

PAPER • OPEN ACCESS

Local time evolution in Personal Communication Service model

To cite this article: Liliia Ziganurova and Lev Shchur 2021 *J. Phys.: Conf. Ser.* **1740** 012005

View the [article online](#) for updates and enhancements.



IOP | ebooks™

Bringing together innovative digital publishing with leading authors from the global scientific community.

Start exploring the collection—download the first chapter of every title for free.

Local time evolution in Personal Communication Service model

Liliia Ziganurova^{1,2} and Lev Shchur^{1,3}

¹National Research University Higher School of Economics, 101000 Moscow, Russia

²Science Center in Chernogolovka, 142432 Chernogolovka, Russia

³Landau Institute for Theoretical Physics, 142432 Chernogolovka, Russia

Abstract. We investigate the local time evolution in the Personal Communication Service (PCS) model simulated with the parallel discrete event simulation method's optimistic algorithm. We propose a model for the optimistic local virtual time evolution (OLVT) in PCS, which is reminiscent of statistical physics's surface growth. We use Rensselaer's optimistic simulation system with the Time Warp implementation. We compare the results of the simulations of both PCS and OLVT models and found good agreement. We discuss the highlights of our approach in the analysis of scalability and synchronization using the OLVT model.

1. Introduction

Parallel Discrete Event Simulation (PDES) [1] is a large-scale simulation technique used in a wide range of fields, from physics and engineering to biology and social sciences. The discrete-event model contains a description of the system's status and the events that can occur in it. Events occur at random time moments and change the system's state (for example, in a magnetic system, an event is a spin-flip; and in the logistics, the event is a change of the status of delivery).

Today, many software platforms (simulators) for PDES facilitate using the method for solving applied problems. Among them, the most famous are ROSS [2], WARPED2 [3], Simian [4], SPEEDS [5], PDEVs [6], and NS-3 [7, 8]. The main functionality of PDES simulators is the event planner, i.e., organizing the execution of events by logical processes and receiving and transmitting messages containing the event information, and providing the other details of the simulation process. The successful application of the optimistic PDES simulator was demonstrated on almost two million cores [9].

Simulators' critical features are the scalability of parallel simulations and the synchronization of the processes [1, 10, 11]. Analysis of modeling profiles is successfully used to optimize modeling tools and algorithms.

In the paper, we present *the model of the model*, i.e., the model of local virtual time evolution (OLVT) of the optimistic simulator running the particular model of interest. The OLVT model can capture the essential properties of the real simulation model and provide information on the — i) possible lockdown; ii) on the scalability, and iii) on the desynchronization of parallel processing units depending on the parameters of simulations. We should stress that information is valid for a particular model and a wide range of parameters. In contrast to the widely used modeling profiles for each particular model and particular values of parameters, the proposed



approach provides information on the particular model's behavior in a wide range of parameters. We do not pretend on a very detailed analysis of the algorithm performance and provide additional information on the simulator's general behavior running the particular model, which can be informative for the general understanding performance. In other words, we do not pretend that our approach is superior to any other approaches or can substitute them. Instead, our point is to present the new and complementary way of algorithm analysis.

We present the preliminary results of applying our approach to the Personal Communication Service (PCS) model. The results of the PCS simulations are in good agreement with the OLVT model. Both models undergo a roughening transition.

Our approach is based on the previous analysis of the models of conservative algorithm [12] and optimistic algorithm [13] and models on the small-world networks [14, 15, 16]. In contrast to these studies, we build the OLVT model of optimistic simulation of a particular model — the PCS model. The new ingredient is the simulation of the model for the specific pair of two parameters in the model, which is correlated with the PCS model.

2. PCS model

PCS (personal communication service) model simulates a wireless communication network that provides communication services for mobile units [17, 18, 19]. The service area of the network is partitioned into checkerboard areas (cells). The simulation model consists of cell objects and portable objects associated with mobile devices. Each cell object represents a cellular receiver/transmitter with a fixed number of channels. The portable object represents a mobile unit that resides within the cell for some time and can move during the call to one of the four neighboring cells. The portable object's events are call arrival, call completion, and move to a neighboring cell. When a portable device moves to the neighboring cell, the currently allocated channel must be freed, and the destination cell object must allocate a channel to the portable object. It happens practically instantaneously.

The model has several customizable parameters, and typically the number of portables is much larger than the number of cells. The most important are the parameters that specify the mean values of the distributions:

- MOVE_CALL_MEAN - the average time between mobile devices moving from one cell to another,
- NEXT_CALL_MEAN - the next call average,
- CALL_TIME_MEAN - the average call length.

We run the PCS model using the ROSS simulator (Rensselaer's optimistic simulation systems) [2]. The ROSS simulator is a parallel discrete event simulation system based on the Time Warp synchronization algorithm [20] and is widely used in the scientific community [21, 22, 23, 24, 25].

The ROSS simulator's output contains the total number of events processed, the total number of rollbacks, the number of events processed per second (event rate), the number of events with the portable moves to the neighboring cell. In our case, it is essential to get in the output the local virtual times of processes. However, it can be done when calculating the global virtual time (GVT), which happens after 256 events processed by the default setting.

3. OLVT model

In this section, we introduce the *model* of the local virtual (LVT) time evolution in PCS *model* simulated with an optimistic algorithm - the OLVT model. It is based on the ideas of the models of local time evolution in the conservative PDES algorithm [12], FaS algorithm [26], and optimistic algorithm [13].

In the OLVT model, we associate each cell object with the logical process (LP_i) and initiate the flat profile of the local virtual time τ_i of each LP_i ($i = 1, 2, \dots, N$).

The simulation of the LVT with an optimistic algorithm consists of two essential parts. The first part is the evolution of the system for some time, assuming there is no causality violation – it is the forward computations. The second part is the check of causality violation with rolling back those events, which violates the time order – it is the rollback computations.

To model the forward part in OLVT we use the same approach as in Refs. [12, 26, 13] computing the increment of LVT with the exponentially distributed random variable η_i with the mean equal to unity

$$\tau_i(t+1) = \tau_i(t) + \eta_i, \quad (1)$$

where $i = 1, 2, \dots, N$. The wall time t in the OLVT model is measured in the units of the Time Warp window [2].

To model the rollback part in OLVT, we choose randomly from the exponential distribution with the mean b the rollback depth value k and choose kN LPs randomly. For each chosen LP, check the rollback condition with the following law.

With probability p the LVT $\tau_j(t+1)$ of the chosen LP_j will remain the same with probability $(1-p)$ and will be updated with probability p

$$\tau_j(t+1) = \begin{cases} \tau_r(t+1), & \text{if } \tau_j(t+1) > \tau_r(t+1) \\ \tau_j(t+1), & \text{otherwise,} \end{cases} \quad (2)$$

where $\tau_r(t+1)$ is LVT of the one of the nearest neighbors chosen randomly. The LVT update corresponds to the event of the active mobile device moving from the cell r to the cell j .

We should stress that the rollback law in the OLVT model is different from the one in the optimistic LVT model [13], and use some ideas of the LVT model defined on the small-world network in Ref. [16].

4. Parameters and observables

4.1. Parameters in OLVT model

The parameter of OLVT model defined in line with the optimistic LVT model [13] as the ratio of the forward increment to the sum of the forward and rollback increments

$$q = \frac{1}{1+b}, \quad (3)$$

which takes the limiting values 1 and 0 while b varies from 0 to ∞ .

Analyzing the simple LVT evolution model in the paper [13] we found that at a low enough value of parameter q , the LVT profile becomes flat and does not increase the mean value. For a large enough value of the parameter q , the LVT profile becomes rough and propagates with the wall time t , and the mean value of the LVT profile becomes proportional to the wall time t . There is some critical value q_c separating the two regimes, the value of the roughening transitions known in statistical physics [27] (for details, we refer the reader to reviews [28, 29]). The critical feature of the roughening transition is that the LVT profile behavior near the critical value q_c is universal, giving the same law of the mean LVT profile dependence from the parameter q , and the same law of the mean LVT profile width dependence from the number of cells N . In the language of computer science, the first dependence is connected to the efficiency of simulations (the loading of the processing elements), and the second dependence is connected to the synchronization of the processing elements.

In addition to the parameter q , we already defined the parameter p for the OLVT model as the probability of the event describing the active mobile device moving to the nearest cell.

4.2. Parameters in PCS model

Let us introduce the parameter q'

$$q' = 1 - \frac{\# \text{ rollbacks}}{\# \text{ events}} \quad (4)$$

which defined through the relative number of rollbacks ($\# \text{ rollbacks}$) to the number of the processed events ($\# \text{ events}$).

As we will see, the parameter q' plays the same role as the parameter q in the OLVT model.

In the PCS model, the parameter p' has the same meaning as the parameter p in the OLVT model. We keep different notations for p, p' and q, q' because in PCS model parameters p' and q' is not set before simulations as in OLVT model and rather computed in PCS simulation. The parameters p' and q' depend on the value of ROSS parameter `MOVE_CALL_MEAN`, which is the average time between calls that leave the tower's coverage area.

4.3. Observables

We calculate time-dependent averages at each time step t and after simulations average them over $M = 10$ realizations of the process in the PCS model and $M = 1000$ in the OLVT model using the following definitions

- (i) The mean LVT profile

$$\langle \tau(t) \rangle = \frac{1}{M} \sum_{j=1}^M \left(\frac{1}{N} \sum_{i=1}^N \tau_i(t) \right)_j, \quad (5)$$

- (ii) The mean velocity

$$\langle v(t) \rangle = \frac{1}{M} \sum_{j=1}^M (\tau(t+1) - \tau(t))_j, \quad (6)$$

- (iii) The mean LVT profile square width

$$\langle w^2(t) \rangle = \frac{1}{M} \sum_{j=1}^M \left(\frac{1}{N} \sum_{i=1}^N [\tau_i(t) - \tau(t)]^2 \right)_j. \quad (7)$$

5. Simulations

5.1. The profile evolution

The PCS model was simulated on 64 cores of the computing node R2D26 computer. The number of logical processes and the number of kernel processes was $N = 256$. The parameter `MOVE_CALL_MEAN` sets the average time between calls moving from one network coverage area to another.

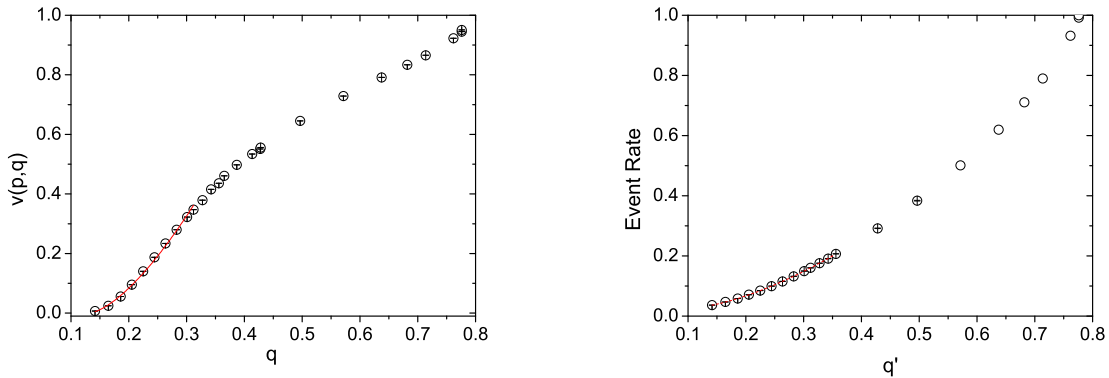
We use values of p', q' calculated in the PCS model simulations as the values of parameters p, q in OLVT simulations. The data presented in Table 1. Below, we use the notations $v(p, q)$ for the profile velocity in the OLVT model and similar notations for the profiles' square width.

The OLVT model was simulated on the R2D26 computer. We run each model with several logical processes $N = 256$ and different parameters p and q independently on the available cores to average over random realizations.

Mean OLVT profile velocity $\langle v(t) \rangle$ saturated after some transition time, forgetting the initial flat state. We name this velocity the steady-state velocity $v(p, q)$. The steady-state velocity does depend on the value of parameter q defined by expression 3) and is shown in Figure 1-a. In the figures, each point supplied the error bars, computed with the number of realizations. The size of error bars is typically less than the symbols. The velocity vanishes at some value

Table 1. Parameters of the simulations.

MOVE_CALL_MEAN	p'	q'
300	0.1821	0.24
450	0.1368	0.30
600	0.1078	0.36
800	0.0828	0.43
1000	0.0666	0.50
1250	0.0531	0.57
1500	0.0439	0.64
1750	0.0373	0.68
2000	0.0323	0.71
3000	0.0209	0.76
4000	0.0154	0.7759
4500	0.0135	0.7761

a) Steady-state velocity in OLVT as function of parameter q .b) Normalized event-rate in PCS model as function of parameter q' .**Figure 1.** Critical behaviour of the a) steady-state velocity $v(p, q)$ in OLVT model and b) normalized event-rate in PCS model.

q_c , which is the critical point of the roughening transition [28, 29]). The approximation of the v near the q_c for the values of $q < 0.33$ gives the following expression

$$v \sim (q - q_c)^\nu, \quad (8)$$

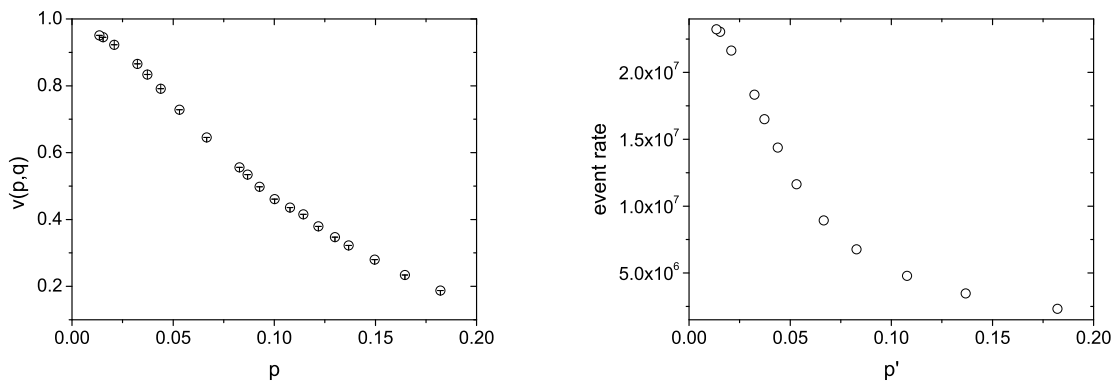
with $q_c \approx 0.132(5)$ (the number in parentheses is the statistical error of the fitting procedure) and $\nu \approx 1.50(1)$.

The normalized event-rate in the PCS model can be estimated as the number of events in the unity time divided by the number of maximum event rates. In a qualitative analogy with the steady-state velocity in the OLVT model, the normalized event-rate in the PCS model demonstrates the roughening transition. The fit to the data in Figure 1-b for the values of $q' < 0.33$ give the following approximation of the event-rate behavior near the roughening point

$$\text{Event rate} \sim (q' - q'_c)^{\nu'}, \quad (9)$$

with $q'_c \approx 0.101(2)$ and $\nu' \approx 1.54(3)$.

The qualitative analogy of the steady-state velocity in the OLVT model and the event rate become even better while plotting depends on them as functions of p and p' , correspondingly. In Figure 2 both the left and right curves drops done about five times in the same variation range of the parameters p and p' .



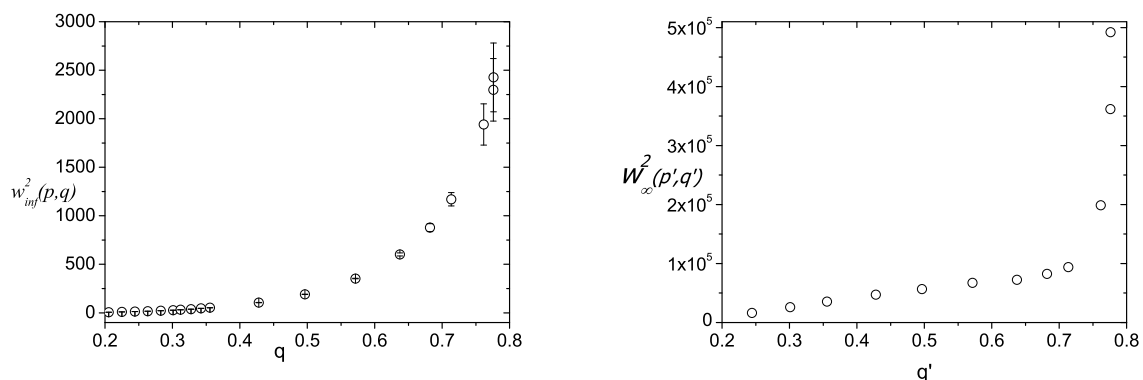
a) Steady-state velocity in OLVT as function of parameter p .

b) Event-rate in PCS model as function of parameter p' .

Figure 2. Behaviour of the a) steady-state velocity v in OLVT model and b) event-rate in PCS model.

5.2. The width of the profile evolution

The width of the profile can be associated with the level of processing elements synchronization. Figure 3 shows the dependence of the square width of the OLVT profile and the square width of the PCS profile. The name roughening of the transitions comes from the fact that below q_c (q'_c), both models have the flat profile $\langle w^2(\infty) \rangle$ after saturation, and for larger values of parameter q_c (q'_c) the width is finite, reflecting the rough random profile in both models.



a) The square width of LVT profile in OLVT model as function of parameter q .

b) The square width of LVT profile in PCS model as function of parameter q' .

Figure 3. The mean square width of LVT profiles in OLVT and PCS models.

We should note the similar qualitative behavior in the left and right panel of Figure 3.

6. Discussions

We introduce the model of the local virtual time (LVT) profile evolution in the PCS model, the OLVT model. The model reflects the essentials of the LVT evolution. The comparison of the direct simulation of the PCS model with the ROSS simulator and OLVT model demonstrates the qualitative analogy in the LVT profile evolution. Moreover, both models have the roughening transition, and the behavior of models near the transition belongs to the same universality class of the transition.

The paper's main results are 1) introduction of the OLVT model and 2) validation of the model by comparing simulation results for the LVT profile of both models.

7. Acknowledgments

The authors are thankful to Caitlin Ross, who helps with the ROSS code's part on the Kernel Processes' local virtual times output. We use the HPC facilities Science Center in Chernogolovka. LS and LZ acknowledge the Russian Ministry of Science and Higher Education's State Assignment through ITP and SCC, correspondingly.

References

- [1] Fujimoto R M 1990 *Commun. ACM* **33** 30
- [2] Carothers C D, Bauer D and Pearce S 2002 *J. Parallel Distr. Com.* 2002 **62** 1648–69
- [3] Weber D Time warp simulation on multi-core processors and clusters 2016 Master's thesis, University of Cincinnati, Cincinnati, OH
- [4] Santhi M, Eidenbenz S and Liu J 2015 *Proc. of the 2015 Winter Simulation Conf. (Huntington Beach)* ed. Yilmaz L et al. (New York City: IEEE press) pp 3013–24
- [5] Steinman J 1991 *Proc. SCS Western Multi-conf. on Advances in Parallel and Discrete Simulation (Anaheim)* vol 23, ed. Madiseti V et al. p 1111
- [6] Cardoen B, Manhaeve S, Van Tendeloo Y and Broeckhove J 2018 *Simulation* **94(4)** 281–300
- [7] Henderson T R, Lacage M, Riley G F, Dowell C and Kopena J 2008 *SIGCOMM demonstration* **14(14)** 527
- [8] Riley G F and Henderson T R 2010 *Modeling and tools for network simulation* 15–34.
- [9] Barnes Jr P D, Carothers C D, Jeferson D R and LaPre J M 2013 *Proc. of the 1st ACM SIGSIM Conf. on Principles of Advanced Discrete Simulation (Montreal)* (Association for Computing Machinery: New York) pp 327–36
- [10] Fujimoto R M, Carothers C D, Ferscha A, Jefferson D, Loper M, Marathe M and Taylor S J 2017 *Proc. of the 2017 Simulation Conf. (Las Vegas)* (New York: IEEE) p 431
- [11] Balci O, Fujimoto R M, Goldsman D, Nance R E, and Zeigler B P 2017 *Proc. of the 2017 Simulation Conf. (Las Vegas)* (New York: IEEE) pp 821–836
- [12] Korniss G, Toroczka Z, Novotny M A and Rikvold P A 2000 *Phys. rev. lett.* **84(6)** 1351
- [13] Ziganurova L, Novotny M A and Shchur L 2016 *J. Phys. Conf. Ser.* **681** 012047
- [14] Korniss G, Novotny M A, Guclu H, Toroczka Z and Rikvold P A 2003 *Science* **299(5607)** 677-79
- [15] Ziganurova L and Shchur L N 2018 *Phys. Rev. E* **98(2)** 022218
- [16] Shchur L N and Ziganurova L F 2019 *J. Exp. Theor. Phys.* **129(4)** 722–32
- [17] Carothers C D, Fujimoto R M, Lin Y B and England P 1994 *Proc. of International Workshop on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (Durham)* (New York City: IEEE press) pp 2-6
- [18] Carothers C D, Fujimoto R M and Lin Y-B 1995 *ACM SIGSIM Simulation Digest* **25(1)** 87–94
- [19] Lin Y-B and Fishwick P A A 1996 *IEEE Trans. Syst. Man Cybern. Syst.* **26(4)** 397–412
- [20] Jefferson D R 1985 *ACM Transactions on Programming Languages and Systems (TOPLAS)* **7** 404
- [21] Wilsey P A 2016 *Proc. of the 2016 ACM SIGSIM Conf. on Principles of Advanced Discrete Simulation (Banff Alberta)* (New York: ACM) pp 165–76
- [22] Bauer Jr D W, Carothers C D and Holder 2009 *Proc. of the 2009 ACM/IEEE/SCS 23rd Workshop on Principles of Advanced and Distributed Simulation (Lake Placid, New York)* (Washington: IEEE Computer Society) pp 35–44
- [23] Williams B, Ponomarev D, Abu-Ghazaleh N and Wilsey P 2017 *Proc. of the 2017 ACM SIGSIM Conf. on Principles of Advanced Discrete Simulation (Singapore)* (New York: ACM) pp 121–32
- [24] Jagtap D, Abu-Ghazaleh N and Ponomarev D. 2012 *IEEE 26th Int. Parallel and Distributed Processing Symposium (Shanghai)* (New York: IEEE Press) pp 520–53

- [25] Eker A, Williams B, Mishra N, Thakur D, Chiu K, Ponomarev D and Abu-Ghazaleh N 2018 *Proc. of the 22nd International Symposium on Distributed Simulation and Real Time Applications (Madrid)* (New York: IEEE Press) pp 87–96
- [26] Shchur L N and Novotny M A 2004 *Phys. Rev. E* **70**(2) 026703
- [27] Grassberger P 1982 *Zeitschrift für Physik B Condensed Matter* **47** 365
- [28] Odor G 2004 *Rev. Mod. Phys.* **76** 663
- [29] Hinrichsen H 2000 *Brazilian J. Phys.* **30** 69