Rough Set Based Efficiency Improvement of Simulation Performance Prediction

László Muka and István Derka
Széchenyi István University, Department of Telecommunications, Győr, Hungary
e-mails: muka@sze.hu, steve@sze.hu

***Abstract*— Parallel and distributed simulation method (PADS) has been showing growing importance in the discrete event simulation (DES) analysis of large-scale, complex network systems. To realize a good performance with the PADS method, it is necessary to predict the behavior of the simulation model in the parallel and distributed execution environment. This paper describes a Rough Set Theory based method of simulation performance prediction efficiency improvement using Systems Performance Criteria (efficacy, efficiency and effectiveness). The new improved performance prediction method – Algorithm of Improved Simulation Performance Prediction (AISPP) – is presented as addition to the traditional Coupling Factor Method (CFM) prediction approach.**

***Keywords*— Parallel discrete event simulation, simulation performance prediction, Rough Set Theory, Systems Performance Criteria, Coupling Factor Method**

I. Introduction and Motivation

Over the last few years, together with the increase of the need for the discrete event simulation (DES) analysis of large-scale, complex systems and networks requiring high computing capacity, a lot of efforts have been made in research of parallel and distributed discrete event simulation modelling and execution methods [1-4], since the parallel and distributed execution simulation turned out to be an appropriate approach to improve simulation runtime performance [5, 6].

The parallel and distributed simulation can be defined in different ways. According to a simpler but broader view, parallel and distributed simulation is any simulation in which more than one processor is used [[3]. According to another accepted definition, the Parallel Discrete Event Simulation (PDES) [6] and the Parallel and Distributed Simulation (PADS) [3] are differentiated on the feature, whether the single simulation model is executed on a set of *tightly* coupled processors (e.g. a shared memory multiprocessor) or on a set of distributed *loosely* coupled processors (e.g. PCs interconnected by LAN or WAN). According to a simple view, parallel and distributed simulation (PADS) is any simulation in which more than one processor is used [3]. In the present paper, PADS is defined as the execution of a single discrete event simulation model on high performance computing platforms (like clusters of homogeneous and heterogeneous computers), and on emerging platform environment (WEB, grid and cloud).

Despite of its beneficial effect on the simulation runtime performance, the PADS method is not in the everyday use in the simulation community, because *development* of simulation models having high runtime performance features in a parallel and distributed execution environment remained a hard task even today [7]. Simulation *performance prediction* methods and tools can help to realize higher performance with the PADS model development by providing *preliminary* knowledge about the likely *behavior* of the model [7-11]. The PADS performance prediction methods should support the performance analysis throughout the development and evaluation process [7].

Summary of the overview above can be as follows: uncertainty of data necessary for the simulation performance modelling and prediction is present in every phase of the simulation process, and both the uncertainty and cost of obtaining data is increasing with the distance (measured in simulation cycle-steps) of prediction from the simulation execution.

The *motivation* of the authors to make the research presented in the paper was the lack of methods which can manage together the accuracy of performance predictions – the basic requirement for performance prediction – and the cost of data observations necessary for parallel and/or distributed simulation performance prediction.

For the new method presented in the paper the *Rough Set Theory (RST) method* and the systems approach of performance – *Systems Performance Criteria (SPC) efficacy, efficiency and effectiveness)* have been selected to handle together the accuracy and the cost of predictions in a train-and-test analysis.

In the present paper, the authors made the following major *contributions*:

The RST model of simulation performance prediction has been defined allowing train-and-test analysis of performance predictions

A set of *operations and measures* are given for the use of *efficacy (E1), efficiency (E2)* and *effectiveness* (E3) to measure for a single and for a series of predictions in order to embed SPC into RST train-and-test process.

The algorithm of the new *Algorithm of Improved Simulation Performance Prediction (AISPP)* has been described. Including the *Coupling Factor Method (CFM)* of the parallel and/or distributed simulation performance prediction in the AISPP process a feedback is provided to support the model identification and refinement stage.

The rest of the paper is organized as follows. The RST, SPC and CFM approaches are introduced in Section 2. In Section 3, the new prediction improvement method is formulated: first definitions of E1, E2 and E3 are given for calculations and attribute and rule dropping then the new AISPP process is described. Section 4 concludes the work.

II. Ingredients of the Prediction Efficiency Improvement Approach

A. *Rough Sets*

The RST (Rough Set Theory) is a mathematical framework particularly suitable for modelling and analysis of information systems with imprecise relations, with uncertain, vague data [25-29].

A *rough set information system* with embedded knowledge consists of two sets: the set of *objects* called the *universe* and the set of *attributes*.

More formally, $I = (U, A, f, V)$ denotes an *information system* of RST, where set U is the universe, A is the set of attributes. Sets U and A are finite nonempty sets where ($U = \{x_1, x_2, x_3, \dots x_{|U|}\}$ and $A = \{a_1, a_2, a_3, \dots a_{|A|}\}$). The attributes define a *transformation function* $f: U \to V$ for U where the set V is the set of values of A ($V = V_{a_1} \cup V_{a_2} \cup V_{a_3} \cup \dots \cup V_{a_{|A|}}$). The set V_{a_i}– named also the *domain* of a_i – contains the collection of values of a_i and $V_{a_i} = \left\{v_{1_{a_i}}, v_{2_{a_i}}, v_{3_{a_i}}, \dots, v_{|V_{a_i}|_{a_i}}\right\}$ where $|V_{a_i}|$ is the size of the domain of a_i.

Discretization is the operation of mapping the primary values and ranges of all attributes to *selected* (possibly optimized) sets of discrete values: $f_{V'}: V' \to V$. The set V' stands for the values of a before discretization.

The *B-indiscernibility relation* $IND(B)$ for a set of attributes $B \subseteq A$is defined in the following way:
$IND(B) = \{(x_i, x_j) \in U^2 | \forall (a \in B)(a(x_i) = a(x_j))\}$
If $(x_i, x_j) \in IND(B)$, then the objects x_i and x_j are indiscernible from each other in B and the *equivalence classes* $[x]_{IND(B)}$ of $IND(B)$ are formed by the objects indiscernible in B.
Rough sets are defined by their lower approximation and upper approximation sets. The set $B^*(X)$ and the set $B_*(X)$ is the B*-lower* and B*-upper approximation* of the set X and defined as follows:

$$B_*(X) = \bigcup_{x \in U} \{x | [x]_{IND(B)} \subseteq X\}$$

$$B^*(X) = \bigcup_{x \in U} \{x | [x]_{IND(B)} \cap X \neq \emptyset\}$$

The set $BN_B(X)$ defined by the equation $BN_B(X) = B^*(X) \setminus B_*(X)$ is the B*-boundary region* of X. If X is a crisp set then, $X = B_*(X)$ thus $BN_B(X) = \emptyset$ which means the boundary region is empty.

A *reduct* R_B is the minimal subset of attributes B that allows the same classification of objects of U as the set of attributes B. This feature of a reduct may be described by indiscernibility function as follows:
$IND_{\forall x (x \in U)}(R_B) = IND_{\forall x (x \in U)}(B),\ B \subseteq A$.

In general, the information system may take the form of $I = (U, A = C \cup D, f, V)$ which is a *decision information system (DIS)*. The set $C = \{c_1, c_2, c_3, \dots, c_{|C|}\}$ denotes the set of condition attributes and D is the set of decision attributes $D = \{d_1, d_2, d_3, \dots, d_{|D|}\}$. The information function $f: U \to V$ may be expressed by information functions $f_C: C \to V_C$ and $f_D: D \to V_D$, where $V = V_C \cup V_D$ ($V_C = V_{c_1} \cup V_{c_2} \cup V_{c_3} \cup, \dots, \cup V_{c_{|C|}}$ and $V_D = V_{d_1} \cup V_{d_2} \cup V_{d_3} \cup, \dots, \cup V_{d_{|D|}}$) and

$V_C = \bigcup_{i=1}^{|C|} V_{c_i}$ where $V_{c_i} = \left\{v_{1_{c_i}}, v_{2_{c_i}}, v_{3_{c_i}}, \dots, v_{|V_{c_i}|_{c_i}}\right\}$

and $V_D = \bigcup_{i=1}^{|D|} V_{d_i}. V_{d_i} = \left\{v_{1_{d_i}}, v_{2_{d_i}}, v_{3_{d_i}}, \dots, v_{|V_{d_i}|_{d_i}}\right\}$.

In a *decision table* $I = (U, A = C \cup \{d\}, f, V)$ based on a DIS, d denotes the *distinguished decision attribute*. Furthermore, a decision information system having the form of $I = (U, C \cup D, f_{V'}, f, V', V)$ denotes a DIS with *discretization information function* $f_{V'}$: $V' \to V$ (which is identical to information functions $f_{V'_C}: V'_C \to V_C$ and $f_{V'_D}: V'_D \to V_D$).

The *classification* may also be described by *a set decision rules* $(S = \{s_1, s_2, s_3, \dots, s_{|S|}\})$ in the form of implication ($s_i = (\varphi_i \Rightarrow \kappa_i), s_i \in S$), where φ_i and κ_i are *logical expressions* of the condition and decision attributes respectively. The formulas φ_i and κ_i may also be quoted as *LHS (Left Hand Side)* and *RHS (Right Hand Side)* part of the rule. A decision rule s_i may be evaluated using its *coverage* and *accuracy* ratios according to formulas

$$Coverage_U(s_i) = \frac{|Match_U(s_i)|}{|U|}$$

$$Accuracy_U(s_i) = \frac{|Supp_U(s_i)|}{|Match_U(s_i)|}$$

where $Match_U(s_i)$ is the number of objects in U the attribute values of which satisfy φ_i (*matching* with the *LHS* part of s_i), and $Supp_U(s_i)$ denotes the number of objects in decision table the attribute values of which satisfy both φ_i and κ_i (matching both with the *LHS and RHS* parts of s_i).

B. *Systems Performance Criteria*

The three *Systems Performance Criteria (SPC)* are the efficacy (E1), efficiency (E2) and effectiveness (E3) [11,30,31]. The SPC are in a hierarchy-like relationship with each other. On the longer term, the performance of a system is checked by the effectiveness criterion, the efficacy criterion shows whether the performance is suitable at all, and the efficiency criterion characterizes the relation of the required output and the resources used to produce the output.

C. *The Coupling Factor Method*

Based on some theoretical considerations about the connectedness of PADS model segments, paper [14] describes a practical simulation performance prediction approach the *Coupling Factor Method (CFM)*. The method – using results that have been got in sequential simulation runs – predicts the *parallelization potential* of simulation models (for models with conservative null message-based algorithm) and formulates requirement on how this potential can be exploited.

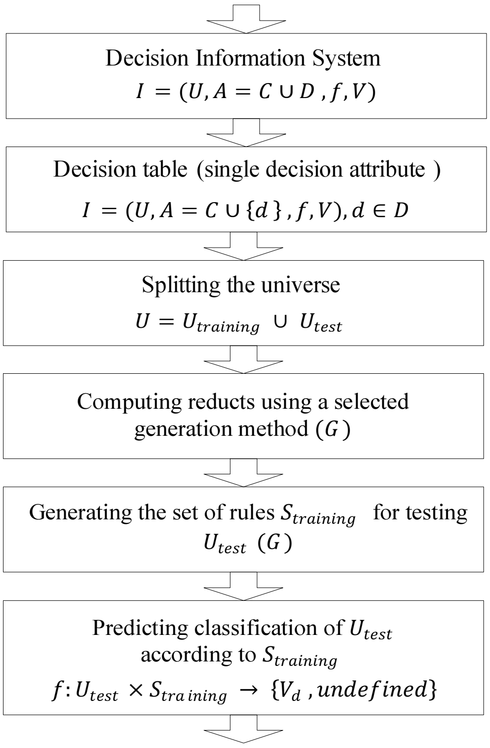

Figure 1. Traditional RST (TRSTA) algorithm for train-and-test analysis and learning

The principle of CFM may be summarized in an inequity:

$L * Ev \gg \tau * P$

where L is the lookahead value characterizing the model (simsec), E is the event density generated by the model (ev/simsec), τ is the latency of messages between logical process (LPs) of the model (sec), and P is the event processing computation hardware performance (ev/sec). In this practical approach, parameters L and E characterise the model itself, parameters τ and P describe the execution environment. According to the method, the coupling factor λ is calculated according to the formula $\lambda = L * E/\tau * P$. The high value of the coupling factor λ shows the good potential for simulation model parallelization. The method involves only four parameters for the performance prediction calculations. These parameters can be measured in simple *sequential simulation runs*. For a separate process, the λ_N parallelization potential of a process is only a part of the whole potential:

$\lambda_N = \frac{\lambda}{N_{LP}} = \frac{1}{N_{LP}} * \frac{L*E}{\tau*P}$

where N_{LP} the number of the LPs [15].

The method has been validated by a series of simulation experiments for *homogeneous* and *heterogeneous* clusters of computers [15-17].

Examples for the telecommunications networks and cloud computing systems are introduced in [16,32].

III. THE NEW PREDICTION EFFICIENCY IMPROVEMENT METHOD

The new method is built around the *Traditional RST Analysis Algorithm (TRSTA)* and the *E1, E2 and E3 Systems Performance Criteria*. (Steps of TRSTA *train-and-test* analysis process are shown in Figure 1.)

For the simulation *performance prediction efficiency improvement,* SPC can be identified as follows: to achieve the necessary prediction quality (E1 criterion), to realize it with an acceptable cost (E2 criterion) and to produce it in a stable manner on long run (E3 criterion).

A. *Prediction performance calculations and dropping criteria*

In case of a simulation performance analysis o*bjects* of the universe are *computer simulation experiments*. In a decision table $I = (U, A = C \cup \{d\}, f, V)$ attributes A describes the explanatory (independent) and dependent variables, decision rules $S = \{s_1, s_2, \ldots, s_{|S|}\}, s_i = \varphi_i \Rightarrow \kappa_i$ describes the classification of the object of the experiments.

In the following, functions are defined allowing to calculate the E1, E2 and E3 criteria and supporting the increase of performance by dropping attributes and rules.

DEFINITION 1. CLASSIFICATION PREDICTION FUNCTION

$$f_{(U_{test}, S_{training})}: U_{test} \times S_{training} \rightarrow \{V_d, undefined\}$$

$$f_{([c](x_l), \varphi(s_k))} = d(x_l)_{predicted} = \begin{cases} \langle d(s_k) \rangle & \left| \begin{array}{c} (Match_{U_{test}}(s_k) = 1, (s_k \in S_{training}), \\ \langle d(s_k) \rangle = v_d \in V_d \end{array} \right. \\ & undefined \mid othervise \end{cases}$$

DEFINITION 2. MATCHING PREDICTION OPERATOR

$$\forall_{l=1}^{|U_{test}|}(x_l)(x_l \in U_{test})([\langle c_{l,1} \rangle, \langle c_{l,2} \rangle \ldots \langle c_{l,|C|} \rangle] \xRightarrow{\forall_{k=1}^{|S_{training}|} Match\,(s_k)} [d(x_l)_{predicted}])$$

DEFINITION 3. PREDICTION CORRECTNESS

Prediction correctness $p(x_l)$ is

$$\forall\,(x_l)(x_l \in U_{test}) \left(p(x_l) = \begin{cases} 1, & if \langle d\,(x_l) \rangle_{predicted} = \langle d\,(x_l) \rangle_{observed} \\ & 0, othervise \end{cases} \right)$$

where $\langle d(x_{()}) \rangle_{predicted,(s)} (s \in S_{training}) = v_d, v_d \in V_d$

DEFINITION 4. EFFICACY CRITERION OF PREDICTION

$E1 = \frac{\sum_{l=1}^{|U_{test}|} p(x_l)}{|U_{test}|} \geq E1_{limit} > 0.5$ (The efficacy of prediction is required to be better than random guess.)

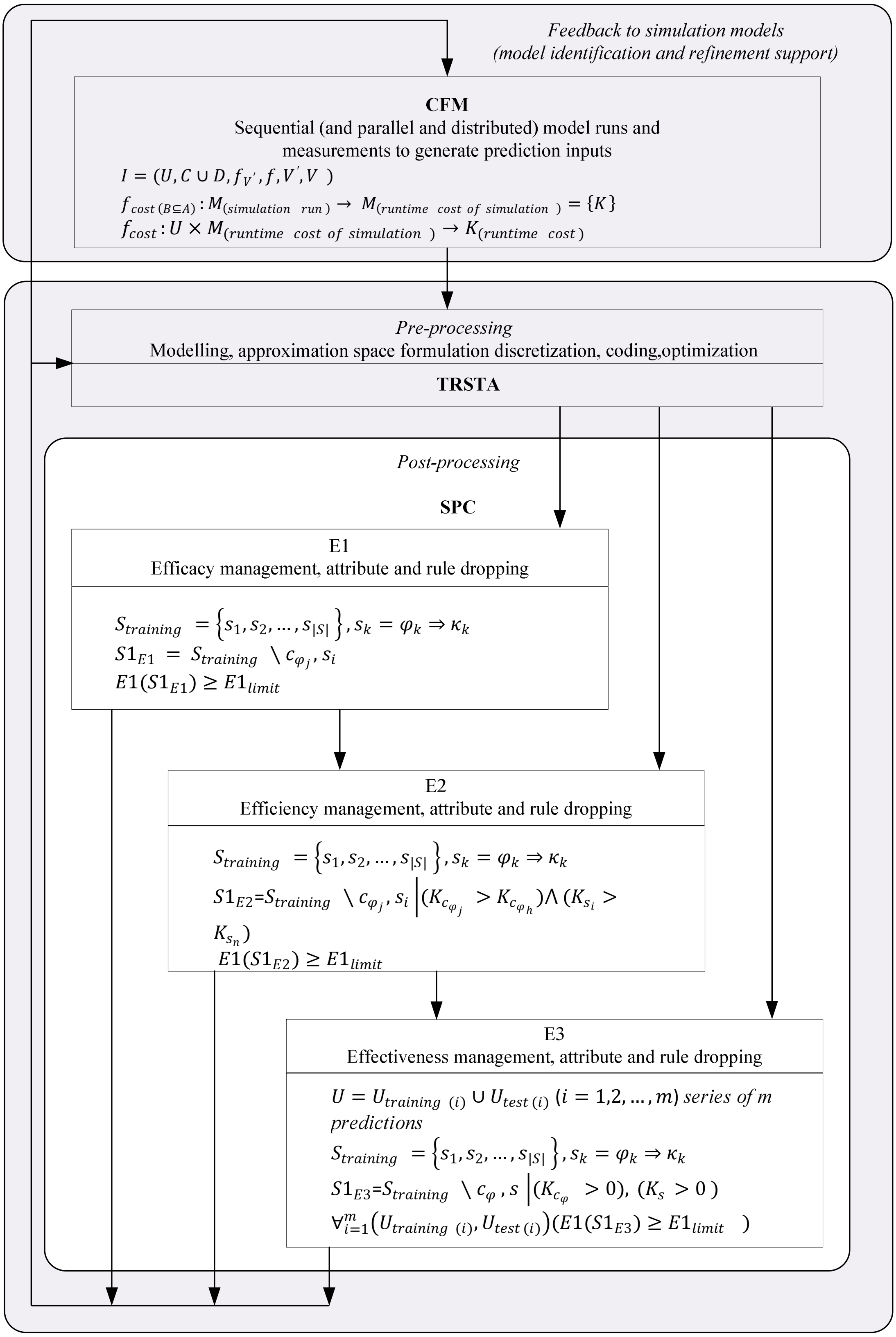

Figure 2. The Algorithm of Improved Simulation Performance Prediction (AISPP)

DEFINITION 5. THE COST ALLOCATION FUNCTION

The f_{cost} allocates the runtime costs of simulation in the information system

$f_{cost}: U \times M_{(runtime\ cost\ of\ simulation\ )} \rightarrow K_{(runtime\ cost)}$

DEFINITION 6. COST OF ATTRIBUTES, EXPERIMENTS AND RULES

Cost of an *attribute* a_i is defined as $K_{a_i} = \sum_{l=1}^{|U|} K_{a_i}(x_l)\ \ [sec], (a_i \in A)$, (cost of an attribute may also take the value of $K_{a_i}\ = 0\ \ [sec]$).

In a decision table, the cost of an *experiment* x_l is determined as

$K_{x_l} = \sum_{i=1}^{|C|} K_{x_l}(c_i)_{x_l}\ \ + K_{x_l}(d)_{x_l}\ [sec], (x_l \in U)$.

Cost of a *rule* s_i is calculated as

$K_{s_i} = \sum_{\forall(x_l)|Match(s_i)=1} K_{x_l}\ \ [sec], (s_i \in S_{training})$.

DEFINITION 6. EFFICACIOUS DROPPING OF ATTRIBUTES AND RULES

$S_{training} = \left\{s_1, s_2, \dots, s_{|S|}\ \right\}, s_k = \varphi_k \Rightarrow \kappa_k$

$S1_{E1}\ =\ S_{training} \setminus c_{\varphi_j}, s_i$

$E1(S1_{E1}) \geq E1_{limit}$

DEFINITION 7. EFFICIENCIENT DROPPING OF ATTRIBUTES AND RULES

$S_{training} = \left\{s_1, s_2, \dots, s_{|S|}\ \right\}, s_k = \varphi_k \Rightarrow \kappa_k$

$S1_{E2}$=$S_{training} \setminus c_{\varphi_j}, s_i \left| (K_{c_{\varphi_j}}\ > K_{c_{\varphi_h}}) \wedge (K_{s_i} > K_{s_n}) \right.$

$E1(S1_{E2}) \geq E1_{limit}$

DEFINITION 8. EFFECTIVE DROPPING OF ATTRIBUTES AND RULES

$U = U_{training(i)} \cup U_{test(i)}\ (i = 1,2, \dots, m)$ *series of m predictions*

$S_{training} = \left\{s_1, s_2, \dots, s_{|S|}\ \right\}, s_k = \varphi_k \Rightarrow \kappa_k$

$S1_{E3}$=$S_{training} \setminus c_\varphi\ , s\ \left| (K_{c_\varphi}\ > 0), (K_s > 0\ ) \right.$

$\forall_{i=1}^{m} \left(U_{training(i)}, U_{test(i)}\right)(E1(S1_{E3}) \geq E1_{limit}\ \ )$

B. *The AISPP process*

Figure 2 shows the process diagram of the *new RST-based method* – the *Algorithm of Improved Simulation Performance Prediction (AISPP)*.

The *tactic* of the RST-based prediction improvement method can be formulated as follows: (1) all the data – got from sequential simulation and PADS (with different parameters – processor number, λ, etc.) – have to be analyzed which supposed to have influence on CFM prediction results. (2) CFM data, attributes and objects of an RST decision information system, are processed in a train-and-test examination process. (3) Using the defined SPC based evaluation of prediction performance criteria in the TRST train-and-test analysis, a feedback is realized to data collection and measurement and modelling steps of simulation performance prediction (to model identification and refinement stage too).

AISPP process has the following features:

- the operations are applied with a traditional RST analysis method (TRST) and a traditional simulation performance prediction method (CFM)
- for the predictions, both the sequential and parallel and/or distributed simulation runs (if any) are used in the RST model
- the method functions in interactive manner using a TRST
- the *operations* that have been defined are for the analysis of the *efficacy (E1)*, *efficiency* (E2) and *effectiveness* (E3) in the TRST process
- the method supports making *feedback* for model feature *identification and refinement support* of the simulation performance prediction models (or to the simulation model)

(The method can be implemented by using the OMNet++ DES software [12] and the ROSETTA Rough Set Software System [13].)

IV. CONCLUSIONS

For the improvement of the simulation performance prediction of a *parallel and/or distributed simulation model execution*, a new methodology, based on RST approach in order to work with unreliable, imprecise preliminary data, has been introduced.

For the new methodology, the set of necessary operations has been created:

- Operations, based on Systems Performance Criteria (SPC) of *efficacy (E1)*, *efficiency (E2)* and *effectiveness (E3)* of predictions
- Operations for efficacious and efficient predictions attribute and rule dropping in predictions (for efficiency evaluation, the cost of attributes and cost of rules have been defined) and the effective attribute and rule droppings for a series of predictions too.

The algorithm of the new methodology has also been presented (Algorithm of Improved Simulation performance Prediction (AISPP)) with its connections to a traditional RST method (TRSTT) including approximation space optimization in pre-processing phase and embedding SPC in the post-processing. The new methodology provides feedback to the simulation performance model (and to the simulation model too) to support model features identification and refinement.

REFERENCES

[1] Perumalla, K. S., 2010. "μπ: a scalable and transparent system for simulating MPI programs", In Proceedings of the 3rd International ICST Conference on Simulation Tools and Techniques, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering). pp. 62.

[2] Perumalla, K. S., 2006. "Parallel and distributed simulation: traditional techniques and recent advances", In Proceedings of the 38th Winter Simulation Conference, pp. 84-95.

[3] D'Angelo, G., 2011. "Parallel and Distributed Simulation from Many Cores to the Public Cloud (Extended Version)", In: Proceedings of the 2011 International Conference on High Performance Computing and Simulation (HPCS 2011), Istanbul (Turkey), IEEE, July 2011. ISBN 978-1-61284-382-7. http://dx.doi.org/10.1109/HPCSim.2011.5999802 arXiv preprint arXiv:1105.2301. pp.1-9.

[4] Yoginath, S. B.; Perumalla, K. S., 2013. "Empirical evaluation of conservative and optimistic discrete event execution on Cloud and VM platforms", In Proceedings of the 2013 ACM SIGSIM conference on Principles of advanced discrete simulation, pp. 201-210.

[5] Fujimoto, R. M., 1990b. "Performance of Time Warp under Synthetic Workloads,"in Proceedings of the SCS Multiconference on Distributed Simulation,San Diego, CA, USA, 17-19 January, 1990, pp. 23-28.

[6] Fujimoto, R.M., 2000. "Parallel and Distributed Simulation Systems", John Wiley & Sons, USA, ISBN: 0-471-18383-0

[7] Kunz, G.; Tenbusch, S.; Gross, J.; Wehrle, K. 2011. "Predicting Runtime Performance Bounds of Expanded Parallel Discrete Event Simulations", In Modeling, Analysis & Simulation of Computer and Telecommunication Systems (MASCOTS), 2011 IEEE 19th International Symposium on. IEEE, pp. 359-368.

[8] Juhasz, Z.; Turner, S.; Kuntner, K.; Gerzson, M., 2003. "A Performance Analyser and Prediction Tool for Parallel Discrete Event Simulation", International Journal of Simulation, vol. 4, no. 1, May 2003, pp. 7-22.

[9] Muka, L.; Lencse, G., 2008. "Cooperating Modelling Methods for Performance Evaluation of Interconnected Infocommunication and Business Process Systems", In Proceedings of the 2008 European Simulation and Modelling Conference (ESM'2008), (Le Havre, France, Oct. 27-29.) EUROSIS-ETI, pp. 404-411.

[10] Muka, L.; Lencse, G., 2011. "Method for Improving the Efficiency of Simulation of ICT and BP Systems by Using Fast and Detailed Models", Acta Technica Jaurinensis, Vol.5 No. 1. 2012., pp. 31-42.

[11] Muka, L.; Benko, B. K., 2013. "Meta-level performance management of simulation: The problem context retrieval approach", Periodica Polytechnica, Electrical Engineering and Computer Science, 55(1-2), pp. 53-64, DOI: 10.3311/pp.ee.2011-1-2.06

[12] Varga, A.; Hornig, R., 2008. "An overview of the OMNeT++ simulation environment", In Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), p. 60.

[13] Komorowski, J.; Øhrn, A.; Skowron, A., 2002. "The ROSETTA Rough Set Software System", In Handbook of Data Mining and Knowledge Discovery, W. Klösgen and J. Zytkow (eds.), Oxford University Press, ch. 2-3, ISBN 0-19-511831-6.

[14] Varga, A.; Sekercioglu, Y. A.; Egan, G. K. 2003. "A practical efficiency criterion for the null message algorithm", Proceedings of the European Simulation Symposium (ESS 2003), (Oct. 26-29, 2003, Delft, The Netherlands) SCS International, pp. 81-92.

[15] Lencse, G.; Varga, A., 2010. "Performance Prediction of Conservative Parallel Discrete Event Simulation", Proceedings of the 2010 Industrial Simulation Conference (ISC'2010) (Budapest, Hungary, 7-9, June, 2010.) EUROSIS-ETI, pp. 214-219.

[16] Lencse, G.; Derka I.; Muka, L., 2013. "Towards the efficient simulation of telecommunication systems in heterogeneous execution environments", In proceedings of: TSP2013, Edited by Norbert Herencsar, Karol Molnar, the 36th International Conference on Telecommunications and Signal Processing, Rome, Italy, pp. 304-310.

[17] Lencse, G.; Derka, I., 2013. "Testing the Speed-up of Parallel Discrete Event Simulation in Heterogeneous Execution Environments", In proceeding of: ISC'2013, Edited by Veronique Limere and El-Houssaine Aghezzaf, ISBN 978-90-77381-76-2, 11th Annual Industrial Simulation Conference, Ghent University, Ghent, Belgium, pp.101-107.

[18] Lencse, G., 1998. "Efficient Parallel Simulation with the Statistical Synchronization Method", Proceedings of the Communication Networks and Distributed Systems Modeling and Simulation (CNDS'98) (San Diego, CA. Jan. 11-14). SCS International, pp. 3-8.

[19] Lencse, G., 1999a. "Applicability Criteria of the Statistical Synchronization Method", Proceedings of the Communication Networks and Distributed Systems Modeling and Simulation (CNDS'99), San Francisco, CA. (1999. Jan. 17-20) SCS International, pp. 159-164.

[20] Lencse, G., 1999b. "Design Criterion for the Statistics Exchange Control Algorithm used in the Statistical Synchronization Method", Proceedings of the Advanced Simulation Technologies Conference (ASTC 1999) part of the 32nd Annual Simulation Symposium (San Diego, CA. April 11-15) SCS International, pp. 138-144.

[21] Pawlak, Z., 1998. "Rough set theory and its applications to data analysis",Cybernetics & Systems, 29(7), pp. 661-688.

[22] Krishnaswamy, S.; Zaslavsky A.; Loke, S. W., 2002. "Predicting run times of applications using rough sets", In Ninth International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU 2002), pp. 455-462.

[23] Song, C.; Guan, X.; Zhao, Q.; Ho, Y. C., 2005. "Machine learning approach for determining feasible plans of a remanufacturing system", Automation Science and Engineering, IEEE Transactions on, 2(3), pp. 262-275.

[24] Muka, L.; Derka, I., 2013. " Improving Performance Prediction of Parallel and Distributed Discrete Event Simulation: A Rough Sets-based Approach", In proceeding of: ISC'2013, Edited by Veronique Limere and El-Houssaine Aghezzaf, ISBN 978-90-77381-76-2, 11th Annual Industrial Simulation Conference, Ghent University, Ghent, Belgium, pp. 95-100.

[25] Pawlak, Z.; 1982. "Rough sets", International Journal of Parallel Programming, 11(5), pp. 341-356.

[26] Pawlak, Z.; Skowron, A., 2007. "Rudiments of rough sets", Information sciences, 177.1, pp. 3-27.

[27] Zhang, J.; Li, T., Ruan, D.; Gao, Z.; Zhao, C., 2012. "A parallel method for computing rough set approximations", Information Sciences,194.2, pp. 209-223.

[28] Bazan, J. G.; Szczuka, M, 2005. "The rough set exploration system", In Transactions on Rough Sets III, , Springer Berlin Heidelberg, pp. 37-56.

[29] Suraj, Z., 2004. "An Introduction to Rough Set Theory and Its Applications", ICENCO, Cairo, Egypt, pp. 1-39.

[30] Gregory, F., 1993. "Cause, effect, efficiency and soft systems models", Journal of the Operational Research Society, vol. 44 (4), pp. 333-344.

[31] Checkland, P., 2000. "Soft systems methodology: a thirty year retrospective", Systems Research and Behavioral Science,17, S11-S58.

[32] Muka,L.; I Derka, I.; 2014. "Simulation Performance Prediction in Clouds", In: Orosz Gábor Tamás (szerk.), 9th International Symposium on Applied Informatics and Related Areas – AIS2014., Konferencia helye, ideje: Székesfehérvár, Magyarország, 2014.11.12 Székesfehérvár, Óbudai Egyetem, pp. 142-147., ISBN:978-615-5460-21-0**Error! Reference source not found.**

BIOGRAPHIES

LÁSZLÓ MUKA graduated in electrical engineering at the Technical University of Lvov in 1976. He got his special engineering degree in digital electronics at the Technical University of Budapest in 1981, and became a university doctor in architectures of CAD systems in 1987. Mr. Muka finished an MBA at Brunel University of London in 1996. Since 1996 he has been working in the area of modeling and simulation of infocommunication systems, which may include human subsystems too. Mr. Muka got his PhD at the Budapest University of Technology and Economics in 2011. Dr. Muka is working as an Associate Professor for the Széchenyi István University.

ISTVÁN DERKA received his Msc at Faculty of Electrical Engineering and Informatics at the Technical University of Budapest in 1995. He worked for the Department of Informatics from 1999 to 2003 and since then has been working for Department of Telecommunications, Széchenyi István University in Győr. He teaches Programming of communication systems and Interactive TV systems. He is an Assistant Professor. The area of his research includes multicast routing protocols and IPTV services in large scale networks.