Box, S. and Waterson, B. (2012) An automated signalized junction controller that learns strategies from a human expert. *Engineering Applications of Artificial Intelligence,* 25, (1), 107-118.

Chen, L., James, P., Kirkwood, D., Nguyen, H.N., Nicholson, G.L. and Roggenbach, M. (2016) Towards integrated simulation and formal verification of rail yard designs – An experience report based on the UK East coast mail line. IEEE International Conference on Intelligent Rail Transportation (ICIRT): 347-355.

Chen, J., Liu, R., Ngoduy, D. and Shi, Z. (2016) A new multi-anticipative car-following model with consideration of the desired following distance. *Nonlinear Dynamics* 85, 2705-2717.

Desrosiers, J. and Lübbecke, M. E. (2005) A Primer in Column Generation. In Desaulniers, G.; Desrosiers, J. & Solomon, M. M. (Ed.) *Column Generation*, Springer US, 1-32

Goetschalckx, M.; Vidal, C. J. & Dogan, K. (2002), Modeling and design of global logistics systems: A review of integrated strategic and tactical models and design algorithms, *European Journal of Operational Research ,* 143, 1 – 18.

Goodall, R., Dixon, R., Bemment, S., Fujiyma, T., Mei, T. Preston, J., Romanovsky, A. and Roggenbach, M. (2013) *The Future of Train Control Systems.* Technical Paper, Institution of Railway Signal Engineers, London, 13 February.

Isobe, Y., Moller, F., Nguyen, H.N. and Roggenbach, M. (2012) Safety and line capacity in railways. *Lecture Notes in Computer Science* 7321:54-68.

James, P., Moller, F. and Roggenbach, M. (2015a) *DITTO Tool Integration: Basic Prototypes for Interfacing.* DITTO Milestone 2. September.

James, P., Roggenbach, M. and Moller, F. (2015b) *OnTrack Documentation.* DITTO Milestone 1. June.

James, P., Lawrence, A., Roggenbach, M. and Seisenberger, M. (2015c) Towards Safety Analysis of ERTMS/ETCS Level 2 in Real-Time Maude. *Communications in Computer and Information* Science, 596, 103-120.

James, P., Moller, F., Nguyen, H.N., Roggenbach, M., Treharne, H. and Wang, X. (2016) Ontrack: The railway verification toolset (extended abstract). *Lecture Notes in Computer Science* 9953:294-296.

Kleywegt, A., Shapiro, A., and Homem-de-Mello, T. (2002) The Sample Average Approximation Method for Stochastic Discrete Optimization, *SIAM Journal on Optimization*, 12, 2, 479-502.

Kontaxi, E. and Ricci, S. (2012) Railway Capacity Handbook: A Systematic Approach to Methodologies. *Procedia - Social and Behavioral Sciences*, 48, 2689-2696.

Kovacs, A., Bektas, T. and Potts, C. (2016a) *Stochastic Optimisation.* DITTO Milestone 9. October.

Kovacs, A., Bektas, T. and Potts, C. (2016b) *Train timetabling and scheduling under uncertainty.* Technical Report, University of Southampton.

Liu, S.Q. and Kozan, E. (2009) Scheduling trains as a blocking parallel-machine job shop scheduling problem, *Computers & Operations Research* 36, 2840-2852.

Liu, R., Ye, H. and Whiteing, A. (2015) *Dynamic Simulation for Real-Time Operations of ERTMS Level 3.* DITTO Milestone 6. September.

Liu, R., Ye, H. and Whiteing, A. (2016) *Simulation and Control of ERTMS Level 2.* DITTO Milestone 7. September.

Moller, F. and Roggenbach, M (Eds) (2016) *Towards Tool Interaction and Safety Assessment.* DITTO Milestone 8. June.

ORR (Office of Rail and Road). (2016) *National Rail Trends.* http://dataportal.orr.gov.uk/ <Accessed 25 November 2016>

Paraskevopoulos, D., Bektas, T. and Potts, C. (2015) *Train Scheduling under Nodal Capacity Constraints.* Technical Report, University of Bath.

Preston, J., Kampantaidis, G. and Armstrong, J. (2015) The Trade-off between Delays and Capacity Utilisation: Observations from the UK *6th International Conference on Railway Operations Modelling and Analysis.* Tokyo, Japan, 23-26 March.

Preston, J., Armstrong, J., Kovacs, A., Liu, R., Ye, H., James, P., Moller, F. and Roggenbach, M. (2016) *Interim Report and Prototype.* DITTO Milestone 3. April.

RSSB (2012) *Rail Technical Strategy.* RSSB, London. http://www.rssb.co.uk/library/future%20railway/innovation-in-rail-rail-technical-strategy-2012.pdf <Accessed 20 November 2016>

Wendler, E. (2007) Influence of ETCS on Line Capacity. *UIC ERTMS World Conference* 11-13 September. Berne, Switzerland. http://www.uic.org/cdrom/2007/14_ERTMS_conference07/docs/presentations/G4_wendler.pdf <Accessed 25 February 2016>

Ye, H. and Liu, R. (2016) A multiphase optimal control method for multi-train control and scheduling on railway lines. *Transportation Research Part B*, 93, 377-393.

Ye, H. and Liu, R. (2017) Nonlinear programming methods based on closed-form expressions for optimal train control. *RailLille2017*, Lille, France.