

HYBRID SIMULATION MODEL SUPPORTING EFFICIENT COMPUTATIONS WITHIN RAIL TRAFFIC SIMULATIONS

Radek Novotný^(a), Antonín Kavička^(b)

^{(a),(b)}Faculty of Electrical Engineering and Informatics, University of Pardubice

^(a)radek.novotny3@student.upce.cz, ^(b)antonin.kavicka@upce.cz

The contribution deals with the unitary hybrid simulation model of railway traffic, which applies different levels of abstraction (microscopic and macroscopic) to different parts of the simulating system. As a result of applying this approach, simulation experiments can reflect the traffic within a large part of the railway network. The entire computational complexity is supposed to be significantly lower in comparison with an application of a pure microscopic simulator. From this point of view, the hybrid model can also be used to optimize the design of simulators of large and complex railway systems.

Special attention is paid to stochastic simulation experiments (applying random train delays) and to transforming train flows at the interface between pairs of microscopic and macroscopic submodels.

Keywords: scalable simulation model, unitary hybrid model, railway traffic, transformation of traffic flows, random train delays

1. INTRODUCTION

A significant part of the research in the field of optimizations of railway systems represents modelling a railway infrastructure and relevant rail traffic. For these purposes, the experimental research method of computer simulation is utilized. Depending on the required level of details, different simulation models (*microscopic*, *mesoscopic* or *macroscopic*) can be applied. From the viewpoint of modelling large railways systems, it is necessary to pay attention to their computational-efficient design.

Currently, the standard approach is based on the application of the identical level of details (*granularity*) to the entire simulator. If traffic simulations mirror large railway systems, time-efficient computations are required. That demand can be met by lowering the level of details either within the entire simulating system or within its selected parts. Hence, there is a strong motivation for designers of traffic simulations to use methodologies for building scalable traffic simulators. Those models enable to combine and interconnect various submodels of infrastructure (applying different levels of details) and different traffic submodels reflecting granularities of relevant infrastructural submodels (Hansen and Pachel 2008; Cui, and Martin 2011; Novotny and Kavicka 2016). Certainly, relevant

transformations of traffic flows are supposed to be carried out on the boundary between corresponding submodels (Novotny and Kavicka 2017; Novotny and Kavicka 2018).

2. UNITARY HYBRID SIMULATION MODEL

Our presented methodology called *HybridRail* (Novotny and Kavicka 2016) is based on a hybrid simulation model implemented within one simulation tool (*unitary hybrid model*). That methodology supports combining submodels exploiting the microscopic and macroscopic levels of details. Microscopic simulation is connected to particular areas, within the frame of which the details about traffic (and infrastructure) are important for an experimenter. On the other hand, macroscopic simulation is applied to those parts of the simulator, where rough operational/traffic observations are sufficient. Unitary hybrid model enables to adjust the *granularity* of selected individual parts of a simulator. The mentioned parts are connected to relevant traffic submodels operating over corresponding infrastructure submodels. Overall computational demands of a unitary hybrid model (composed of non-homogeneous submodels) are certainly lower than demands related to a corresponding model executing pure microscopic simulation (Novotny and Kavicka 2016).

3. HYBRID INFRASTRUCTURE SUBMODEL

The methodology *HybridRail* (describing hybrid simulator building process) focuses primarily on the construction of a track infrastructure submodel. From the implementation viewpoint, it is necessary to distinguish between *micro-layer* and *hybrid layer*. *Micro-layer* contains a microscopic infrastructure submodel that corresponds to the highest level of details which can be applied to the track infrastructure. On the other hand, the *hybrid-layer* is composed of *micro-segments* and *macro-segments*. *Micro-segments* are directly taken from the micro-layer. *Macro-segments* apply higher degree of granularity (i.e. lower level of details) to relevant parts of the *micro-layer*. Two types of macro-segments (*macro-nodes* and *macro-edges*) are distinguished within the presented methodology. *Macro-edges* typically encapsulate line sequences of edges from micro-layer (typically reflecting rail lines).

Macro-nodes can enclose general continuous micro-layer areas. Constructions of *macro-segments* support constructing variant configurations of hybrid submodels of railway infrastructure (Novotny and Kavicka 2016). For the purposes of visual deformation of the hybrid infrastructure model (especially around the *macro-segments*), it is necessary to distinguish between schematic visualization of infrastructure and definition of numerical values (parameters) related to metric (topological) and slope properties within the track layout (Novotny and Kavicka 2015).

4. HYBRID TRAFFIC SUBMODEL

Because of combining *micro-* and *macro-segments* within the *hybrid-layer*, it is necessary to apply different traffic submodels (exploiting various levels of abstraction), which are connected to different traffic indicators. From the viewpoint of the implementation of the unitary hybrid model it is necessary to distinguish between traffic submodels carried out over microscopic (*microscopic traffic submodels*) and macroscopic (*macroscopic traffic submodels*) infrastructure segments. Because several different traffic submodels coexist within a hybrid model, it is necessary to solve transformations of traffic flows, i.e. it is necessary to unambiguously define the information about railway traffic on the interface between each microscopic and macroscopic submodel in order to maintain consistency of data (Burghout 2004; Burghout 2006; Novotny and Kavicka 2016).

5. MICROSCOPIC TRAFFIC SUBMODEL

Microscopic traffic submodel is based on applying realistic calculations connected with the train ride dynamics and train interactions (Salva 2005; Divis and Kavicka 2015). Train rides computations use detailed information about railway infrastructure (real topology/metric and slope properties) defined in *micro-layer* of infrastructure submodel. In other words, the movement of trains on the infrastructure reflects the real driving characteristics with respect to the train, track properties and traffic restrictions (Novotny and Kavicka 2017).

6. MACROSCOPIC TRAFFIC SUBMODEL

The basic idea is that trains do not have to be individually monitored during their movement in *macro-segments* (unlike within the above-mentioned microscopic traffic submodel). In other words, if the train can enter the macroscopic submodel, the moment of the simulation time, when the train leaves the submodel, is individually calculated (certainly, with the respect to railway timetable).

Because of existing differences between *macro-segments* due to the encapsulation of different parts of the microscopic model of infrastructure (railway stations may typically be encapsulated into *macro-nodes* and sections of rail lines to *macro-edges*) within the hybrid model, there are various concepts of realization of macroscopic traffic submodels. Those concepts can be based on an analogy with fluid flow or competition for shared resources (Novotny and Kavicka 2016). Due to the results of simulation experiments (carried out in a deterministic regime) that tested different concepts of macroscopic traffic models, a model based on the competition for shared resources was finally accepted for both types of *macro-segments* (Novotny and Kavicka 2018).

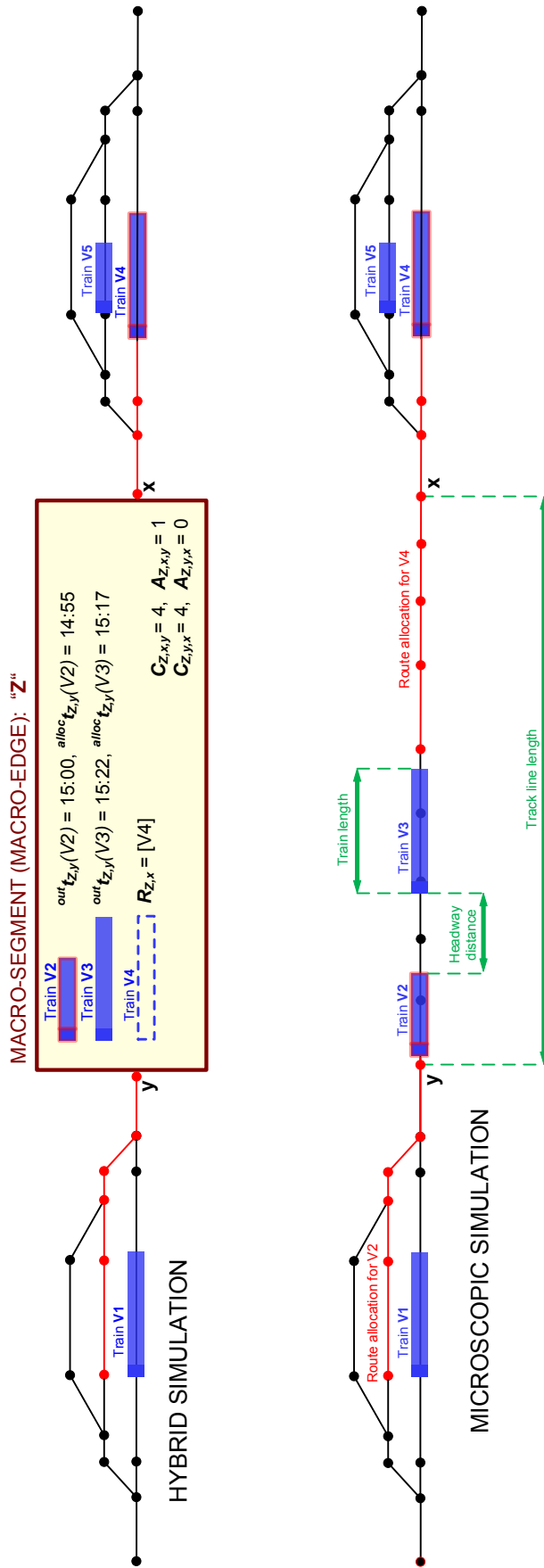
For the needs of stochastic simulation experiments (applying random train delays), it is necessary to specify the following *macro-segment* characteristics:

- the maximum permissible number of trains within the submodel,
- the total train dwell time in the submodel,
- the method of detecting the occupancy of adjacent submodels for the given future time windows and, where applicable,
- the train parameters needed for their movement in adjacent submodel to which they will enter.

6.1. Determination of the maximum permissible number of trains within the macro-segment

Generally, from the viewpoint of the selected macroscopic traffic submodel, the *macro-segment* can be understood as a form of shared resource that is able simultaneously/parallelly to hold only a certain number of trains. The maximum permissible number of trains within this submodel can be calculated on the basis of evaluations applied to selected attributes defined in the *micro-layer*.

In the case that a *macro-edge* consists of only one possible train path, the train input is dependent on the length of the train path, headway distance (determined for example by a table of stopping distances), the length of the train and, of course, the number of trains currently sojourning inside the submodel. On the other hand, the *macro-node* can consist of a set of primary and alternative train paths, which can be allocated to each direction. Thus, it is necessary to divide the station tracks according to all possible directions and then determine the possibility of entering the next train into *macro-node* (the number of currently available station tracks in the given direction and the current number of staying trains within the submodel are considered).



LEGEND

$out_{M,x}(i)$	the expected output time of the train i from macro-segment M through the adjacent element x
$alloc_{M,x}(i)$	the time of an earliest possible application (related to the train i outgoing from macro-segment M through the adjacent element x) to an allocation/occupation of a relevant resource (e.g. the train path) within a neighbouring segment
$C_{M,x,y}$	the maximum capacity of macro-segment M with regard to the ordered pair of adjacent elements $[x,y]$; element x represents an input element of M and y an output element of M
$A_{M,x,y}$	the current available capacity of macro-segment M with regard to the ordered pair of adjacent elements $[x,y]$; element x represents an input element of M and y an output element of M
$R_{M,x}$	dynamic vector of currently realized reservations/allocations/occupations for inputs of trains into macro-segment M through to the adjacent element x

Figure 1 - Hybrid simulation model of railway traffic

In the case of definition *macro-node* capacity by the number of station tracks, it is possible to take into account the occupation of one station track by more trains at the same time. It corresponds to the real situation at the railway station.

Due to the nature of the railway traffic, it is necessary to predict the occupation of the *macro-segment* by a given train (via the dynamic “reservation” vector R) before it actually enters the *macro-segment* (Figure 1). Of course, this pre-reservation of the *macro-segment* capacity must be feasible not only in the microscopic traffic submodel (adjacent to the *macro-segment*), but also potentially in the macroscopic submodel, especially when is possible to combine *macro-segments* within a hybrid model.

6.2. A priori determination of train dwell time within the macro-segment

Calculation of the instant of simulation time, when the train leaves a given *macro-segment*, is always carried out at the moment when the train enters that segment (thus, a relevant predicted dwell time is determined). The computation of the train dwell time takes into account the following aspects:

- the number of admissible train paths for a given train transit through the *macro-segment*,
- the average lengths of the mentioned paths,
- the average train speed and
- the relevant time addition (e.g. because of stopping a train at a railway station or decelerating in front of a railway station, etc.).

Correct calculation of the time addition $^{add}_t$ is quite important namely for the realistic determination of overall train running times (including acceleration and deceleration phases of the runnings).

In cases, when the train stops within *macro-node* according to the timetable, the time addition can be calculated as $^{add}_t = ^{dec}_t + ^{acc}_t + ^{dwl}_t$, where

- $^{dec}_t$ represents the time addition due to train braking from average speed (defined in the submodel) to complete stop,
- $^{acc}_t$ represents the time addition due to train acceleration from zero speed to average speed and
- $^{dwl}_t$ corresponds with the train dwell time at the platform in railway station according to the timetable.

If the train has to brake in front of the railway station in a part of infrastructure that is encapsulated in the *macro-edge*, it is necessary to determine not only the train speed (when leaving the submodel), but again in particular the correct time addition due to braking from the average speed defined in the submodel.

For above mentioned calculations of time additions is applied an approach based on the principle of uniformly accelerated respectively decelerated motion with using acceleration parameter (derived from the traction characteristic according to the average speed in the submodel) and train braking rate.

However, when calculating the train dwell time in the case of a *macro-edge*, it is necessary to consider the fact that train movements follow a FIFO-regime, i.e. they cannot be overtaken like within the *macro-node* (e.g. encapsulating a railway station). From the viewpoint of calculating dwell time of the individual trains and the FIFO-regime, it is important to consider the minimal time interval between trains, i.e. headway distance.

6.3. Determining the occupancy time of the subsequent submodel

Due to the coexistence of different submodels in the hybrid traffic model, the problem of train path occupancy within a relevant adjacent submodel (microscopic or macroscopic) must be potentially dealt. In other words, this means, that even if the train is currently in a given *macro-segment*, the train path or other kind of resource within an adjacent segment must be (reserved and) occupied sufficiently in advance (Figure 1). The train enters that adjacent segment later. As an example, the following case can be mentioned: a train plans to move from the *macro-edge* to the adjacent microscopic submodel (reflecting a railway station).

As a result, when the train enters a macro-segment, the moment ($^{alloc}_t$) of a relevant allocation time is calculated. Allocation time represents an earliest possible moment, when a needed occupation within a relevant subsequent segment is launched.

If all required resources are occupied at the given time ($^{alloc}_t$), it is necessary to wait, which will dynamically affect the train dwell time in the current submodel and also the value of the simulation time $^{out}_t$, when the train will leave this submodel. If the train cannot leave the *macro-edge* for a reason defined above, it is necessary to keep the minimal time interval between trains according to the headway distance. In other words, the dwell time of other trains (which entered to the *macro-edge* behind this train) can be affected.

6.4. Determination of other train parameters for their movement in the subsequent submodel

From the viewpoint of the train leaving the *macro-segment* and its consistent transition to the adjacent submodel (*microscopic* and *macroscopic*), it is necessary to determine the values of other necessary train parameters. These parameters include:

- the current output speed of the train (affected by average speed or acceleration / braking of the train in the *macro-segment* etc.) and
- specific output element/point of the *macro-segment*.

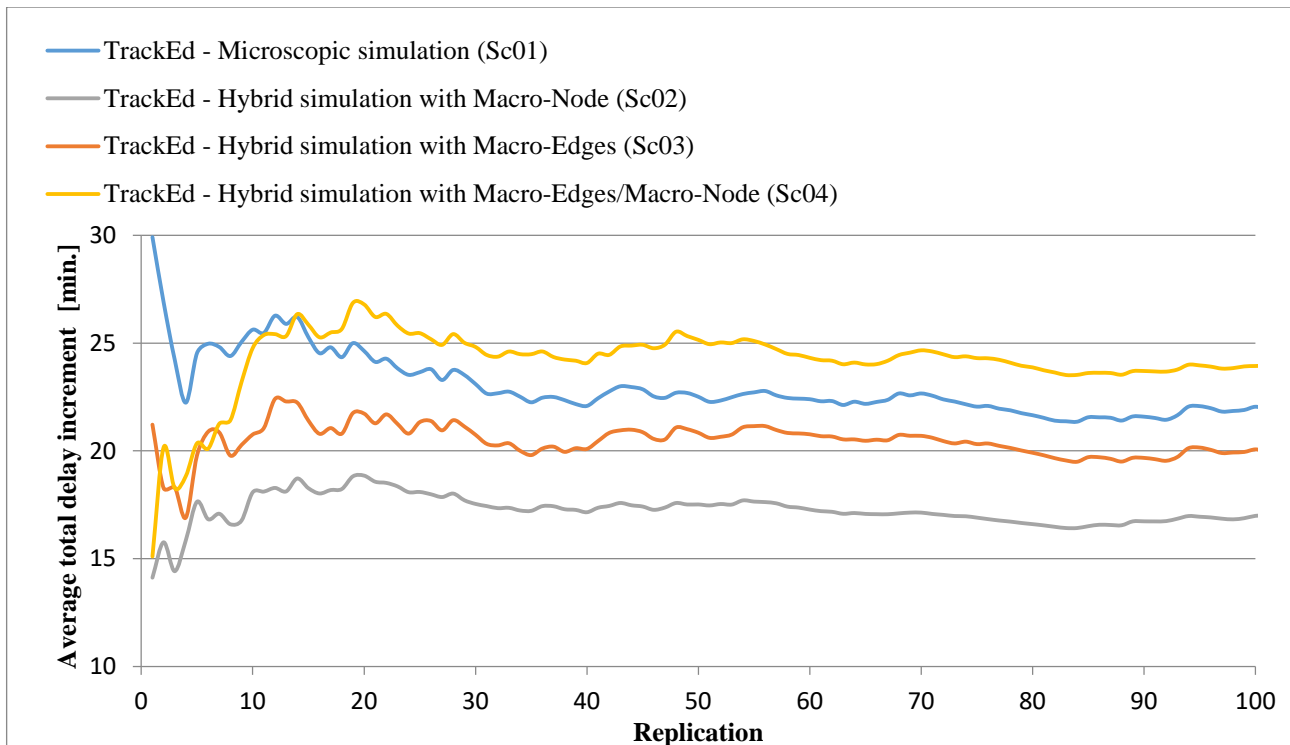


Figure 2 – Comparison of average total delay increments

7. VALIDATION

Regarding applications of a unitary hybrid model in practice, it is necessary that the results of deterministic and especially stochastic simulation experiments are correct and correspond to reality. In order to validate a unitary hybrid simulator, the following comparisons are supposed to be realized. The results of the simulation experiments for different scenarios related to the hybrid simulations are compared with the corresponding results obtained from microscopic simulations (both carried out within the same integrated development environment – e.g. the tool *TrackEd* can be applied). In addition, the comparisons with the results from other simulation tools (specialized in rail traffic simulations) are processed as well (e.g. simulation tool *Villon* is applied).

The mentioned comparisons will consider achieved running times of trains and their total delay increments. In other words, the desired behavior of hybrid models is that the transformations of traffic flows at the interfaces of the microscopic and macroscopic segments must have a minimal effect on the total train delays (compared with the values obtained from microscopic models). It is also important to compare the computational complexities of the simulation experiments performed within the microscopic and hybrid models.

In order to verify the correctness, the infrastructure of a part of the railway network of the Czech Republic comprised of a prototype station and border stations used for dispatching trains was used. Within the validation of the hybrid simulation model, both

deterministic and stochastic simulation experiments were performed, for selected four different scenarios:

- (Sc01) simulation model working exclusively on microscopic level of details,
- (Sc02) hybrid simulation model with prototype station encapsulate in the *macro-node*,
- (Sc03) hybrid simulation model with sections of rail lines (paths between prototype station and the border stations) encapsulate in the *macro-edges*,
- (Sc04) hybrid simulation model with incidents of *macro-edges* and *macro-node* (combination of the previous two scenarios).

Primarily, it was necessary to validate and verify the implementation of train running dynamics and thereby confirm the correctness of used microscopic traffic model. A series of deterministic simulation experiments was performed according to a real timetable (containing several dozen trains in both directions). Basically, it was a comparison of a tachograph compiled from a real environment with a tachograph obtained from simulator results (a series of simulation experiments). Furthermore, the results obtained from the *TrackEd* tool (using the *MesoRail* computational core) were compared with the results from the already validated simulator *Villon*, which represents a simulation tool generally accepted by railroad experts (Divis and Kavicka 2015; Novotny and Kavicka 2017).

In the deterministic mode, both the above-mentioned concepts of macroscopic traffic models (based on an analogy with fluid flow or competition for shared resources) were tested in different scenarios of the

hybrid simulation model. From viewpoint of comparison with the results obtained in purely microscopic traffic simulation, the concept of traffic based on the analogy of shared resource was chosen for both types of *macro-segments* (Novotny and Kavicka 2017; Novotny and Kavicka 2018).

As a result of the comprehensive validation of the unitary hybrid model, it was necessary to load railway traffic with random train delays (stochastic mode) and monitor how train delays would change (the most important indicator of interest is the value of the potential increase of delay) when passing through the infrastructure under all of the above scenarios.

Table 1: Parameters of random trains delay

Train category	Probability of train delay [%]	Mean value [s]	Maximum [s]
Express	50	420	7200
Stopping	33	270	7200
Cargo	50	1800	7200

In accordance with the methodology of SŽDC, s.r.o. (Rail Infrastructure Administration of the Czech Republic), the values of random train delays during simulation experiments were driven by exponential distribution of probability with parameter values specified for different train categories (*Table 1*).

Within all of the scenarios mentioned above, 100 replications were carried out with the application of random train delays. Initial testing was focused on the validation of a microscopic traffic model by comparing

running times and delay increments achieved in another already validated simulator, Villon (similar as with the deterministic regime). In both cases, almost identical average total running times per replication was achieved - the average total delay increment of all trains (46) in replication then reached about 22 minutes.

In the next validation phase, the above indicators were monitored using *macro-segments* in relation to the results obtained using a simulation model working exclusively on microscopic level of detail (*Figure 2*). In a scenario using only railway station encapsulation into a *macro-node* (Sc02), the average total delay increment per replication was reduced by 5 minutes. This deviation in tens of percent may be due to the fact that in the microscopic traffic submodel, a part of the station (i.e. the track branching) may be occupied when the train is running to the station track, which can delay other trains entering or leaving the railway station.

Conversely, in the scenario using the encapsulation of only sections of rail lines into *macro-edges* (Sc03), the average total replication delay increment was slightly reduced (namely by 2 minutes). This may be due to the application of a different logic in traffic submodels (macroscopic versus microscopic) in cases where the train has to wait in front of a fully occupied railway station.

When comparing the hybrid model, which includes *macro-edges* and *macro-nodes* (Sc04), with a purely microscopic simulation model (Sc01), there was a difference in the achieved average total delay increments on replication. This difference reached about 2 minutes. Due to the many differences between the microscopic and macroscopic traffic submodel of traffic, this result can be assessed as valid.

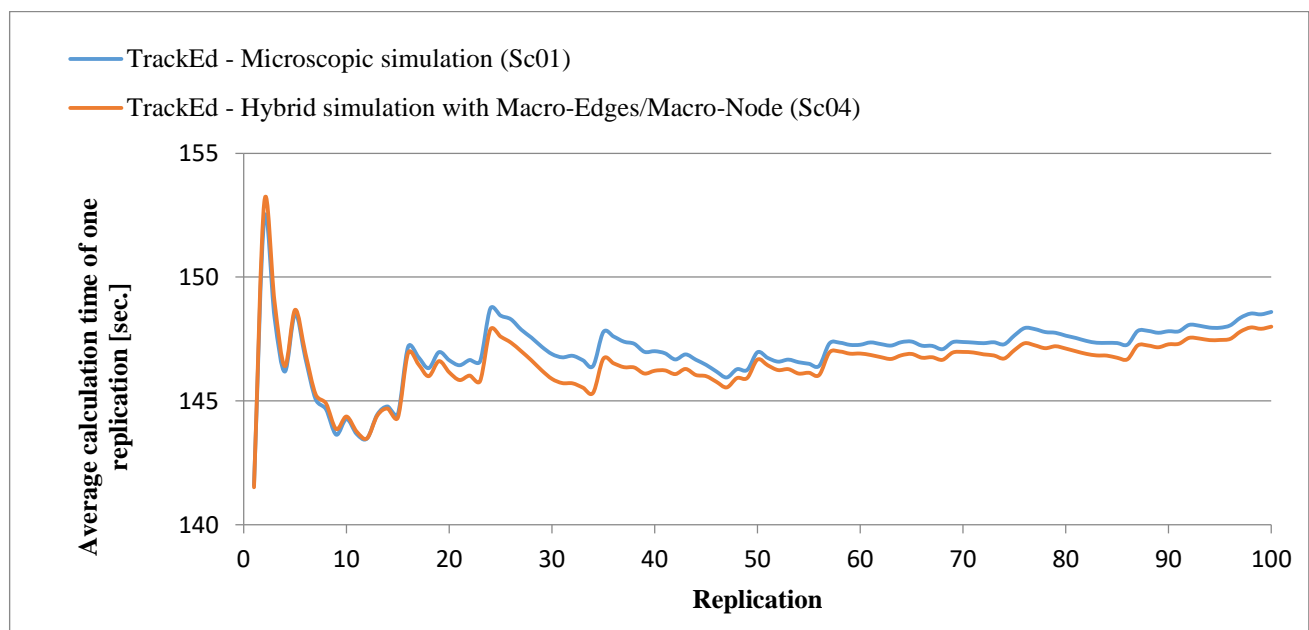


Figure 3 – Comparison of average calculation time of one replication between hybrid and purely microscopic simulation model