Data-Driven Simulation and Optimization for Incident Response in Urban Railway Networks

Jácint Szabó
IBM Research Lab, Zurich
Säumerstrasse 4
8803 Rüschlikon, Switzerland
jsz@zurich.ibm.com

Sebastien Blandin
IBM Research, Singapore
10 Marina Boulevard
18983 Singapore
sblandin@sg.ibm.com

Charles Brett
IBM Research, Singapore
10 Marina Boulevard
18983 Singapore
cbrett@sg.ibm.com

ABSTRACT

In this article we introduce an integrated agent-based train, passenger and incident simulation engine for data-driven incident response in urban railway networks. We model the movement of passengers and trains as individual agents behaving according to parsimonious models defined by data availability. Appropriate statistical routines are implemented for model calibration. We also design a generic incident model appropriate for typical localized mechanical failure scenarios in which the transport supply is adversely impacted in a short spatio-temporal window. Given the brief and localized nature of these events, a mathematical programming formulation is proposed, which computes the optimal action plan for a specific incident. The set of action plans considered includes re-scheduling existing train services as well as running temporary services. The numerical performance of the simulation engine is presented using a large dataset of real anonymized smart card data. The results of the proposed optimization framework are then evaluated using real incident scenarios.

Keywords

Agent-based simulation, urban public transportation networks, mathematical optimization

1. INTRODUCTION

Recent estimates [17] of traffic externalities illustrate the magnitude of the congestion challenge facing medium and large cities. Increasing space constraints are limiting the construction of additional infrastructure, so prioritizing the use of more space efficient transport modes remains one of the most effective mechanisms available to curb congestion impact. In particular, soft approaches such as modal shift, focusing on increasing the share of public transport, have seen renewed interest in the recent years.

However, increased utilization puts a heavy strain on public transport networks, often resulting in more frequent mechanical failures, having negative short term impact on travel delay (see [9] for examples on the road network), and negative long term impact on the trust placed by commuters in the public infrastructure. Thus, there is high value in improving the real-time response to unexpected incidents that adversely impact the transport supply.

A lot of efforts have focused on agent-based simulation of transport networks, with some of the existing major commercial and academic solutions dating back to the 90's [2] [8]. Current simulators have been shown to scale to nationwide size [13], [20]. Due in large part to the lack of sufficient data for fine-grained calibration of a large set of parameters, agent-based simulators have traditionally been used for planning applications, such as transit modeling, network design and traffic assignment [18].

In the era of the smart card [14], in which the entry and exit points of passengers are recorded in near real-time, advanced data analytics can be designed to accurately generate, model and simulate the movements of passengers. In particular, data-driven and statistical methods have proven very efficient in the extraction of meaningful insights from large datasets [7], [10], [15], [19].

In contrast to simpler problems such as journey planning [3], [5], [12], [16], for which unified data-driven methodologies exist for network modeling, performance evaluation, and mathematical optimization, there remains a gap between the identification and qualification of various local transport inefficiencies using data-driven methods (see for instance [11] for bus bunching and [6] for reduced efficiency overall), and the use of other methods, either agent-based simulations or semi analytic methods, for scenario evaluation and optimization of the issues identified.

In this work, we design a tractable data-driven agent-based engine for real-time incident response in national railway networks. The engine includes calibration routines leveraging smart card data. The proposed solution is particularly relevant for online applications such as real-time train re-scheduling. Our engine consists of an agent-based simulation of train movements and passenger trajectories. Trains and passengers are considered as agents, moving at discrete time points in the network according to a pre-defined adaptive behavior model, and interacting with other agents such as network elements, trains and passengers.

We also propose a detailed model of an action plan in case of supply disruption, involving train re-scheduling and temporary services. An action plan is introduced as a response to an incident, for example a network disruption, to mitigate its impact on increased travel time and passenger crowd. The effectiveness of an action plan can be simulated and evaluated in real-time. The simulation engine also includes an optimization module which finds an optimal response to a specific incident given an objective function. The problem of

Appears in: *Proceedings of the 16th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2017), S. Das, E. Durfee, K. Larson, M. Winikoff (eds.), May 8–12, 2017, São Paulo, Brazil.*

finding an optimal action plan is formulated as a mixed integer program based on an underlying multi-commodity flow problem. Our solution is a special case of the timetabling problem [1], [21], for more specific and limited instances.

The main contributions of this work include:

- design and implementation of a fast agent-based simulator including statistical routines for parameter calibration,
- integrated optimization framework appropriate for real-time applications,
- evaluation of the simulation-optimization framework on a large smart card dataset and many incident scenarios.

The rest of this article is organized as follows. In Section 2 we detail the modeling assumptions underlying the simulation engine introduced in this work. Section 3 then introduces the optimization model corresponding to the optimal action plan problem. Section 4 describes calibration results, and Section 5 walks through case studies on incident scenarios. Finally, Section 6 presents concluding remarks.

2. TRAIN AND PASSENGER SIMULATION

In this section we present the mathematical models guiding the behavior of the simulated agents. In particular, we highlight the parsimonious modeling of the train speed profiles, the safety system, and the passenger route choice behavior, both for incident and no incident scenarios.

In the interest of space, we present the models in the context of offline simulations where the initial condition is a train and passenger free network, but the models presented naturally extend to an online setting where the initial condition contains trains and passengers already on their way.

2.1 Network and safety model

The underlying urban railway network is modeled in a mesoscopic manner as a multi-scale directed graph where locations such as stations, depots, turning points and platforms are modeled as nodes, and tracks are modeled as edges. Connectivity between locations and real distances are accurately modeled by the graph structure and link attributes, but the exact geometry and geographical layout of the physical objects is not taken in consideration. We also highlight that the graph is multi-scale in the sense that certain locations such as station platforms belong to parent locations, stations in this case.

In urban railway networks, every line usually has its own dedicated track for each direction, hence a simplified safety system is being used based on minimum headway times upon entering a track, and minimum headway space between trains along a track.

2.2 Train modeling

In the simulation engine, the public timetable is extended to encompass all network elements along the path of the train, including platforms, tracks, junctions and depots, resulting in an explicit specification of the network path that the train has to follow. Trains evolve along the tracks according to a discrete time dynamical system based on the train kinetic parameters and the tracks speed limit, see Figure 1. Trains attempt to run at the maximum speed that is allowed at the current location, under the constraint that their acceleration and deceleration bounds, and train and track speed limits are continuously respected. In particular, when approaching a station, the train starts to brake just in time to reach a full stop when entering the station. Trains try to follow the arrival and departure times prescribed in the timetable, under the constraints imposed by the movement dynamics and the safety system.

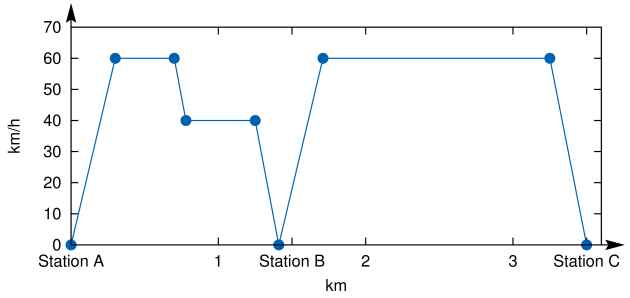

Figure 1: Example of a simulated speed profile of a train, constrained by location-specific speed limits.

The exact location of the available rolling stock is known, and is automatically dispatched to the trains. When the timetable indicates that a train has to be launched at a given starting station S, the closest available rolling stock is assigned to station S at the corresponding time. If the rolling stock is not located at S on the spot, it has to first move to S before the time indicated by the timetable. Rolling stock composition is not modeled, as usually along an urban railway line every train has the same composition and rolling stock is very rarely reshuffled.

2.3 Passenger behavior model

The simulation engine includes a hierarchical decision model wherein passengers first plan their route from their origin to their destination, and then evolve along their route either on board of a train or on foot according to a dynamical walking time model. The route from the current position to the destination with the minimum cost is chosen, along the following metrics: total travel time, total walking time, number of connections.

Given coefficients for these metrics, we define the cost of a route as the linear combination of the route's metric values with the coefficients. The coefficients used in the simulator are calibrated as described in Section 4. Typically good values for the coefficients are 1 both for total travel time and total walking time in minutes, and 2 for number of connections. For every origin-destination (O/D) pair the passenger simulation engine calculates the O/D route with minimum cost.

The passengers always recalculate their routes when they arrive at a platform or entrance or exit, either on foot or alighting from a train. This allows for updating the preferred path in case of a change in the traffic conditions, for instance at the onset of the peak time or when new temporary lines are launched during an incident.

Deterministic walking times between any pair of locations in a station (platforms and gates) are given as input to the simulator. The simulated walking time is a continuous piece-

wise affine function of the number of passengers currently at the platform or gate, see Figure 2.

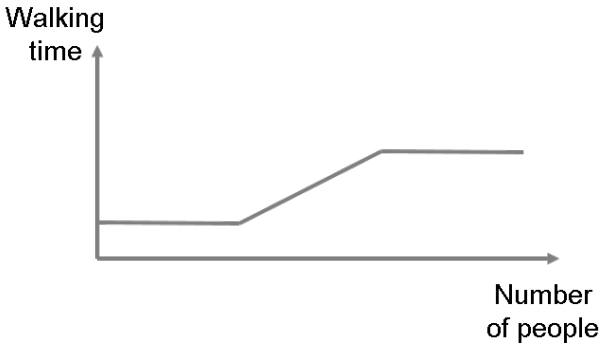

Figure 2: Between any pair of locations the walking time is constant under a certain crowd level, and then increases linearly up to a certain maximum value.

The walking time of an individual passenger is obtained from a random draw of a log-normal distribution with mean value corresponding to the walking time as shown in Figure 2. Further details on the calibration procedure for the walking time distributions are provided in Section 4.

2.4 Incident modeling

The main functionality of the solution introduced in this work is the simulation and management of incidents, such as network disruption, train breakdown or even a social event. Given an incident we simulate the resulting disturbance in train schedules and passenger flows, automatically generate action plans as a response and optimization mechanisms in order to achieve the best response.

An action plan consists of additional temporary train lines defined by their itineraries and frequencies. Any feasible train itinerary in the network may be considered as a candidate for the definition of additional specific train services in an action plan, for example a shuttle service running parallel to a disrupted track back and forth, see Figure 3. Similarly, any user-defined bus line connecting train stations are allowed, for example a shuttle service that stops at every train station along the disrupted section of the line. Frequencies of existing lines can also be changed.

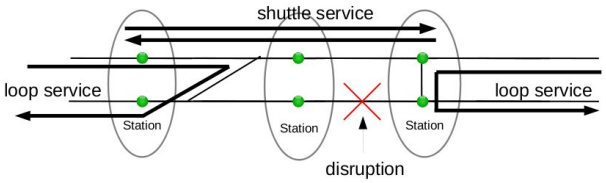

Figure 3: Example for candidate response lines in case of a disrupted track.

Action plans are simulated in real-time by the simulation engine, their adequacy are evaluated, and key performance metrics such as crowd levels and travel delays are compared against reference values. The incident management solution calculates a set of optimal action plans for the incident, using mathematical optimization (see Section 3).

3. OPTIMIZATION MODEL

In this section the problem of finding an optimal action plan is formulated as a mathematical program. The set of objective functions considered include linear combinations of the overall passenger delay, defined as the increment of the total travel times of the passengers relative to the no incident scenario, as well as the platform crowd levels.

The optimization constraints we consider include upper bound on the follow-up times for incident response train and bus lines, as well as number of trains and buses available for incident response services. The optimization module returns not only the optimal plan but also a subset of the best sub-optimal plans.

3.1 Modeling assumptions

A mathematical program that accurately models reality is often very complex, and cannot be directly solved by an optimization solver. In order to make the model tractable, especially for real-time applications, one usually needs to simplify the model. In this section we present specific modeling assumptions made for optimization, consistently with the simulation modeling assumptions described in the previous section.

We formulate the optimization problem as an extended version of a multi-commodity flow problem [1] over a space-time graph, which is then formulated as a *mixed integer program* (MIP), and solved by a mathematical programming solver.

Let us define the following quantities:

- a transportation network $G = (V, E)$ consisting of stations V and direct train and bus connections E between the stations,
- a collection of (train or bus) lines L, and for each line $l \in L$
  - an itinerary consisting of a sequence of stations $(v_l^1, v_l^2 \ldots)$, $v_l^i \in V$,
  - runtimes r_l^i for every leg i, that is between consecutive stations v_l^i, v_l^{i+1},
  - the number of services n_l of the line,
  - a common passenger capacity c_l of the services of the line,
  - the turnaround time τ_l, which is the total time for the services to travel on the line from start to end,
- a collection of aggregate passengers P, for $p \in P$ with
  - origin and destination stations o_p, $d_p \in V$,
  - start time s_p,
  - volume v_p.

For every triple (o, d, s) there is only one aggregate passenger p with (o_p, d_p, s_p) = (o, d, s), and this is the aggregation of all real passengers going from o to d starting at the discretized time s, with their total number as the volume v_p.

Abusing the standard definition of a frequency, we use frequency and follow-up times as synonyms, and so a frequency $\varphi = 2$ means one train every 2 minutes. Under the assumption that trains arrive steadily, i.e. between two consecutive trains the difference in arrival time is constant, the frequency thus reads $\varphi_l = \tau_l / n_l$.

3.2 Multi commodity flow problem

In order to solve the optimal action plan problem, we first convert the discrete transport supply provided by individual train and bus services to a continuous flow supply, and model the passenger movements as flows in a time expanded transportation network. In this space-time network the movement of a train along a leg uv with a runtime of r minutes will be represented by temporal copies of uv of time span r as follows.

Transforming service-based supply to continuous flow supply: let us replace every service of a line $l \in L$ with $1/\varphi_l$ new services with capacity $c_l \varphi_l$, as if trains were split into smaller equally sized trains with the same itinerary, running with a frequency of 1 minute. As an example, if the follow-up times of a line are 2 minutes, then every train of this line is replaced by 2 trains of halved capacity. This transformation preserves the total capacity of the line, and allows for adding a temporal copy of uv between times t and $t + r$, with r the runtime, for every integer minute time slot t, as described next.

Time expansion of the network graph: we define a simulation horizon $[0, T]$ with integer time-slots $0 \leq t \leq T$, $t \in \mathbb{N}$, for example $T = 120$ (minutes). We create a space-time supply network $G' = (N, A)$ and flow commodities as follows. For every station $v \in V$ and time slot t, N contains a node $n_{v,t}$ (representing the entrance gate of the station), and nodes $n_{v,t,l}$ for every line $l \in L$ (representing the platforms for these lines). As mentioned above, for every line $l \in L$ and leg $u = v_l^i$, $v = v_l^{i+1}$ in its itinerary with runtime $r = r_l^i$ we add *leg-arcs* $a = \overrightarrow{n_{u,t,l} n_{v,t+r,l}}$ for every time-slot t. Arc a has capacity $c_a = c_l \varphi_l = c_l n_l / \tau_l$ and length $h_a = r$.

Walking and waiting time modeling: for every station v, line l and time-slot t we add *waiting arcs* $\overrightarrow{n_{v,t} n_{v,t+1}}$, $\overrightarrow{n_{v,t,l} n_{v,t+1,l}}$ with length $h = 1$. We also add destination nodes z_v for every station $v \in V$, and *arrival arcs* $a = \overrightarrow{n_{v,t} z_v}$ with length $h_a = 0$. To model the walking times, we add *walking arcs* $a = \overrightarrow{n_{v,t,l} n_{v,t+w,l'}}$ with $h_a = w$ at every station $v \in V$, time-slot $t \in T$ and line pairs l, l', where w is the deterministic walking time in v between the platforms of l and l'. For $l = l'$ we take $w = 0$. We do the same for the walking between the gates and platforms by adding the walking arcs $\overrightarrow{n_{v,t} n_{v,t+w,l}}$, $\overrightarrow{n_{v,t,l} n_{v,t+w}}$ with length $h = w$ for every v, t and l.

Passenger demand: Finally, for every passenger $p \in P$ we add a commodity from n_{o_p, s_p} to z_{d_p} with demand v_p.

3.3 Mixed integer program formulation

The resulting fractional multi-commodity flow problem is presented below. Equations (2)-(3)-(4) are the flow conservation constraints, and (5) contains the service capacity constraints. The objective (1) is the total travel time of the passengers. Observe that the only reason why a passenger uses a waiting arc $a = \overrightarrow{n_{v,t,l} n_{v,t+1,l}}$ is that the service of the line l leaving from $n_{v,t,l}$ is full.

Taking n_l as a variable we get a mixed integer program solving the optimal action plan problem. We mention that with appropriate choice of the arc lengths other linear objectives can also be handled, for instance total travel time.

$$\min \sum_{p \in P,\, a \in A} f_{p,a} h_a \text{ s.t.} \tag{1}$$

$$\sum_{n' \in N} f_{p,n'n} = \sum_{n' \in N} f_{p,nn'} \;\forall\; n \neq n_{o_p,s_p}, z_{d_p}, p \in P \tag{2}$$

$$\sum_{n' \in N} f_{p,n'n} = v_p \text{ for } n = n_{o_p,s_p} \;\forall\; p \in P \tag{3}$$

$$\sum_{n' \in N} f_{p,nn'} = -v_p \text{ for } n = z_{d_p} \;\forall\; p \in P \tag{4}$$

$$\sum_{p \in P} f_{p,a} \leq c_l n_l / \tau_l \;\forall\; \text{leg-arc } a = \overrightarrow{n_{u,t,l} n_{v,t+r,l}} \in A. \tag{5}$$

3.4 Runtime optimization

The number of integer variables $|L|$ is usually very low. However, the number of fractional variables is large, on the order of:

$$\# f_{p,a} = |P| \cdot |A| \approx (|V|^2 |T|) \cdot (|E||T|). \tag{6}$$

where we recall that P denotes the number of passengers; V, E the number of vertices and edges, respectively, in the static network graph; T the number of time steps; and A the number of edges in the time-expanded graph. For a typical network the resulting mixed integer program is intractable.

In order to make the problem tractable, we introduce the following steps. First, we make a further aggregation step in the passengers' level. Above we aggregated the passengers with same origin $o \in V$, destination $d \in V$ and starting time s triple. Now we do the same for the origin o, time s pairs. In exact terms, for each commodity source $n_{o,t} \in N$ we aggregate the collection of $n_{o,t} \to z_d$ commodities where $d \in V$ to one single "tree commodity" from $n_{o,t}$, requiring that $n_{o,t}$ has as net outflow the number of passengers going from o to any destination in the network starting at time t; and the destination nodes z_d have as net inflows the number of passengers going from o to d starting at time t.

Second, for such a tree commodity only a subset of the arcs $a \in A$ can carry flow, so we define temporal corridors between $n_{o,t}$ and the z_d's for which non-zero flow values are allowed. This corridor consists of those arcs that the passenger may traverse assuming a route that is at most twice as long as the shortest route. This method drastically reduces the number of fractional variables, and makes the problem solvable by commercial optimization solvers such as CPLEX.

Thanks to these two improvements, the number of variables can be reduced from (6) to

$$\# f_{p,a} = |P| \cdot |A| \approx (|V||T|) \cdot (|E|\tau), \tag{7}$$

where τ is the maximum length of a shortest route.

We mention that it would also be possible to model the waiting times by increasing the walking time w with the expected waiting time y_l for line $l \in L$. Clearly, $y_l = 1/(2\varphi_l) = \tau_l/(2n_l)$. This function is non-linear, but convex in n_l, and we could take linear sub-gradients to bound y_l. As n can only have integer values, if B is an upper bound on n_l, only B sub-gradients suffice. So for every line we would have one new variable and B new constraints to add.

We remark that the first-come-first-served rule at the stations between the passengers is not considered. However,

this shortcoming only affects the individual passenger travel times, and the total travel time of the passengers is accurately modeled.

The solution to the MIP is used as a starting point for a local search performed using the simulation engine. The runtime improvements made to the MIP enables spending more computational resources on the local search.

4. EVALUATION

In this section we discuss the calibration procedure and present experimental results on the accuracy of the simulation engine in terms of passenger travel times.

4.1 Implementation

The simulation engine is implemented in C++ for performance reasons. A careful software performance optimization design, including efficient custom data structures yield a runtime performance of about 5000× real-time (i.e. simulating 5000 hours would take about 1h) for a typical city on a standard desktop machine. A simple user interface presenting key performance indicators is presented in Figure 4.

Figure 4: User interface showing some key performance indicators produced by the simulation engine: origin-destination demand, platform and train crowds.

4.2 Data sources

We consider a large regional urban railway system consisting of around 100 train stations, 10 train lines, and 2 million daily passenger travels (for confidentiality we anonymize the city and the stations). The network topology and station model are generated from machine-readable data sources and images providing connectivity information.

The passenger demand is provided by a 6 months dataset of anonymized smart card data, including the following data fields for every trip; anonymized identifier, (2) origin station, (3) destination station, (4) entry time, (5) exit time. Train movements from the same period are obtained from actual timetable data for the entire network.

Knowledge and details of the incident are constructed from a complete list of incidents in the railway system during the same period. While location of the incident and start and end time of the incident are always populated, the level of details regarding the actual response plan can vary, but often includes the number of resources used.

4.3 Calibration procedure

In this section we describe the calibration procedure for commuter walking time, which is one of the key simulation parameters. Indeed, a change in the walking time from the gate to the platform can cause simulated passengers to board a different train, which has immediate impact on the travel-time and platform crowd metrics. We calibrate the walking time based on smart card data and actual train trajectories.

The calibration procedure relies on inference of the train from which a given tap-out record alighted. This inference is possible when there is a dominant route between an origin and a destination, and the walking time from platform to exit gates is much less than the time between consecutive train arrivals. In this case, the on-arrival incoming direction of the passenger is derived as the minimal cost error, and a passenger can be associated with the latest train to arrive from the correct direction before the tap-out time. The difference between the tap-out time and the train arrival time provides a walking time sample. We then fit a log-normal distribution to the sample walking time distribution obtained for every station platform, for many trains and many days. A histogram of the walking times from a platform to an exit gate for one station is presented in Figure 5.

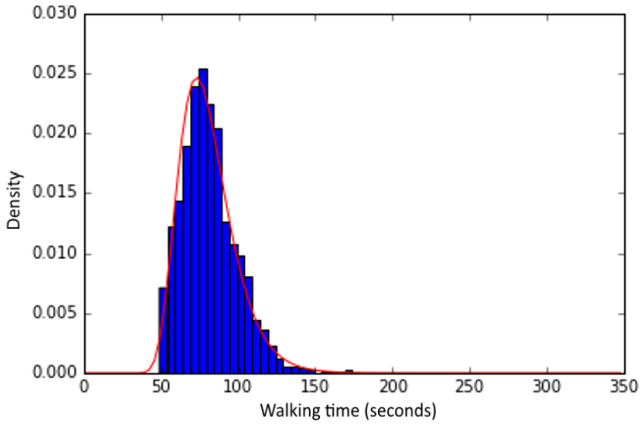

Figure 5: Histogram of walking times (in seconds) obtained from train arrival times and smart card data, with log-normal fit (red).

The calibrated walking time mean can then be used to fit with L^2 regression the piecewise affine walking time model illustrated in Figure 2. The fitted piecewise model is used for the platform to gate travel-time. Close parameter values are used for walking time between other station locations (gate to platform, platform to platform, platform to corridor, etc.).

4.4 Numerical results

To measure the accuracy of the simulator we consider the mean average error (MAE) and the mean relative error (MRE) between the simulated and the smart card based travel-times of the passengers. We consider half of the dataset as training set and the other half as test set.

We also measured the Bhattacharyya coefficients [4] as an aggregated exit time error, because contrary to MAE and MRE, this metric is oblivious to individual passengers with

Table 1: Travel time error depending on walking time parameters.

platform to platform (far)	platform to platform (close)	gate to platform	MAE (min)	MRE (%)	avg BC coeff.
4 min	1 min	2 min	4.9	18	0.93
4 min	0 min	2 min	5.5	21	0.91
4 min	1 min	1 min	5.7	21	0.91
3 min	1 min	2 min	4.8	19	0.93

Table 2: Travel-time error depending on number of line changes.

passenger set	MAE (min)	MRE (%)	avg BC coefficient
all	4.9	18	0.93
within one line (avg)	2.5	19	0.95

the same origin, destination, and start time triple. We used the Bhattacharyya coefficient to measure the distance of the real and historical exit time distributions. In details, for every origin – destination pair O/D and start time bucket $[t, t + 20\text{min})$ we collect those passengers going from O to D who start their travel in the time interval $[t, t + 20\text{min})$. We take the simulated and the historical travel times for this group, smooth them with the kernel $N(0, 3\text{min})$, and calculate the Bhattacharyya coefficient between these distributions. Finally, we take the weighted average of the coefficients across the groups.

We present in Table 1 the error metrics corresponding to specific parameter values for the walking time model. In the interest of space, the values reported in the Table include only the walking time in low crowd conditions.

The errors are presented in Table 2 for passengers traveling within a single line and for passengers traveling across multiple lines. The latter group is expected to exhibit a larger error since the connection time is an additional source of error.

We illustrate in Figure 6 a typical comparison output in terms of tap-out counts. The simulated values correctly capture the trend of the ground-truth tap-out data, but several limitations can be observed. First the simulated values lack some of the variability inherent to true smart card data and appear as a smoothed version of the actual data, effectively because all agents are modeled as rationally minimizing their travel cost. The performance of the simulator output also reduces significantly in the case of large variations, illustrated here for the afternoon peak, where the simulated peak appears reduced and shifted compared to the true peak.

5. SCENARIO ANALYSIS

In this section we analyze two real historical incidents. We compare the actual action plan deployed in reality to alternative action plans produced by the simulation-optimization engine, in order to demonstrate the capability of the incident management module to offer effective action plan options for incident handling. For evaluation of the effectiveness of a given action plan, we consider the following key performance indicators, motivated by actual business metrics:

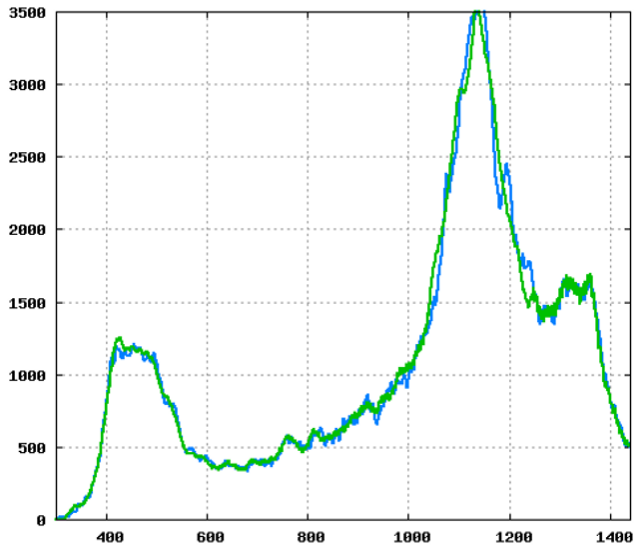

Figure 6: Tap-out count per 10 min, based on the smart card dataset (blue) and as simulated (green) assuming knowledge of passenger start time and destination station.

- **average delay (min):** Average delay in minutes over all affected passengers (delay greater than 5 minutes).
- **delay ≥20min (#):** Number of passengers with a delay of at least 20 minutes (we take 20 minutes as the threshold where the delay gets unacceptable to passengers).
- **overcrowding (min):** Total duration of overcrowding events at stations, in minutes, where overcrowding means crowd above 90% of station capacity.

The additional buses in the action plans refer to additional shuttle buses which are deployed during incidents, and operate on incident-specific routes, in addition to normal bus services operating on regular routes in the public transportation system. As illustrated in the results produced by our simulation optimization engine, quite often, a significant benefit in terms of resource usage maximization and travel delay reduction would be obtained by making the incident-specific routes also demand-specific, i.e. tailored to the impacted origin-destination pairs, and not only to the impacted tracks location.

5.1 Incident Scenario 1

For this real incident, the service of a line AB..C...X between Stations A and B is disrupted in both directions due to an infrastructure fault, see Figure 7 (we only represent 4 key stations and not all the stations on the line).

The disruption starts at 5.30AM and full service resumes at 3PM once the infrastructure fault is resolved. In reality service continues to run between Stations B and X during the incident with the normal frequency of 3 minutes of the line. There is no train service between Stations A and B, hence 15 bus shuttle services are in operation during this time period.

The base plan used to respond to the incident, as well as a subset of illustrative plans considered by the optimizer,

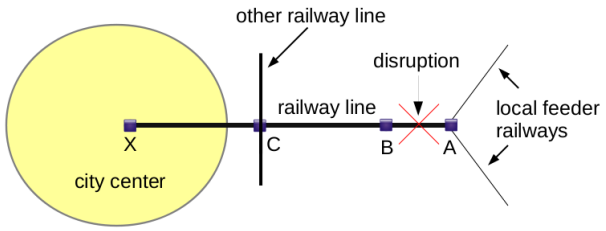

Figure 7: Incident Scenario 1., 5.30AM - 3PM

Table 3: Action plan (AP) comparison for Incident Scenario 1.

	base	AP1	AP2	AP3
train service	X ↔ B $\varphi = 3$	X ↔ B $\varphi = 3$	X ↔ B $\varphi = 2.5$	X ↔ B $\varphi = 3$, C ↔ B $\varphi = 6$
# additional trains	0	0	4	10
shuttle bus service	B ↔ A $\varphi = 2$ 15 buses	B ↔ A $\varphi = 1$ 30 buses	B ↔ A $\varphi = 2$ 15 buses	B ↔ A $\varphi = 2$ 15 buses
average delay (min)	11	6	11	11
delay ≥ 20 min (#)	4400	900	4000	3500
overcrowding (min)	60	10	50	40
max crowd in bus stop	2500	300	2500	2500

are presented in Table 3. Action Plan 1 (AP1) proposes to decrease the bus shuttle service follow-up times compared to the base bus plan; Action Plan 2 (AP2) proposes to decrease the train service follow-up times along the entire impact line; and Action Plan 3 (AP3) proposes to create a non-standard train service serving the vicinity of the impacted area between Stations C and B, in addition to the regular train service on the entire line. In the base plan there are significant delays between Stations A and B in the morning peak, because the deployed bus capacity is not enough to carry all passengers. For all action plans with bus follow-up times of 2 minutes it holds that from Station A the average delay compared to the no incident case is more than 30 minutes due to the high upstream demand in the morning peak.

We populate the mixed integer problem (1) of Section 3 in our optimization module with several bus line options between Stations A, B, C and the city center. The optimization module found that the optimum solution consists of the bus line B ↔ A with a frequency of only 1 minute. With this bus line it is possible to completely disperse the crowd at Station A, and also the average delay is reduced to only 6 minutes, see Figure 8. This value of the delay is close to the minimum physically possible.

A secondary effect of changing the bus frequency from 2 to 1 minute is that the maximum crowd in the bus stop at Station A decreases from 2500 to 300 (Figure 9).

AP1 is not viable if the number of available buses is below 30, so we also present the optimal action plans AP2

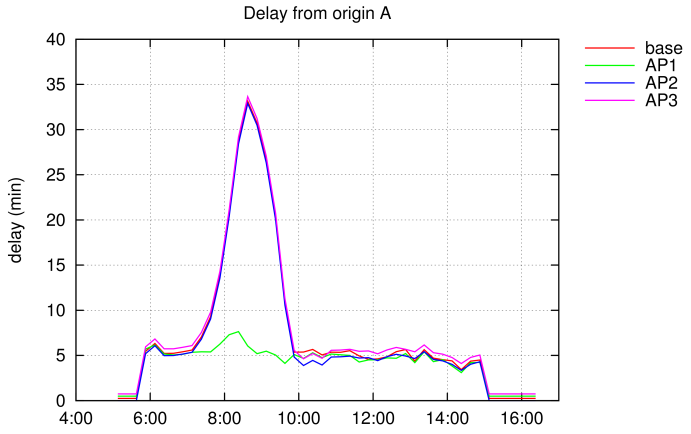

Figure 8: Delay from Station A towards X in Incident Scenario 1. Maximum delay of action plans "base", AP2 and AP3 are ≥ 30min, for AP1 it is ∼6min.

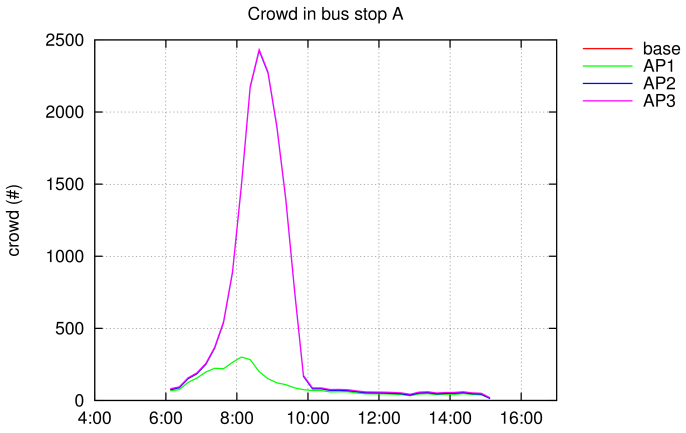

Figure 9: Crowd in Station A bus stop in Incident Scenario 1. Maximum crowd of action plans "base", AP2 and AP3 are ∼2500, for AP1 it is ≤ 400.

and AP3 obtained by the optimization engine in the case of constrained bus resources at most 15.

In the absence of sufficient buses for absorbing the supply-demand gap between Station A and Station B, the delay cannot be further reduced by adjusting train services. However, train services can be adjusted to better manage the incident from the perspective of the other objectives such as minimizing overcrowding time. During the incident, the crowd increases in Station B because passengers coming from Station A via bus, transfer there to train. This crowd can be reduced with a more frequent train service (AP2), and even more with the C ↔ B loop line (AP3), because more trains disperse the crowd faster.

However, the simulator detects an unexpected side effect of AP3. During morning peak the frequency of the X ↔ B long loop is 3 minutes, and that of the additional C ↔ B short loop is 6 minutes. The joint frequency then is 2 minutes, which is just the minimally required headway time. This means that every third train is a short loop train, and

Table 4: Action plan (AP) comparison for Incident Scenario 2.

	base	AP1	AP2
train service	C ↔ X $\varphi = 4$	C ↔ X $\varphi = 4$	C ↔ X $\varphi = 4$, D ↔ X $\varphi = 8$
# additional trains	0	0	10
shuttle bus service	B ↔ C $\varphi = 2$ 12 buses	B ↔ C $\varphi = 1$ 24 buses	B ↔ C $\varphi = 2$ 12 buses
average delay (min)	7	7	7
delay $\geq$ 20 min (#)	300	280	200
overcrowding (min)	20	19	10

from Station C only the long loop trains continue to run downstream to X, with alternating follow-up times of 2 and 4 minutes. In turn, this results in a bunching of trains, because every second train carries almost twice as many passengers, causing even more imbalanced follow-up times and overcrowding events down the line. This shows the importance to balance follow-up times.

5.2 Incident Scenario 2

This incident takes place during morning peak hour commute. A train fault disrupts service between Stations B and C in both directions along a line A..B..C..D...X (we only represent 5 key stations on the line), see Figure 10. The disruption starts at 8AM and lasts only 10 minutes. The response plan activated consists of free shuttle bus service between Station B and C with a frequency of 2 minutes.

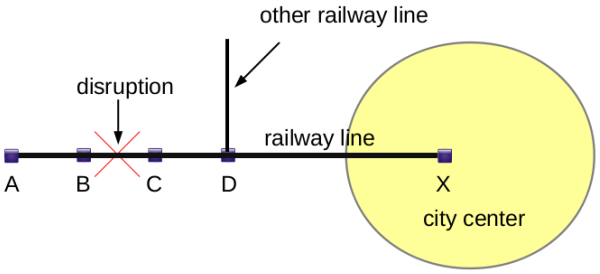

Figure 10: Incident Scenario 2.

With the deployed plan around 1500 passengers are stranded between Station B and Station C during the incident because the shuttle bus service cannot absorb all the demand. At the end of the incident these passengers return from the bus stops to the train stations between B and C, according to the passenger behavior model from Section 2.3. This large post-incident travel demand leads to large crowds in the station and some full trains, until crowds are finally dispersed around 8.45AM.

With more frequent buses in AP1 there is a slight reduction in stranded passengers, and so reduced post-incident impact, but this is very limited as the short duration of the incident does not allow shuttle bus service to have significant effect. The optimal action plan AP2 returned by the optimization module employs additional trains along D ↔ X. With this additional temporary loop line, the stranded crowd is more effectively dispersed in the busy interchange Station D.

As we observe by comparing AP1 with the real action plan, buses are of limited help in this incident. The reason is that the demand from B is too high to be managed by buses, and in addition due to the short duration of the incident, for many buses the incident is over before they arrive at Station C.

6. CONCLUSION

In this article we introduced a discrete time train and passenger simulation engine for urban railway networks. We described the solution architecture as well as the calibration procedures supporting fast and accurate simulation, and presented numerical experiments on a large real world smart card dataset illustrating the accuracy of our simulation engine. We also presented a mixed integer programming formulation for the problem of finding an optimal action plan as a response to a localized spatio-temporal incident. We analyzed several real incident scenarios, and quantitatively illustrated the performance of the simulation-based optimization framework. While the results presented in the context of real-time applications improve on traditional offline planing calibration and evaluation results, significant effort is still needed to more closely simulate seemingly random behavior of certain commuters, which can significantly impact network-wide metrics.

REFERENCES

[1] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin. *Network flows: theory, algorithms, and applications*. Prentice hall, 1993.

[2] M. Balmer, K. Meister, M. Rieser, K. Nagel, and K. W. Axhausen. *Agent-based simulation of travel demand: Structure and computational performance of MATSim-T*. ETH, Eidgenössische Technische Hochschule Zürich, IVT Institut für Verkehrsplanung und Transportsysteme, 2008.

[3] H. Bast, E. Carlsson, A. Eigenwillig, R. Geisberger, C. Harrelson, V. Raychev, and F.Viger. Fast routing in very large public transportation networks using transfer patterns. In *Algorithms–ESA 2010*, pages 290–301. Springer, 2010.

[4] A. Bhattacharyya. On a measure of divergence between two multinomial populations. *Sankhyā: The Indian Journal of Statistics*, pages 401–406, 1946.

[5] P. Borokhov, S. Blandin, S. Samaranayake, O. Goldschmidt, and A. Bayen. An adaptive routing system for location-aware mobile devices on the road network. In *IEEE Intelligent Transportation Systems Conference, Washington, D.C., October 2011*, 2011.

[6] M. Cepeda, R. Cominetti, and M. Florian. A frequency-based assignment model for congested transit networks with strict capacity constraints: characterization and computation of equilibria. *Transportation Research Part B: Methodological*, 40(6):437–459, 2006.

[7] N. de Lara and S. Blandin. Sparsely observed agent-based systems: a generative model for instantaneous crowd modeling. In *International*

Conference on Autonomous Agents and MultiAgent Systems, Singapore, May 2016, 2016.

[8] M. Fellendorf. VISSIM: A microscopic simulation tool to evaluate actuated signal control including bus priority. In *64th Institute of Transportation Engineers Annual Meeting, Dallas, TX*, 1994.

[9] Y. He, S. Blandin, L. Wynter, and B. Trager. Analysis and real-time prediction of local incident impact on transportation networks. In *Data Mining Workshop (ICDMW), 2014 IEEE International Conference on*, pages 158–166. IEEE, 2014.

[10] T. Kusakabe, T. Iryo, and Y. Asakura. Estimation method for railway passengers train choice behavior with smart card transaction data. *Transportation*, 37(5):731–749, 2010.

[11] D.-H. Lee, L. Sun, and A. Erath. Study of bus service reliability in Singapore using fare card data. In *12th Asia-Pacific Intelligent Transportation Forum*, 2012.

[12] T. Nonner. Polynomial-time approximation schemes for shortest path with alternatives. In *Algorithms–ESA 2012*, pages 755–765. Springer, 2012.

[13] T. Osogami, T. Imamichi, H. Mizuta, T. Morimura, R. Raymond, T. Suzumura, R. Takahashi, and T. Ide. IBM Mega traffic simulator. *IBM Res., Tokyo, Japan, IBM Res. Rep. RT0896*, 2012.

[14] M.-P. Pelletier, M. Trépanier, and C. Morency. Smart card data use in public transit: A literature review. *Transportation Research Part C: Emerging Technologies*, 19(4):557–568, 2011.

[15] H. Poonawala, V. Kolar, S. Blandin, L. Wynter, and S. Sahu. Singapore in motion: insights on public transport service level through farecard and mobile data analytics. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining, San Franciso, CA, August 2016*, 2016.

[16] S. Samaranayake, S. Blandin, and A. Bayen. A tractable class of algorithms for reliable routing in stochastic networks. *Transportation Research Part C: Emerging Technologies*, 20(1):199–217, 2012.

[17] D. Schrank and T. Lomax. The 2009 urban mobility report. *Texas Transportation Institute*, College Station, TX, 2012.

[18] H. Spiess and M. Florian. Optimal strategies: a new assignment model for transit networks. *Transportation Research Part B: Methodological*, 23(2):83–102, 1989.

[19] L. Sun, D.-H. Lee, A. Erath, and X. Huang. Using smart card data to extract passenger's spatio-temporal density and train's trajectory of MRT system. In *Proceedings of the ACM SIGKDD International Workshop on Urban Computing*, pages 142–148. ACM, 2012.

[20] T. Suzumura, S. Kato, T. Imamichi, M. Takeuchi, H. Kanezashi, T. Ide, and T. Onodera. X10-based massive parallel large-scale traffic flow simulation. In *Proceedings of the 2012 ACM SIGPLAN X10 Workshop*, page 3. ACM, 2012.

[21] P. Thorlacius, J. Larsen, and M. Laumanns. An integrated rolling stock planning model for the copenhagen suburban passenger railway. *Journal of Rail Transport Planning & Management*, 5(4):240–262, 2015.