Meta-level Performance Management of Simulation of Organizational Information Systems: The Problem Context State Approach

László Muka, Gábor Lencse

Abstract—**Simulation has become a frequently used tool for the analysis of ICT and BP systems and for fitting the features of these systems with each other and with the goals of the enterprise. For example, the change management of ERP (Enterprise Resource Planning) systems is a significant generator of the need for the common analysis of ICT/BP systems and the use of simulation may play crucial role in their analysis. The paper formulates the problem context state approach to the meta-level performance management of simulation in the form of efficiency management principles. The formulation is based on the investigation of the features of the dynamic behavior of problem contexts – using the 4-state and 2-state models of problem context types – for the common modeling and simulation of organizational ICT/BP systems. The process of the occurrence and elimination of the methodological gap is explained too.**

Index Terms—**efficiency of simulation, problem context state model, efficiency principles, ICT and BP systems, efficiency management**

I. Introduction

SIMULATION has been accepted as an appropriate tool for the analysis of ICT and BP systems and for fitting the features of these systems with each other and with the goals of the business.

Examining in an organizational environment, the *simulation process* is a *participative* and *collaborative* process with many *participants* [10]. Sierhuis and Selvin define the *simulation process* as a *holon*[1] in terms of Soft Systems Methodology (SSM, [2]) [11]. As the system approach to the simulation, the *simulation methodology* may be defined as a *structured set of methods* applied by a *HAS* (Human Activity System, [2]) performing the process of simulation. In an organizational environment, the process of simulation may also be treated as a *project process* with predefined goals aimed to be reached within time and cost limits with prescribed quality requirements. The *phase* of the simulation process is determined by the method of the simulation methodology being executed. *Discrete-Event Simulation* (DES) is a frequently used method for the analysis of the ICT and BP systems [5].

Simulation projects aimed at supporting the analysis and design of the *dynamic behavior* of organizational ICT (Information and Communication Technology) systems and BP (Business Process) systems are usually separate projects but these systems may have significant influence on each other. (ICT and BP systems of an organization may also be referred to as an *Organizational Information System* or OIS). Thus, the common analysis of these systems may have *significant benefits* – this is why there is an increasing need for the *common modeling and simulation* occurs. In the common analysis, we need models of ICT and BP systems that can *interact* with each other just as these systems interact with each other in the real world.

Depending on the task, *distinct* or *integrated* ICT and BP models may be used for the common analysis.

- Examples of tasks for which distinct models are appropriate: Support for the BSM (Business Service Management) method [12] in defining the performance and availability of ICT and the features of BPs operating the ICT systems (for example, in an ERP (Enterprise Resource Planning) system) according to the service-level requirements determined by the current and future BPs. Another example is to analyze the dynamic relationship between ICT and BP performance of system functions by using simulation in order to help BP and ICT designers and analysts [9].
- Examples for tasks with integrated ICT and BP model: It may be beneficial to integrate the model of the BP system into the model of the ICT system if, for example, the BP system is an intensive traffic source for the ICT system (for example, customer service offices in the ICT infrastructure). The BP model may integrate ICT model, for example, in the task of the optimization of the proportion of automatic (produced by some answering software) and operator performed activities of a help desk system.

In the common simulation analysis of ICT and connected

Manuscript received March 29, 2011.

László Muka is with the Elassys Consulting Limited, Budapest, Hungary (e-mail: muka.laszlo@elassys.hu).

Gábor Lencse is with the Department of Telecommunications, Széchenyi István University, Győr, Hungary (e-mail: lencse@sze.hu).

[1] An SSM-holon is a whole - with emergent properties - from some point of view of abstraction.

BP systems, we may easily be faced with the case of large and complex systems where the necessary computing capacity may reach or even exceed the reasonably available. The increase of the *efficiency of simulation* may be an answer to this problem.

The efficiency of simulation is influenced by many factors including *methodological fact*ors too (for example, the occurrence of *unstructured problems* and the problem of *efficient applicability of methods*).

The aim of this paper is to address the problem of the increase of the efficiency of simulation on *meta-level*, by developing *the principles of managing the efficiency of simulation* on the level of methods and problem contexts.

The *new results* in the paper can be summarized as follows. On the basis of 4-state and 2-state models of problem context types, the *features of the dynamic behavior of problem contexts* are investigated for the case of common modeling and simulation of organizational ICT/BP systems (OIS). Using the set of efficiency principles – referring to the re-definition of Checkland's systems performance criteria and to the criterion of gap-efficiency with an explanation of the process of the occurrence and elimination of the methodological gap – *efficiency management principles* are formulated serving for the managing of the efficiency of simulation of OIS on meta-level.

The new approach introduced in the paper has significant advantages comparing with other approaches. The classic simulation methodologies (for example, those described in [15]) are efficient only for the hard-systems type of problem contexts. Other context based systems approaches ([13], [14]) have static approach and do not take into account the occurrence of context-type changes in the execution of a simulation task. Furthermore, they do not use an explicit and general approach to efficiency such as the rules for the management of efficiency formulated in this paper.

The paper is organized in the following way. Section 2 describes the problem context state models and analyses the simulation of ICT and BP systems from the point of view of problem contexts. Section 3 introduces the efficiency criteria and defines the efficiency principles. Section 4 formulates the principles for the managing of efficiency of simulation on meta-level. Section 5 examines the work of the efficiency management principles. Section 6 refers to the current and potential applications. Section 7 summarizes the work.

II. The Problem Context State Models

In this section, the meta-level analysis of efficiency of simulation, the problem context state models[2] – the environment for the functioning of the simulation process – will be defined and explored, then the features of the process of simulation will be analyzed using the defined models.

A. The Jackson-Keys Classification of Problem Contexts

Jackson and Keys [4] defines the classes of problem contexts according to two dimensions: the *simple-complex* (or simple-systemic) dimension describes the *system feature* and the *unitary-pluralist*[3] dimension characterizes the *actors* (decision makers) of the problem context. According to this classification, the *types of problem contexts* may be: *simple-unitary*, *simple-pluralist*, *complex-unitary and complex-pluralist*.

B. The 4-state model of the dynamic problem contexts

The *4-state-type model of the problem contexts* shown in Figure 1 is created in the way of utilizing the *subset* of classes of Jackson and Keys in a different way, using them in a dynamic manner.

The extension-restriction relationship of the *generality* of the problem context types is demonstrated with dashed lines in Figure 1: the simple-unitary context type is a special case of the complex-unitary and of the simple-pluralist context types and all these three context types are the special cases of the complex-pluralist context type. The relationship in the level of determination of problem contexts is also shown: the lighter the shade of color a problem context in Figure 1 has the more determined the context is.

In Figure 1, straight lines with arrows on both ends show the transitions between different types of problem contexts which are reached by changes of dimensions (A-transitions), the straight lines with single arrow show the possibility of the occurrence of a new context (B-transitions), curved lines with arrows demonstrate the transitions between similar contexts (inside of a problem context type) (C-transitions).

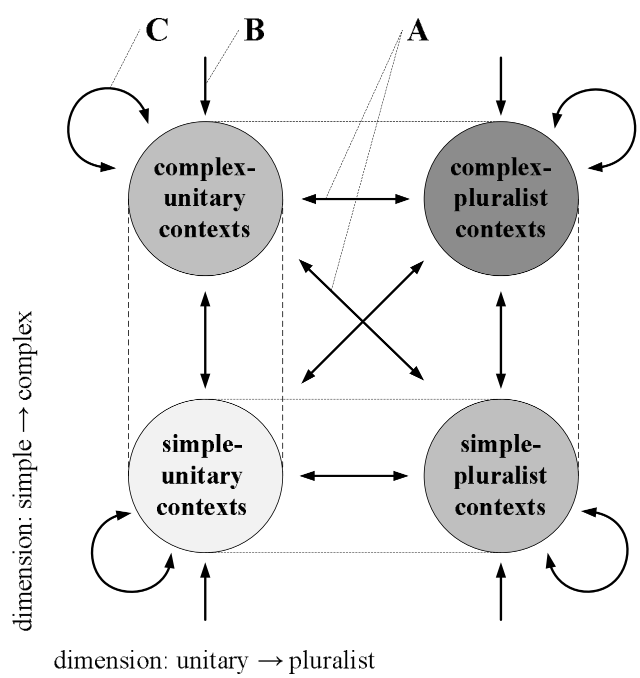

Fig. 1. Problem contexts and transitions

[2] The reader is referred to [17] too, regarding context state models in. The present paper analyses their operation in the way necessary for the formulation of the efficiency management principles.

[3] The "coercive" category of the actors has no significance in our analysis.

C. *The 4-state model of the dynamic simulation problem context*

The *dynamic simulation problem context (DSPC)* – which is the problem context environment of the functioning of the simulation process [7, 8] – *may be defined as the sequence of problem contexts and transitions of contexts* that occur in the process of simulation.

The following *propositions* about the features of DSPC are examined:

- The dynamic simulation problem context may contain all the problem context types.
- In the dynamic simulation problem context any type of problem context may occur in any phase of the simulation process.
- The occurrence of the problem contexts may happen independently from the process of simulation too.

The above propositions are examined for the case of the common modeling and simulation of ICT and BP systems.

Defining the simple-complex dimension

Vemuri (Vemuri, in [4]) defines the system feature of complex systems: *partially observable, subject to behavioral influences*[4]*, probabilistic and evolving*. According to Jackson and Keys, [4] complex systems – in addition to Vemuri's criteria – have a *large number of elements* (that are highly interrelated) and the *evolving* feature may be replaced by the features that complex systems are *open* and they *have purposeful parts (subsystems)* as well as the selection of *boundaries* of a system may have influence on its complexity.

Taking into account the previous points, the following features to characterize complex systems are defined: *(1) partial observability,* (*2) wide boundaries and high resolution*, both in *structures* and in *time* (which results in large number of elements, relations and events), *(3) openness* and *(4) purposeful subsystems (with behavioral attributes)*. (The *probabilistic feature* is also taken into account as it is explained later.) Thus, the *system features* of the *simple-complex dimension* of the *simulation problem contexts* for the *modeling and simulation of ICT and BP systems* are defined as follows:

1. Systems of interest are often only *partially observable*: this may be caused by data availability problems (for example: data are not collected or cannot be collected because of technical reasons, cost, time and resource limits; collected and available data are enough only for partial description of the system; data sources may be located in other systems and are not available for the modeling purposes, etc.).
2. The wide *boundaries* (including both structural and time limits) of the models of systems of interest and their high *resolution* too (including both structural and time boundaries and resolution) may make the problem complex: the wider the boundary is set the more complex the system may become and the same is true for the resolution.
3. The complexity is increased by taking into account the influences among systems (subsystems) – including, of course, the influences between ICT and related BP systems. Interacting systems are *open* to influences between each other. The more detailed the model of *interactions* is the more complex the system may become.
4. BP systems may have *active*, *purposeful parts*: their *behavior* cannot be predicted exactly (for example people in the system may act in opposition to simulation project goals).

A *simulation problem context* is *simple* if the systems of interest are observable, the boundaries and the resolution of modeling of the systems are set at a necessary but low level, the influences among the systems (subsystems) of interest are limited in the model (systems are reasonably closed) and the purposeful parts of processes are passive. Any of the above listed conditions may make the *simulation problem context complex*: if the systems of interest are not observable (partially observable), the boundaries and the resolution of modeling of the systems are set at a too wide/high level for simulation, the influences among the systems (subsystems) of interest are not limited enough in the model (systems are open) and the purposeful parts of processes are active.

Remarks:

- The *probabilistic* feature of the behavior of the analyzed systems is the basic object of the simulation investigation.
- The behavioral influences of systems of interests are taken into account in the examination of active, purposeful features of the BP systems.
- In determination of system features the *emergent properties*[5] has to be taken into account. (For example, on the one hand, the boundary for modeling should be set wide enough and the resolution of models high enough to examine the emergent properties and to get the necessary answer and on the other hand, the boundary should be narrow enough and the resolution low enough to be able to simulate the system.)

Defining the unitary-pluralist dimension

The decision makers of the simulation problem contexts in an organization environment are determined by the simulation project. The problem context is *unitary* if the *set of decision makers* have a *common set of goals* (agree) and *pluralist* (disagree) if they do not. *Problem solvers* (as participants of the problem context) may also become decision makers in the simulation process.

D. *Defining the 2-state model of DSPC*

In the following, according to the 4-state model of DSPC, the *2-state-type of model (or hard-soft model)* of DSPC will be defined (Figure 2).

[4] Political, cultural, ethical and other similar type of influences should be taken into account in the analysis of these systems.

[5] The *emergent property* may be, for example, an analysed functional capability of the system of interest. This capability may disappear or occur in correlation with the selected system features of a problem context.

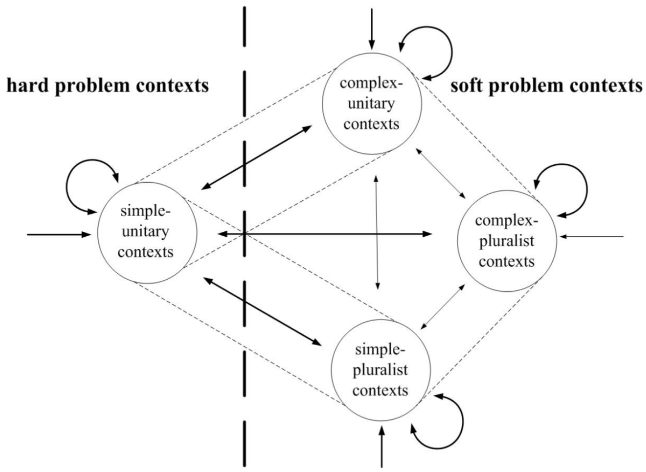

Fig. 2. The hard-soft model of problem contexts

Problem contexts could also be classified from the point of view of appropriate approaches. *Hard-systems approaches*[6] are suitable for looking for solutions to well-defined problem situations, starting from clearly defined objectives. Thus, *simple-unitary problem contexts* with well defined system features and with a common set of goals of decision makers are *hard problem contexts*. *Soft-systems approaches* are to cope with ill-defined, unstructured problem situations, in which objectives are themselves problematical. The *complex-pluralist problem contexts* with undefined system feature and with pluralist set of decision makers are *soft problem contexts*. The *complex-unitary* problem context may have an active purposeful part and thus, it will also require a soft-systems method to deal with the situation therefore this problem context can be classified as a *soft problem context*. The *simple-pluralist* problem context requires a soft approach to deal with the pluralist set of decision makers that is it is a *soft problem context* too.

(Remark: Soft-systems approaches may be appropriate both hard- and soft problem contexts but hard-systems approaches are suitable only for hard problem contexts.)

In the following, the 2-state and the 4-state models of DSPC are applied in the argumentations in a mixed way.

E. Analyzing the DSPC transitions

Now, in order to reveal the features of DSPC, the transitions of types of problem contexts will be examined. (The transitions are investigated as they are shown in Figure 1.)

A-transitions:

A change of the simulation process phase may generate transition: for example, in the simulation process, after the phase of the analysis of results a need occurs to change the resolution of the simulation model. (The simulation process phases are described, for example, in [8].) The system feature of the problem context may remain simple or change for complex and there can be agreement or disagreement about the measure of the resolution: these are transitions from simple-unitary to complex-unitary, simple-pluralist and complex-pluralist contexts. When entering a new phase, a new problem situation is created thus *any type* of problem contexts may be identified. In general, *any phase* of the simulation process may lead to a *pluralist* problem context: different opinions may occur concerning the goal setting for the phase and concerning the further use of the results of the phase.

Transition may also be generated by *transformation decision*: for example after entering a simulation process phase it is found that the problem context is one of the complex-unitary, simple-pluralist or complex-pluralist contexts. There should be made transformation decision for the transition into the simple-unitary context, because the simulation methodology is appropriate only for the simple-unitary context [3, 4]. This may be done by the way of agreed changes of system features (if it is necessary) and by the way of finding the consensus about the set of goals. The system features for modeling may be changed into the *simple* direction by decisions (and actions taken according to the decisions) about data availability (1), by decisions about setting up the boundaries and the resolution of systems of interest (2) and decisions about the modeling of interactions among systems (3). The passivity of the purposeful part of the system (4) may be reached typically by using some consensus building method.

In the decision process, transitions between complex-unitary, simple-pluralist and complex-pluralist contexts may also occur: for example, the purposeful part of the system feature may change between active and passive (all other system features show a simple system) and the set of the opinion of decision makers may change between agree and disagree.

B-transitions:

These transitions are of “*insertion*” type: a new problem context may be generated *independently* from the earlier problem contexts. For example, because of the influence of the changes in the organization (in the wider environment of the simulation process) new requirements may occur concerning the system feature of the problem context and a new set of the decision makers opinion may occur too. The starting problem contexts of the simulation process may also be of *any type* and the initial problem analysis (*structuring)* may lead to a pluralist set of opinions about the goals even if there was an agreement about the initiation of the simulation project.

C-transitions:

These transitions show the change of the problem context features without changing the problem context type. If the system features show a complex (simple) system it remains complex (simple) after this type of transition only with other set of problem context features and if the set of decision makers is unitary it remains unitary or if the set of decision makers is pluralist it remains pluralist only with other set of disagreeing decision makers. These transitions may occur, for example, in the decisions process but this type of transitions

[6] A more detailed description of hard- and soft-systems approaches can be read, for example, in [1].

may take place between simple-unitary contexts in the execution of the phases of simulation too.

Remarks to problem context transitions:

- The unitary-pluralist dimension of the problem contexts may also be changed if there is a *change in the set of actors* (for example, there are two collaborating teams of actors in the process of simulation).
- The interaction of actors with purposeful parts of the systems of interest may also change the unitary-pluralist dimension of a problem context.
- It is important to notice, that the *worldview* (Weltanschauung) of actors may also influence the simulation problem context through the decisions made about the features of the simple-complex dimension [4].
- A pluralist set of opinions may also occur about the different ways of *implementation* of results (for example: who is responsible for what during the implementation).

III. Introducing Efficiency Criteria

In the following, the rather static approach of the Jackson-Keys method [4] will be *extended*: for the DSPC, efficiency criteria are introduced and developed and the efficiency principles are defined.

A. Efficiency in the Jackson-Keys method

According to Jackson and Keys, a method is appropriate for a problem context if it is selected to be the same type as the explored type of the problem context. Methods, which are suitable for complex-pluralist problem contexts, are potentially able to address problems in all other problem contexts but using methods for complex-pluralist problem contexts in other problem contexts may lead to *inefficiency*.

B. Checkland's systems performance criteria

If the system approach of simulation is used – the holon and HAS concepts of the simulation process –, it seems to be fruitful to apply the approach to *efficiency of activity systems* [2]. According to Checkland, the problem of *efficiency* is addressed together with the examination of questions of *efficacy* and *effectiveness*. According to Checkland, there is a *hierarchy-like relationship between* the three criteria efficiency, efficacy and effectiveness: the question of the adequacy for the longer term and for the wider environment is checked by the *effectiveness (criterion E3)*, the *efficacy (criterion E2)* investigates the question whether the solution will be suitable and work in all circumstances and the *efficiency (criterion E1)* examines the traditional question of efficiency (the question of direct efficiency) which can be measured by the proportion of the required outputs and the resources used to produce the outputs.

C. Defining the efficiency principles

Now, the *principles of efficiency* will be formulated applying Checkland's *criteria* for the process of simulation and introducing a new criterion, the criterion of *gap-efficiency*. Efficiency principles are defined for the relationship of fitting of methodology, method and problem context.

The *extension-restriction relationship* of the scopes of method types is the same as the extension-restriction relationship of the *generality* of the problem context types.

(E1) The *principle of methodological efficiency* can be defined as follows: for a methodology to be efficient the best fit with a specific problem context should be found. It means that the type of the selected method should be the same as the type of the problem context. Furthermore, it also means that the best fit with a specific problem context should be found within the set of methods of the same type of the methodology – if there are more methods of the same type in the methodology.

(E2) The *principle of hardening up and softening up (or the principle of methodological efficacy)* is the principle of dealing with a problem context that does not fit into one type of problem contexts in the sense that it has some aspects belonging to other problem context type. In other words, an aspect of a problem context, which has been revealed by applying the selected (according to the condition E1) method of the methodology, defines a problem context with a type different from the original one. In this case, in order to find the exact fit and to avoid inefficiency, the methodology should be *hardened up* or *softened up* by involving a method which is efficient for that different-type problem context.

(Eg) The *principle of the elimination of the methodological gap (or the principle of gap-efficiency*) is the principle for dealing with the problem of inefficiency that may be caused by soft-hard *problem context transitions*. These transitions are necessary and crucial because the traditional *simulation methodology,* as it was mentioned before, *is a method appropriate only for simple-unitary context* [3, 4]. A *methodological gap* ([7, 8]) may occur in the execution of the process of simulation if a soft-systems method and a hard-systems method of the methodology is applied for two sequencing problem contexts: the set of hard-level information for further processing by some hard-systems method is produced from the set of soft-level information by the way of using some soft-systems method and executing *ad-hoc, occasional condensing*[7] (Figure 3). For example, in the process of simulation, executed according to the framework for collaborative modeling and simulation [11], a methodological gap may occur when the team of hard modelers builds the simulation model using modeling data got from the team of soft modelers. The methodological gap may lead to *inefficiency* because of the fact that not the necessary condensing has been carried out which results in that not the required simulation model will be built.

The methodological gap may be *eliminated* by a *methodology constructed to connect the soft and hard levels* [6-8]. In order to tell whether a methodological gap has occurred or not, a new criterion the *criterion of gap-efficiency (Eg)* is introduced. The principle based on this criterion is *the*

[7] Checkland defined occasional condensing as the relationship between the soft-systems thinking and the hard systems thinking [1].

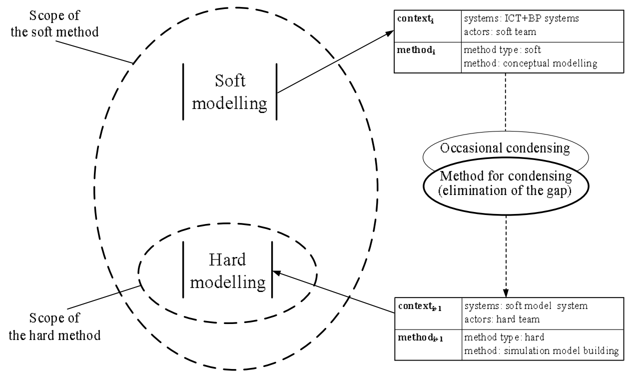

Fig. 3. The occurrence and elimination of a methodological gap

principle of elimination of the methodological gap (or principle of gap-efficiency).

(E3) The *principle of methodological effectiveness* expresses the efficiency requirement for the whole process of simulation resulting in the reduction of the number of problem contexts to deal with, reduction of the number of methodological cycles (number of iterations) in the process.

E1, *E2* and *Eg* refer to the *step-by-step efficiency* (efficiency in problem context states and transitions) of application of methods while *E3* stands for *the efficiency on long-range* (to have less states and transitions in the process of application of the methodology).

IV. Formulating Efficiency Management Principles

Now, the efficiency management principles below will be formulated taking into account the requirements set by to the DSPC and by the efficiency principles described above.

Points a – c are the principles for collecting an efficient set of methods. For finding the "best fit", only the methods of the set of methods can be taken into account. For the methodology to be efficacious, the set of problem contexts should contain an efficacious method for any problem context of the DSPC. *Points d – i* refer to the structural and application (operation) type efficiency management principles.

a As a *soft-systems method*, we need a method appropriate for the complex-pluralist problem context. In theory, it may be used to any problem context but the *function of application* of the soft-systems method *is different in* the different phases of the simulation process (for example: scanning the relevant set of systems, scanning for simulation scenarios, etc.).

b The set of *hard-systems methods* should contain the methods of the traditional *simulation methodology* – according to the requirement of the methodological efficiency it should be a set of methods for the typical hard problem contexts in the simulation process that has been identified – and further methods required by the principle of the methodological efficacy and effectiveness (for example methods supporting goal setting, or methods supporting fast modeling).

c For the *elimination of the methodological gaps*, the set of methods is proposed to contain a *methodology connecting the soft-systems and hard-systems levels*. This is a methodology consisting of a soft-systems method and hard-systems methods defined on the basis of the identified soft- and hard-systems level contexts and the constraints for condensing (hard contexts may be for example contexts relating to the tasks of the analysis of time relations in ICT and BP systems).

d It should be taken into account that the traditional simulation methodology – which is based on a set of hard-systems methods – is a hard-systems approach. Hard-systems methods are appropriate only for hard problem contexts, therefore soft problem contexts of DSPC should be transformed into hard problem contexts.

e For managing efficiency, it is proposed to help to realize the change of problem contexts: hard-systems methods of the set of methods cannot see beyond the hard problem context, thus the need for insertion for a new problem context (generated by the observed systems) can be realized only by a soft-systems method.

f The hardening up and softening up of the methodology is proposed to be supported in any phase of the simulation process: inefficacy (and inefficiency) may occur in any phase of the simulation process when applying a method to a problem context which does not fit exactly into one problem context type. For managing similar situations, it is proposed to have soft-hard (hard-soft) method pairs for such problem contexts.

g It is proposed to support the method-selection decisions inside a context type: according to the principle of the methodological efficiency, it is necessary to find the best fit of a problem context and the method inside the given type too.

h To be efficient, the whole set of methods should be taken into account in the method-selection decisions: to reach the best fit of a problem context and a method, it may be necessary to choose a method other than the next method in the process of simulation.

i It is proposed to support to take into account – with an appropriate use of the necessary methods – the principle of the methodological effectiveness. It is necessary to find the best fit for a method and for the whole sequence of methods taking into account a wider systems environment and longer time frame generated by the observed systems and by the process of simulation itself.

V. Efficiency Management Example

Let us examine the work of the efficiency management principles on the example of a simulation task execution.

Let us use the *4-state model* of problem contexts and let *X* denote the set of problem contexts that have been identified in the process of simulation. In the 4-state model, the *types of problem contexts* are *simple-unitary (su), simple-pluralist (sp), complex-unitary (cu) and complex-pluralist (cp)*. Let *M* denote the set of methods which contains the methods of a *classic simulation methodology [8] (SM1(Goal-definition), SM2(Data-gathering), SM3(Modelling), SM4(Simulation), SM5(Evaluation), SM6(Implementation-support)), the SSM (Soft Systems Methodology [2]), and the MCMM (Modified Conceptual Modelling Methodology [8]).* Thus, the method set is $M = \{SM1; SM2; SM3; SM4; SM5; SM6; SSM; MCMM\} = \{1; \ldots 6; 7; 8\}$.

The efficiency management principles listed in the previous section will be referred as $p_a, p_b, \ldots, p_i$.

The transition between two subsequent problem context states of the process of the simulation task execution may be described by the expression:

$$x_{k,(type_x)}\left(m_{(type_m),(method)}\right) \xrightarrow[\text{}]{\overset{(insertion)}{(transition\ type)}} x_{k+1,(type_x)}\left(m_{(type_m),method}\right),$$

where $x_{k,(type_x)}$ is k-th probem context in the problem context sequence ($k = 1,2,3, \ldots, |X|$, $\boldsymbol{type_x} = \{\boldsymbol{su}; \boldsymbol{cu}; \boldsymbol{sp}; \boldsymbol{cp}\} = \{\mathbf{1}; \mathbf{2}; \mathbf{3}; \mathbf{4}\}$) and $\boldsymbol{m_{(type_m),(method)}}$ is the method assigned to the problem context ($\boldsymbol{type_m} = \{\boldsymbol{su}(\boldsymbol{SM1}, \ldots, \boldsymbol{SM6}, \boldsymbol{MCMM}); \boldsymbol{cu}; \boldsymbol{sp}; \boldsymbol{cp}(\boldsymbol{SSM}, \boldsymbol{MCMM}\} = \{\mathbf{1}; \mathbf{2}; \mathbf{3}; \mathbf{4}\}$, $\boldsymbol{method = method\ identifier\ in\ the\ set\ M} = \{\mathbf{1}; \mathbf{2}; \ldots \mathbf{6}; \mathbf{7}; \mathbf{8}\}$) and the variables of the transition operator are**:** $\boldsymbol{transition\ type} = \{\boldsymbol{A}; \boldsymbol{C}\}\ \boldsymbol{and\ insertion} = \{\boldsymbol{B}\}$.

Now, let us examine the problem context sequence

$$x_{1,4} \xrightarrow{A} x_{2,1} \xrightarrow{C} x_{3,1} \xrightarrow{A} x_{4,4} \xrightarrow{A} x_{5,1} \xrightarrow{C} x_{6,1} \xrightarrow{C} x_{7,1} \overset{B}{\xrightarrow{A}}$$

$$\overset{B}{\xrightarrow{A}} x_{8,4} \xrightarrow{A} x_{9,1} \xrightarrow{C} x_{10,1} \overset{B}{\xrightarrow{A}} x_{11,4} \xrightarrow{C} x_{12,4} \xrightarrow{A} x_{13,1} \xrightarrow{C}$$

$$\xrightarrow{C} x_{14,1} \xrightarrow{C} x_{15,1} \xrightarrow{C} x_{16,1} \xrightarrow{()} \ldots \xrightarrow{()} x_{|X|-1,1} \xrightarrow{A} x_{|X|,4}$$

which shows a part of the process of a simulation task execution. (Let us denote this example sequence by DSPC(E)).

Showing also the methods assigned to the contexts, the DSPC(E) has the form:

$$x_{1,4}(m_{4,7}) \xrightarrow{A} x_{2,1}(m_{1,1}) \xrightarrow{C} x_{3,1}(m_{1,2}) \xrightarrow{A} x_{4,3}(m_{4,7}) \xrightarrow{A}$$

$$\xrightarrow{A} x_{5,1}(m_{1,2}) \xrightarrow{C} x_{6,1}(m_{1,3}) \xrightarrow{C} \xrightarrow{C} x_{7,1}(m_{1,4}) \overset{B}{\xrightarrow{A}} x_{8,4}\ (m_{4,7}) \xrightarrow{A}$$

$$\xrightarrow{A} x_{9,1}\ (m_{1,2}) \xrightarrow{C} x_{10,1}(m_{1,3}) \overset{B}{\xrightarrow{A}} x_{11,4}(m_{4,7}) \xrightarrow{\mathrm{C}} \mathrm{x}_{12,4}(\mathrm{m}_{4,8}) \xrightarrow{\mathrm{A}}$$

$$\xrightarrow{A} x_{13,1}(m_{4,8}) \xrightarrow{A} x_{14,1}(m_{1,4}) \xrightarrow{C} x_{15,1}(m_{1,5}) \xrightarrow{C} x_{16,1}(m_{1,6}) \xrightarrow{()} \ldots$$

$$\xrightarrow{()} x_{|X|-1,1}\ (m_{1,5}) \xrightarrow{A} x_{|X|,4}(m_{4,7}).$$

The goal is to introduce the work of the approach through the examination of *typical sequence patterns* of DSPC(E).

The sequence fragment $x_{1,4} \xrightarrow{A} x_{2,1}$ shows the transformation of the starting *cp* problem contest to a *su* problem context (p_d) which is a problem structuring pattern. Using the *su*-type method $(m_{1,1})$ for the transformation would be inefficacious and thus inefficient.

The sequence fragment $x_{3,1} \xrightarrow{A} x_{4,3} \xrightarrow{A} x_{5,1}$ is a softening up pattern: for $x_{3,1}$, it is necessary to involve a soft method (p_f). There is no M_3-type method in the set of methods (lack of soft method), thus SSM used $x_{4,3}$ which is efficacious for the case. (Using the *su*-type method $(m_{1,2})$ for $x_{4,3}$ would be inefficacious.)

In the DSPC(E), the $\xrightarrow{C} x_{7,1} \overset{B}{\xrightarrow{A}} x_{8,4}$ and the $x_{10,1} \overset{B}{\xrightarrow{A}} x_{11,4}$ *insertion* transitions occur. Without taking into account *insertions* (p_e does not function in the methodology), the examined steps of DSPC(E) may have, for example, the following forms

$x_{7,1}\ (m_{1,4}) \overset{B}{\xrightarrow{C}} x_{8,1}(m_{1,5})$ and $x_{10,1}(m_{1,3}) \overset{B}{\xrightarrow{C}} x_{11,4}(m_{1,4})$.

The execution of the next phase of the simulation (the use of methods $m_{1,5}$ and $m_{1,4}$ for the problem context $x_{8,4}$ and $x_{11,4}$) is inefficacious, or from other point of view, the processing of the contexts $x_{8,1}$ and $x_{11,4}$ are not the contexts to process.

The $x_{12,4} \xrightarrow{A} x_{13,1} \xrightarrow{A} x_{14,1}$ sequence of DSPC(E) shows the pattern of elimination of the methodological gap (p_c). In the examined sequence, the use $m_{4,8}$ is used for the sequence $x_{12,4} \xrightarrow{A} x_{13,1}$ before the use of $m_{1,4}$ for $x_{14,1}$ (before the simulation phase). The use of $m_{1,4}$ would be inefficacious for $x_{12,4} \xrightarrow{A} x_{13,1}$ and $m_{4,7}$ would also be inefficient for the context $x_{13,1}$.

The closing sequence fragment of DSPC(E) $x_{|X|-1,1} \xrightarrow{A} x_{|X|,4}(m_{4,7})$ is the reverse case of the starting sequence $x_{1,4} \xrightarrow{A} x_{2,1}$: the use of $m_{1,5}$ instead of $m_{4,7}$ would be inefficacious for the $x_{|X|,4}$ context.

The set of methods may be described as $M = M_{su} \cup M_{cu} \cup M_{sp} \cup M_{cp}$ and in the examined case, $M_{sp} = \emptyset$ and $M_{cu} = \emptyset$. To improve efficiency, for example, the user methods may be used in the design of set of methods (p_a, p_b).

Remark: The example does not give an exhaustive analysis: there are no example patterns for the principles p_g, p_h and p_i.

VI. Applicability of the Results

The results of the paper have already been successfully used in several large projects such as, for example: improving the performance of BP of a large telecommunication service company by integration of ICT services [16], modelling and simulation of a large CRM system (together with the problem context retrieval approach [17]), simulation project of BCP-DRP (Business Continuity Plan – Disaster Recovery Plan) in order to support planning at a large power service company.

The approach proposed by the paper is general: it is generally applicable for the meta-modelling of any ICT/BP because no domain-specific restrictions were used. Results may also be applied for other systems with components *technical-subsystem/human-subsystem*: for example, *traffic-subsystem/service-process-subsystem*, *environment-protection-subsystem/human-supervision-subsystem*. For these cases, the system features of the 4-state and 2-state models and the question of the necessary methods should be revised.

VII. Conclusion

In this paper, a new method for the management and increase of the efficiency of modelling and simulation of Organisational Information Systems has been formulated.

Using the 4-state and the 2-state models of problem contexts, the features of the dynamic simulation problem context for the case of modelling and simulation of organisational ICT and BP systems were investigated.

The method of Jackson and Keys (the approach for appropriate fitting of problem contexts and methods) was extended: Taking into account the features of the dynamic simulation problem context (the 2-state and 4-state models of problem context types) and the efficiency principles – the Checkland's systems performance criteria *re-defined* simulation efficiency measures and the criterion of *gap-efficiency* with the concept of explaining the occurrence and elimination of the methodological gap in the process of simulation –, the *principles of the managing the efficiency of simulation* were formulated. These principles of the problem context state approach are able to deal with both aspects of the efficiency of simulation: with the step-by-step efficiency of fitting of states and state-transitions with methods and with the long-range requirement of efficiency according to the amount of states and transitions.

The newly formulated set of principles *of the managing the efficiency of simulation* may also be taken as general requirements (including requirements for the set of methods, for the structure and operations) for building and implementation of a simulation meta-methodology.

The work of the new approach is illustrated by the analysis of an example of a simulation task. The applicability of the approach is shortly overviewed for the common analysis of systems with different cooperating subsystem components.

References

[1] P. Checkland, "From Optimizing to Learning: A Development of Systems Thinking", *J. Opl. Res. Soc.* Vol. 36, No. 9, 1985. pp. 757-767.

[2] P. Checkland, "Soft Systems Methodology", in *Rational Analysis for a Problematic World*, Edited by J. Rosenhead, John Wiley & Sons, 1989.

[3] M. C. Jackson, *Systems Methodology for the Management Sciences*, Plenum Press, New York, 1991.

[4] M. C. Jackson, P. Keys, "Towards a System of Systems Methodologies", *J. Opl. Res. Soc.* Vol. 35, No. 6, 1984.

[5] R. Jain, *The Art of Computer Systems Performance Analysis*, John Wiley & Sons, New York, 1991.

[6] L. Muka, G. Lencse, "Developing a Meta-Methodology Supporting the Application of Parallel Simulation", in *Proc. 2006 European Simulation and Modelling Conference*, Toulouse, 2006, pp. 117-121.

[7] L. Muka, G. Lencse, "Hard and Soft Approaches in a Simulation Meta-methodology", in. *Proc. 5th Industrial Simulation Conference,* Delft, 2007, pp. 17-22, 2007.

[8] L. Muka, G. Lencse, "Developing a meta-methodology for efficient simulation of infocommunication systems and related processes", *Infocommunications Journal*, Vol. LXIII, No. 7, pp. 9-14, 2008.

[9] P. Ray, A. Serrano, "Collaborative Systems and Business Process Design Using Simulation", *Proc. 37th Hawaii International Conference on Systems Sciences*, 2004.

[10] M. Sierhuis, W.J. Clancey, "Modeling and Simulating Work Practice: A Method for Work System Design", *IEEE Intelligent Systems*, September/October, Vol. 17, No. 5, pp. 32-41. 2002.

[11] M. Sierhuis, A. M. Selvin, "Towards a Framework for Collaborative Modeling and Simulation", *Workshop on Strategies for Collaborative Modeling & Simulation*, CSCW '96, Boston, MA

[12] M. Warmerdam, P. Bredveld, "A Holistic Approach to Delivering the Value of IT: Business Service Management", *IDC White Paper*, Framingham USA, www.idc.com, 2003.

[13] T. Nakamura, K. Kijima, *System of system failures: Meta methodology for IT engineering safety*, John Wiley & Sons, 2008.

[14] M. C. Jackson, *System Thinking - Creative Holism for Managers*, John Wiley & Sons, 2003.

[15] R. J. Paul, V. Hlupic, G. Giaglis, "Simulation Modelling of Business Processes", Accepted for UKAI'98 – *UK Academy of Information Systems Conference*, Lincoln, UK, 1998.

[16] L. Muka, G. Muka, "BPR projekt támogatása komplex szinulációs modellel", *Minőség és Megbízhatóság,* 2. szám, 88-93 oldal, 2009.

[17] L. Muka, B. K. Benko, "Meta-level performance management of simulation: The problem context retrieval approach" *Periodica Polytechnica,* to be published.

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 level doctor in architectures of CAD systems in 1987. Dr Muka finished an MBA at Brunel University of London in 1996. Since 1996 he has been working in the area of simulation modeling of telecommunication systems, including human subsystems. He is a regular invited lecturer in the topics of application of computer simulation for performance analysis of telecommunication systems at the Multidisciplinary Doctoral School of Engineering, Modelling and Development of Infrastructural Systems at the Széchenyi István University of Győr.

Gábor Lencse received his M.Sc. in electrical engineering and computer systems at the Technical University of Budapest in 1994 and his Ph.D. in 2001. The area of his research is (parallel) discrete-event simulation methodology. He is interested in the acceleration of the simulation of info-communication systems. Since 1997, he has been working for the Széchenyi István University in Győr. He teaches computer networks and networking protocols. Now, he is an Associate Professor. He is a founding member of the Multidisciplinary Doctoral School of Engineering, Modelling and Development of Infrastructural Systems at the Széchenyi István University. He does R&D in the field of the simulation of communication systems for the Elassys Consulting Ltd. since 1998. Dr Lencse has been working part time at the Budapest University of Technology and Economics (the former Technical University of Budapest) since 2005. There he teaches computer architectures.