Available online at www.sciencedirect.com

ScienceDirect

Transportation Research Procedia 52 (2021) 123–130

Transportation Research Procedia

www.elsevier.com/locate/procedia

23rd EURO Working Group on Transportation Meeting, EWGT 2020, 16-18 September 2020, Paphos, Cyprus

Flows Over Time as Continuous Limits of Packet-Based Network Simulations[1]

Theresa Ziemke[a,b,*], Leon Sering[b], Laura Vargas Koch[c], Max Zimmer[b], Kai Nagel[a], Martin Skutella[b]

[a] *Technische Universität Berlin, Transport Systems Planning and Transport Telematics, Salzufer 17-19, 10587 Berlin, Germany*
[b] *Technische Universität Berlin, Combinatorial Optimization and Graph Algorithms, Straße des 17. Juni 136, 10623 Berlin, Germany*
[c] *RWTH Aachen University, Management Science, Kackertstraße 7, 52072 Aachen, Germany*

Abstract

This study examines the connection between an agent-based transport simulation and Nash flows over time. While the former is able to represent many details of traffic and model large-scale, real-world traffic situations with a co-evolutionary approach, the latter provides an environment for provable mathematical statements and results on exact user equilibria. The flow dynamics of both models are very similar with the main difference that the simulation is discrete in terms of vehicles and time while the flows over time model considers continuous flows and continuous time. This raises the question whether Nash flows over time are the limit of the convergence process when decreasing the vehicle and time step size in the simulation coherently. The experiments presented in this study indicate this strong connection which provides a justification for the analytical model and a theoretical foundation for the simulation.

Keywords: dynamic traffic model; Nash flows over time; agent-based transport simulation; user equilibrium; Braess network

1. Introduction

Modeling traffic is an essential but difficult task for which a lot of different approaches have been developed in the last decades. This study highlights the strong connection between large scale simulations and mathematical models and compares their strengths and weaknesses. MATSim, a large-scale, agent-based simulation tool, is widely used to model real-world traffic scenarios in reasonable time (see Horni et al., 2016). To achieve this, MATSim simplifies real-world traffic by discretizing time and aggregating vehicles, and in addition, the co-evolutionary algorithm computes

[1] Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy – The Berlin Mathematics Research Center MATH+ (EXC-2046/1, project ID: 390685689).

* Corresponding author. Tel.: +49-30-314-78783 ; fax: +49-30-314-26269.
E-mail address: tziemke@vsp.tu-berlin.de

10.1016/j.trpro.2021.01.014

an approximated user equilibrium only. On the mathematical side, traffic can be modeled with flows over time (FOT). The FOT model yields exact user equilibria, called *Nash flows over time* (see Koch and Skutella, 2011). These Nash flows over time are guaranteed to exist (Cominetti et al., 2015) and their structure has been studied intensively, but there is no efficient algorithm known for their computation. This mathematical model is simplified in several ways and it can only represent basic traffic features.

While the link dynamics in the FOT model and in MATSim are almost the same (see Sec. 2), the big difference is that MATSim discretizes time and aggregates vehicles, whereas the FOT model considers continuous time and continuous flow. Even though real-world traffic consists of non-splittable vehicles, continuous flows describe average traffic rates. This study shows that, from a stochastic point of view, the discrete MATSim model can be interpreted as a realization of a random experiment where the average of the distribution is given by a Nash flow over time. To confirm this strong connection, this study analyzes the discretization error by decreasing the time step and vehicle size within MATSim. The experiments provide a justification for the FOT model and a theoretical foundation for MATSim. Structural insights into Nash flows over time can thus be transferred to user equilibria in MATSim. The analysis is based on experimental studies for illustrative but meaningful scenarios and for a small instance of a real-world scenario. Hereby, the focus is mainly on the convergence from discrete to continuous time steps and vehicles sizes and to a lesser extend on the convergence of the user equilibrium within the iterative process of MATSim.

Related work can be found on the theoretical side with similar convergence examinations in static routing games (see Haurie and Marcotte (1985); Scarsini et al. (2018)). In the dynamic setting, the influence of decreasing time steps was considered for an optimization problem (see Fleischer and Skutella (2007)) but not yet for routing games.

The paper is organized as follows: Sec. 2 introduces and compares the two models. Sec. 3 describes the experiments and presents the results. Finally, Sec. 4 concludes the findings and gives an outlook on further research directions.

2. The Models

2.1. Multi-Agent Transport Simulation (MATSim)

In MATSim, the road network is represented by a directed graph, where each link is equipped with a free flow transit time and an outflow capacity. In order to model spillback effects, a storage capacity per link determines the number of vehicles that fit onto that link. Streets (i.e., links) are modeled as first-in-first-out (FIFO) waiting queues. That is, a vehicle that enters a link immediately lines up and its earliest exit time is set to the entrance time plus free flow transit time. In each time step, vehicles are allowed to leave the queue, if (1) they are at the front of the queue, (2) their earliest exit time is reached, (3) the flow capacity of the link is sufficient, and (4) the next links have sufficient space left. When a vehicle leaves a link, its size is subtracted from the remaining flow capacity for this time step. If a positive amount remains, another vehicle is allowed to leave the link. Otherwise, the link's flow capacity has to accumulate over the following time step(s) before the next vehicle can go. When no vehicle wants to leave the link for some time, the flow capacity does not accumulate more than its value per time step, i.e., flow capacity cannot be saved for the future.

To be able to better compare the underlying models in this study, two adaptions have been made to MATSim's node transition that differ from the standard MATSim set up. The first adaption ensures that nodes (i.e., intersections) are blocked when at least one outgoing link gets full. This means that spillback affects crossing traffic more strongly than in standard MATSim, where nodes are only blocked once from each incoming link vehicles want to enter the full link. The second adaption concerns the order in which vehicles from multiple links are merged when they want to enter the same link. When vehicles are moved over a node in standard MATSim, incoming links are chosen randomly proportional to their outflow capacity; a selected link is allowed to send as many vehicles as the link's outflow capacity and the respective downstream link's storage capacities allow. With the new adaption, incoming links are chosen exactly proportional to their outflow capacity; when a link is selected, it is only allowed to send the first vehicle but might be selected again in the same time step. Thereby, MATSim's node transition becomes deterministic and vehicles from different incoming links are merged more evenly into downstream links.

MATSim uses a co-evolutionary algorithm, i.e., an iterative process where in each iteration a fraction of agents is allowed to change their plans by choosing from a set of good responses with the goal to improve their score. This procedure leads to a state where most of the agents do not have any incentive to deviate, but this does not necessarily

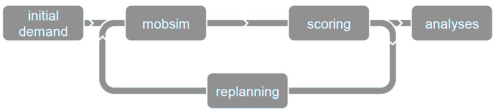

Fig. 1. Iterative cycle of MATSim (Horni et al. (2016)).

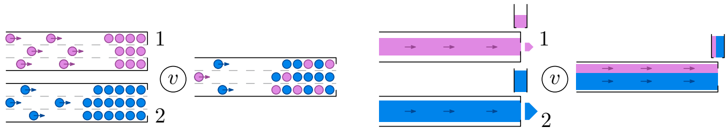

Fig. 2. Illustration of flow dynamics in MATSim (left hand side) and FOT model (right hand side). The numbers denote the capacities.

correspond to an exact user equilibrium. Fig. 1 illustrates this iterative process: The flow model described above corresponds to the *mobsim* module, where plans of agents are executed on the network. Afterwards, all executed plans get evaluated by the *scoring* module (in this study only based on the experienced travel times). Based on these scores, some agents are allowed to change their plans within their choice set while others select completely new ones during *replanning*. In this study, agents are only allowed to change their routes; in general also other choice dimensions (e.g., departure time or mode choice) are possible within MATSim. For more details on MATSim see Horni et al. (2016).

2.2. Flows Over Time (FOT) Model

An instance of an FOT problem is given by a directed graph $G = (V, E)$ where each arc is equipped with a transit time and an outflow capacity restricting the outflow rate. In order to model spillback, each arc is additionally equipped with an inflow capacity, restricting the inflow rate, and a storage capacity, bounding the total flow located on the arc at any point in time. The demand is specified by a finite set of commodities J, each $j \in J$ with its own origin-destination pair (o_j, d_j), a network inflow rate r_j, and an inflow interval I_j. An FOT is described by commodity-specific in- and outflow rate functions for every arc $e \in E$. More precisely, for every $j \in J$ and $e \in E$, the function $f^+_{j,e} : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ describes the rate at which flow of commodity $j \in J$ enters into arc $e \in E$ for every point in time. Analogously, $f^-_{j,e}$ denotes the outflow rate function. The sum over all commodities is called the total flow on an arc. Note that the FOT model is a continuous model with respect to time and flow, i.e., a flow rate can change at any point in time and flow can split up in arbitrary fractions. To obtain a feasible FOT, the family of functions $(f^+_{j,e}, f^-_{j,e})_{j \in J, e \in E}$ has to satisfy the following constraints:

- *Flow conservation:* Flow can neither be generated or vanish at intermediate nodes nor on the arcs. Flow of commodity j enters the network at the origin o_j at rate r_j during I_j and leaves at its destination d_j.
- *Arc dynamic:* Flow entering arc e first traverses the arc (duration is specified by the arc's transit time) and if possible leaves the arc immediately. The total outflow rate is restricted by the outflow capacity. That is, whenever the total flow rate exceeds the outflow capacity, a queue builds up. If there is a positive queue length, newly arriving flow has to wait and, as long as there is no spillback, the queue operates at outflow capacity. This means that the total outflow rate has to be equal to the outflow capacity in case of a positive queue length.
- *Perfect merge:* The commodity-proportion on the total flow is the same at the point of entrance and later at the point of exit. Thus, the continuous flow merges perfectly, as illustrated on the right-hand side of Fig. 2.

A more formal description of these flow dynamics can be found in the work of Cominetti et al. (2015) (single-commodity) and Sering and Skutella (2018) (multi-commodity).

In order to model spillback, an arc e becomes full when the total flow currently located on e reaches its storage capacity. In this case, the inflow rate into e is additionally bounded by the current outflow rate in order to ensure that e is not overloaded. To this end, the outflow rates of the preceding arcs must be reduced (*throttled*) causing spillback. If several preceding arcs need to be throttled, flow merges proportionally to their outflow capacities. In other words, if an arc (u_1, v) of capacity 1 and an arc (u_2, v) of capacity 2 both want to send flow into a full arc (v, w) with in- and outflow rate of 1, then (u_1, v) is allowed to send at a rate of $1/3$ and (u_2, v) can send at a rate of $2/3$. A formal definition of this spillback model, including some additional restrictions (e.g., no cycles of full arcs, to avoid deadlocks) can be found in the paper by Sering and Vargas Koch (2019).

A **Nash flow over time** is a dynamic equilibrium in the FOT model. More precisely, a Nash flow over time is a feasible FOT where each infinitesimally small flow particle arrives at its destination as early as possible. Hereby, the travel time depends not only on the network topology but also on the waiting times caused by all other particles. These

Nash flows over time are well-studied, and it has been proven that they always exists (Cominetti et al. (2015)). If all commodities share the same origin and destination (single-commodity), they can be constructed algorithmically (not efficiently though) and structured into different phases, where all flow takes the same paths within a phase.

2.3. Model Comparison

In general, MATSim and the FOT model are very similar. A closer look reveals some differences, see Tab. 1. The **flow dynamics** of the models have much in common, as they both use a deterministic queuing model combined with a FIFO policy. Delays occur by waiting in a queue due to the capacity constraints of arcs, and in both models these queues are spatial, such that spillback can occur. However, MATSim considers discrete vehicles and time steps, whereas the FOT model considers continuous flow and time; see Fig. 2. Note, that some adaptions have been made to MATSim's flow model (see Sec. 2.1) to be able to better compare the structural differences of the underlying models. Both models assume that users behave selfishly (looking for shortest paths), which is a reasonable assumption for modeling individual road traffic. However, the concept of a **dynamic equilibrium state** differs between MATSim and the FOT model. On the one hand, the co-evolutionary algorithm of MATSim is able to handle large-scale, real-world scenarios and leads to a state where *most* of the agents do not have any incentive to deviate (approximate user equilibrium). On the other hand, while Nash flows over time cannot be computed efficiently, they describe exact user equilibria where every single flow particle travels along a fastest path. Note, that both models cover multi-commodity instances; but so far there is no algorithm known to compute a Nash flow over time in this setting, while MATSim can compute an approximate user equilibrium for arbitrary s-t-pairs and inflow rates (i.e., individual agents).

	MATSim	Flows Over Time Model
Demands	individual agents	aggregated commodities
Agents	**discrete vehicles**	**infinitesimal flow units**
Time	**discrete time steps**	**continuous time**
Link Dynamics	deterministic FIFO queues	deterministic FIFO queues
Queues	spatial queues with spillback	spatial queues with spillback
Link Selection	*exact* prop. to outflow cap.	exact prop. to outflow cap.
Node Dynamics	*nodes blocked by spillback*	nodes blocked by spillback
User Equilibria	iteratively approximated	exact

Table 1. Overview on the most important similarities and differences of both models. Adaptions to standard MATSim are shown in italic.

2.4. Refinement Process

In order to examine the convergence of MATSim for decreasing vehicle and time step size, two refinement variables are used: The duration of a single time step is denoted by α and the volume of a single vehicle by β (only vehicles of uniform size are considered here). The coarsest refinement level is $\alpha = \beta = 1$, which corresponds to a time step size of one second and a vehicle size of one flow unit. A decrease in α means that there is an increasing number of time steps per second. Hence, the per-second-capacities of the arcs needs to be distributed uniformly to the new time steps in order to guarantee the same overall throughput. If α goes to 0 while β stays constant, the vehicles get isolated as the capacity is exhausted for many time steps after a vehicle has left a link. To avoid this, β needs to be decreased accordingly to maintain a constant flow. As illustrated in Fig. 3, the most natural coupling of the two parameters turns out to be $\beta = \alpha^2$, since it not only distributes the vehicles more evenly over time and possible routes but also allows for merging of different commodities per time step. Note that both parameters highly influence the run time of MATSim.

Since both models rely on the same deterministic queuing model, one can expect that for fixed routes MATSim's flow model converges to a flow over time for $\beta = \alpha^2 \to 0$. However, on each arc and in each time step rounding errors can occur which might amplify during the simulation. This is the case whenever the free flow transit time of an arc is not dividable by α or the outflow capacity is not dividable by β or whenever the perfect share of the commodities on an arc cannot be represented by multiples of β.

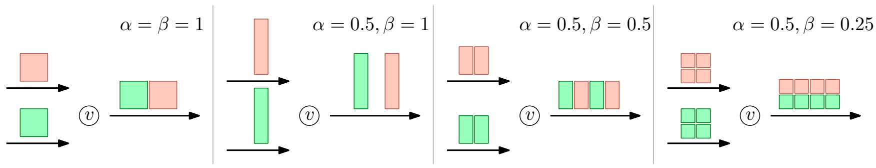

Fig. 3. Merging of vehicles for different values of α and β in MATSim, when one unit of flow is allowed to enter the downstream arc per second. From left to right: In the starting situation, one packet traverses in the first and the other one follows in the next time step. By choosing a time step size of $\alpha = 0.5$ and a vehicle size of $\beta = 1$ the packets become isolated. For $\alpha = \beta = 0.5$ packets distribute more evenly but still do not merge properly during time steps. Choosing $\beta = \alpha^2 = 0.25$ enables a perfect share of the capacity per time step to the two commodities.

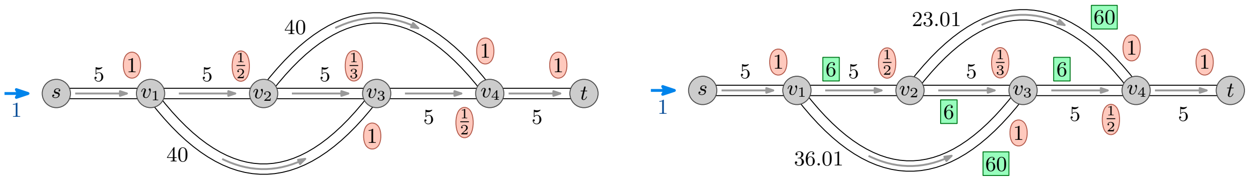

Fig. 4. The Braess network without spillback (left-hand side) and with spillback (right-hand side). The outflow capacities in flow units per second are given in the red ellipses; the transit times in seconds are displayed next to the arcs. The network inflow rate is one flow unit per second. For the spillback instance, the storage capacity in flow units is given in the green boxes. Inflow capacities are large enough to never restrict the flow.

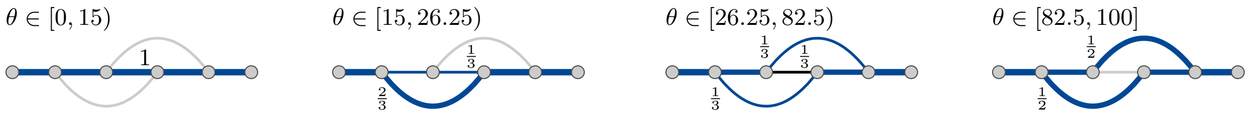

Fig. 5. The four consecutive phases in the Nash flow over time in the Braess network without spillback.

In order to compare both models, cumulative flow rates and travel times in MATSim are analyzed for the different refinement levels and compared to the Nash flow over time values. Travel times of vehicles are a meaningful characteristic in this case, as they are invariant for small time shifts and continuously depend on the cumulative flow.

3. Scenarios

In this section, the behavior of MATSim for decreasing packet and time step sizes is analyzed in two illustrative single-commodity examples of the Braess network and in a small multi-commodity real-world instance.

3.1. A Single-Commodity Instance without Spillback

The left part of Fig. 4 shows the Braess instance without spillback with one commodity traveling from s to t with a constant inflow rate of one flow unit per second over 100 seconds. The **Nash flow over time** in this instance consists of four consecutive phases, which are depicted in Fig. 5. All flow particles entering within the time interval $[0, 15]$ take the shortest middle route. As a queue builds up on arcs (v_1, v_2) and (v_2, v_3), travel times of the middle and bottom route balance and two thirds of the particles departing during time interval $[15, 26.25]$ take the bottom route. Particles entering the network during time interval $[26.25, 82.5]$ use all three routes in equal share while the queue on arc (v_3, v_4) increases. Finally, from time 82.5 to the end, the middle route has become unattractive as particles would need to wait on both arcs (v_1, v_2) and (v_3, v_4). Hence, half of the particles take the top and the other half the bottom route.

To compare the models, **MATSim** is run with multiple refinement levels starting with a vehicle size corresponding to one flow unit and a time step size of one second as the coarsest refinement level. As described in Sec. 2.4, this is refined by multiplying the time step size by $1/2$ and the vehicle size by $1/4$ until a time step size of $1/64$ and a vehicle size of $1/4096$ is reached. For all fixed refinements, 200 iterations of the co-evolutionary algorithm are run. In the first 100 iterations, 10% of the agents are allowed to reroute, i.e., select the shortest route based on the travel times of the last iteration; afterwards, this plan is added to the choice set together with a score for the experienced travel time in this

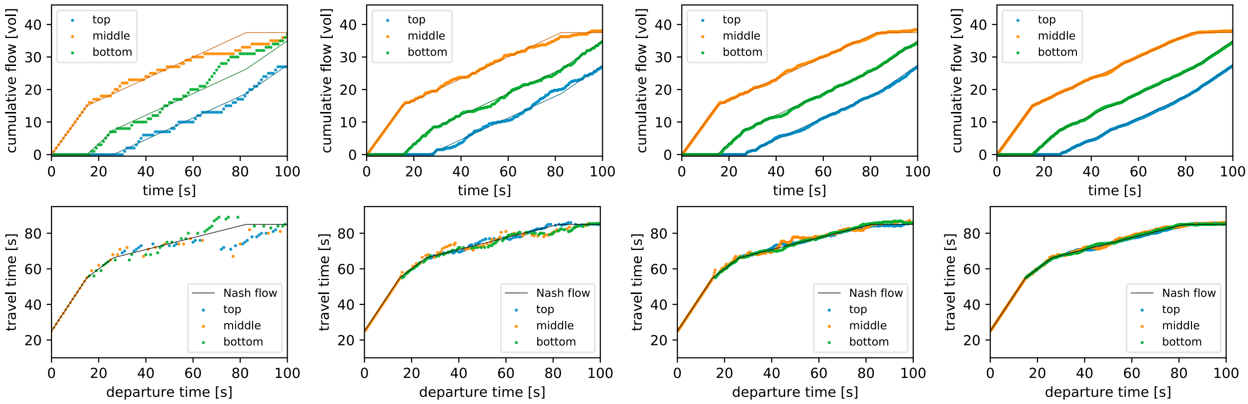

Fig. 6. Cumulative flow values and travel times of vehicles (in MATSim) and flow particles (in the Nash flow over time) for the three different routes in the network without spillback. Flow values and travel times in MATSim are given as colored points, the respective Nash flow over time values as solid lines. The plots show the results for time step sizes 1, 1/2, 1/4, 1/8 (from left to right) for one of the random seeds.

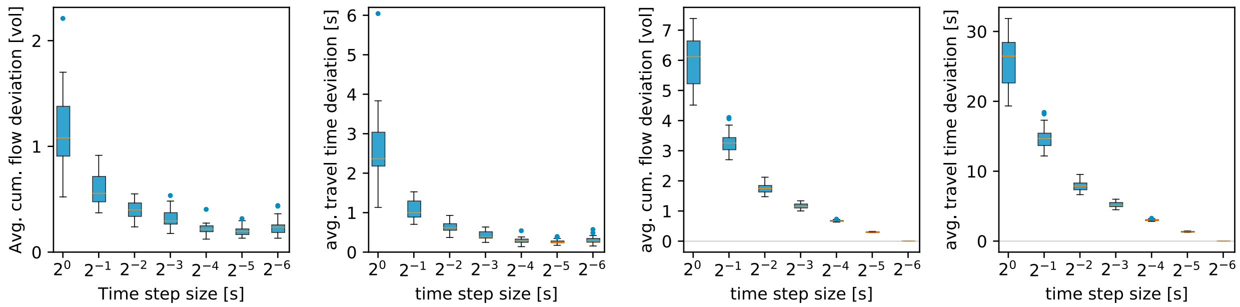

Fig. 7. Average deviation between MATSim and the Nash flow over time in the network without spillback (first/second) and with spillback (third/forth). Depicted are the deviations in cumulative flow values (first/third) and travel times averaged over all time steps (second/forth). One underlying data point of the box plot corresponds to a run with a specific random seed for the respective refinement level given on the x-axes. The boxes give the 50%-quantile and the orange line marks the median value.

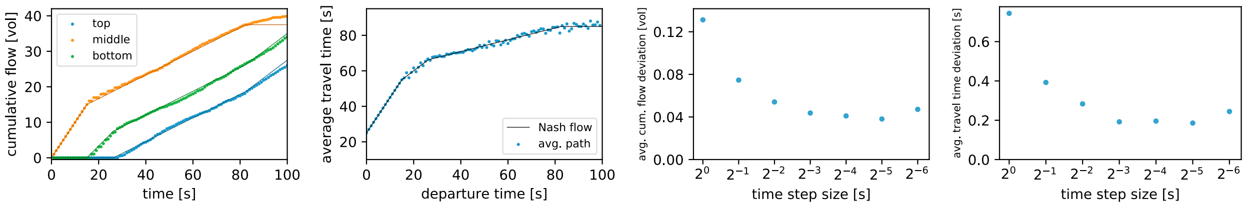

Fig. 8. Results averaged over all random seeds for Braess without spillback. Most left: Average cumulative flow values per path for $\alpha = \beta = 1$. Second left: Average travel times of all vehicles for $\alpha = \beta = 1$. Third and fourth: Average deviation between cumulative flow values / vehicle travel times in MATSim and the Nash flow over time.

iteration. The remaining 90% of the agents apply a logit model on the plans in their choice set, i.e., a plan is chosen with a certain probability based on the score. After the first 100 iterations, the fraction of players that are allowed to reroute is decreased stepwise. To be more precise, a method of successive averaging is applied on the probability with which the strategy reroute is chosen: It is set to the inverse of the iteration number minus 90. This means that in the last of the 200 iterations less than one percent of the agents reroute while all the remaining agents choose a plan within their choice set. Some parts of MATSim depend on random values – in this setup this mainly applies to the logit model that chooses plans from the choice set of the agents. This is why each refinement step is run for 20 different random seeds to be able to average out the deviations depending on the specific random values.

The **results** are depicted in Fig. 6-8. Fig. 6 shows the cumulative flow values and travel times in MATSim compared to the ones from the Nash flow over time for one of the random seed runs with the first four simulated refinement levels. It can be observed that the four phases of the Nash flow over time (see Fig. 5) also occur in MATSim. With increasing refinement, the curves get smoother and converge, up to small discrepancy, to the values of the Nash flow over time. In order to measure this discrepancy, Fig. 7 shows the average deviation of all time steps between the MATSim and Nash flow over time values. The variance and the median of the different random seed runs decreases with increasing refinement level but the data suggest that this process is not converging to zero. One major reason is that for some random seeds the flow along a path is underestimated while it is overestimated for others. By taking the average over all 20 random seeds in both plots the deviation is even decreased further; see Fig. 8. This confirms the stochastic perspective that the Nash flow over time is the expected value of a distribution, which each MATSim run is a realization of.

3.2. A Single-Commodity Instance with Spillback

This section considers the same example as the previous section but enables spillback in both models. Additionally, transit times are modified on the bypass arcs such that the resulting user equilibrium better visualizes the differences of the models; see right part of Fig. 4. Considering this example is interesting from two perspectives. On the one hand, it is not clear that both models still align when spillback occurs. On the other hand, enabling spillback in this example highly impairs the efficiency of the user equilibrium. For MATSim this effect was already previously discussed by Thunig and Nagel (2016); a similar example for the FOT model was given by Sering and Vargas Koch (2019).

The **Nash flow over time** in this instance is rather simple. All particles take the middle route without any deviation. The reason for this is that at time 12 arc (v_1, v_2) becomes full causing spillback on (s, v_1) and later on, at time 36, arc (v_2, v_3) becomes full reducing the total throughput rate to $1/3$. As the travel time along (v_1, v_2, v_3) peaks at 36, the long arc (v_1, v_3) will never be on a quickest path. Similarly, the segment (v_2, v_3, v_4) has a maximal travel time of 23, which is the reason that no particle will ever take arc (v_2, v_4) in the Nash flow over time.

For **MATSim**, the same simulation setup as in the case without spillback is used (see Sec. 3.1). Since link storage capacity values are now much lower, links can get full and congestion spills back to upstream links. This has the effect that a queue builds up on the middle route spilling back until the first link, and travel times of the different routes do not balance. As a consequence, the middle route is the shortest for (almost) all vehicles.

The **results** for this scenario are depicted on the right of Fig. 7. Again, for increasing refinement levels the variance and the deviation decrease. After the middle path has become fully congested due to spillback, its travel time in the FOT model is 0.01 seconds shorter than the travel time of each bypass route. However, in MATSim, this travel time difference depends on the refinement level, as links can get slightly overfull, which causes longer waiting times on the middle path as in the FOT model – especially for coarse refinement – such that the bypass routes are still attractive. Vehicles taking the bypass routes lead to shorter overall travel times compared to the Nash flow over time in this Braess instance, as it is closer to the system optimal route distribution, in which only the top and bottom routes are used in the long run. For smaller vehicle sizes the waiting time deviation due to slightly overfull links decreases and the middle route becomes the shortest. Thus, the MATSim solution converges to a situation where all agents travel along the middle path and experience the same travel time as in the Nash flow over time. This state is reached for $\alpha = 1/64$.

3.3. A Multi-Commodity Real-World Instance

The last example presented here is a simplified network of the inner city of Cottbus, Germany. It is considered here in order to compare the flow dynamics of the FOT model with MATSim in a more realistic scenario with multiple commodities. Since there is no algorithm known to compute a Nash flow over time in a multi-commodity setting, the routes in both models are fixed to shortest paths according to the free flow transit times. Note, that this revokes the need of multiple random seed runs in MATSim, since due to the adaptions described in Sec. 2.1 the flow model is deterministic. The demand in this scenario consists of 20 commodities connecting each ordered pair of the five labeled entry/exit nodes *A* to *E* in Fig. 9. Each commodity has an inflow rate of $1/5$ flow units per second for a time interval of 1000 seconds. Spillback is not considered in this example.

As can be seen in Fig. 10, the travel time deviation between MATSim's flow model and the FOT converges to zero for smaller refinements. Interestingly, the deviations stay more or less constant during the simulation; see right-hand

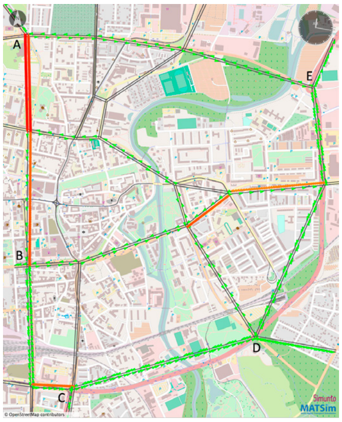

Fig. 9. Inner-city network of Cottbus. The colored triangulars depict the vehicles in the simulation at second 1000; the more red, the more they are delayed.

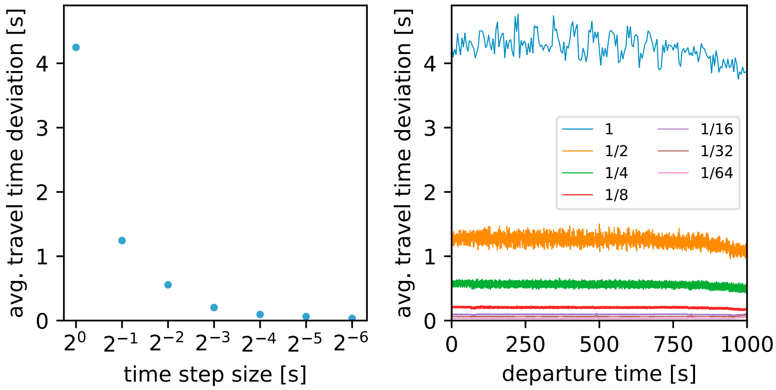

Fig. 10. Deviations in travel time between MATSim and the FOT model in the Cottbus scenario. Left: Travel time deviations for different refinement levels averaged over all commodities and all points in time. Right: Travel time deviations for different refinement levels over time; the average is taken over all commodities.

side of Fig. 10. This suggests that waiting times do not diverge and deviations are mainly caused by the rounding of transit times to the next discrete time steps. The small oscillations are probably due to the translation of accumulated capacities to discrete vehicles in MATSim and the merging of different commodities, which is less smooth for coarse vehicle sizes. For higher refinements, the total deviation and the oscillation amplitude decrease as the rounding accuracy increases and vehicles of different commodities merge more fluently.

4. Conclusion

This study shows a strong connection between MATSim and the FOT model and their respective equilibria. It confirms that a MATSim run can be seen as the realization of a random distribution of which the Nash flow over time constitutes the expected value. Furthermore, the experiments indicate that the flow in MATSim with fixed routes converges to the corresponding flow over time for decreasing time and vehicle sizes. The agent dynamics of the models, however, cause a bigger deviation when comparing a Nash flow over time with a user equilibrium in MATSim – especially for coarse discretization levels. Still, during the refinement process the resulting states approximate very well. The achieved results strengthen the relevance of both models. This provides a justification for the FOT model and a theoretical foundation for MATSim.

These observations open up a couple of further research directions. Ongoing research aims at formalizing and rigorously proving convergence of the two flow models. In addition, it might be possible to transfer some of the proven properties of Nash flows over time, e.g., uniqueness, long term behavior, and price of anarchy, to user equilibria in MATSim. Finally, in order to strengthen the connection between MATSim and the FOT model even further, it is interesting to analyze user equilibria in more complex scenarios including kinematic waves and multiple commodities.

References

Cominetti, R., Correa, J., Larré, O., 2015. Dynamic equilibria in fluid queueing networks. Operations Research 63, 21–34.
Fleischer, L., Skutella, M., 2007. Quickest flows over time. SIAM Journal on Computing 36, 1600–1630.
Haurie, A., Marcotte, P., 1985. On the relationship between Nash—Cournot and Wardrop equilibria. Networks 15, 295–308.
Horni, A., Nagel, K., Axhausen, K. (Eds.), 2016. Multi-Agent Transport Simulation MATSim. Ubiquity Press, London.
Koch, R., Skutella, M., 2011. Nash equilibria and the price of anarchy for flows over time. Theory of Computing Systems 49, 71–97.
Scarsini, M., Schröder, M., Tomala, T., 2018. Dynamic atomic congestion games with seasonal flows. Operations Research 66, 327–339.
Sering, L., Skutella, M., 2018. Multi-source multi-sink Nash flows over time, in: 18th ATMOS Workshop, pp. 12:1–12:20.
Sering, L., Vargas Koch, L., 2019. Nash flows over time with spillback, in: Proceedings of the 30th Annual ACM-SIAM SODA, pp. 935–945.
Thunig, T., Nagel, K., 2016. Braess's paradox in an agent-based transport model. Procedia Computer Science 83, 946–951.