Terminology of Nested Simulation Models

Ivan Krivy and Eugene Kindler

Abstract: *The aim of this paper is to propose a terminology of nested simulation models. The terminology is based on our recently recommended classification of nested simulation, i.e. of simulation of systems containing simulating elements, with respect to three basic criteria (reflectivity, size and depth of nesting). The criteria are explained and supported by giving references to existing implementations of simulation models. The criteria can be combined and confused and, therefore, the real terminology must be limited to the cases that seem to be realistic in the near future.*

Key words: *Simulation, Terminology, Nested models, Reflective models, Anticipatory simulation, Object-oriented programming.*

INTRODUCTION

Let us start from the explanation of the term *nested simulation*. Nested simulation is considered to be a simulation of systems containing some simulating elements e.g. computers. We proposed to use the term *external models* or *nesting models* for the models of such systems and the term *internal models* or *nested models* for the models carried by the simulating elements. The importance of nested simulation is continue to grow, namely because of the following two reasons:

1. The modern man-made systems like transport, production and other ones apply computers for so called anticipatory simulation, i.e. simulation existing during real time existence of the system and making possible to inform users and/or automatic control units on the future behavior of the system. As it was shown by Kindler [6], during the design of such a system it is desirable to include the anticipatory simulation into the simulation models that exist before the simulated system is physically realized. This idea was applied for example to transitic systems [1,2] and patient-in bed systems in hospitals [7-9].

2. The quality of the simulation programming tools is so high that one can apply them to a so-called fictitious simulation or pseudosimulation [6], i.e. for implementing certain non-simulation-oriented routines by means of simulation models of fictitious system. When such an implementation is used in a man-made system, the simulation model of the fictitious system is nested into the simulation model of the man-made system. Regarding the examples of fictitious simulation, we can mention the Dijkstra-Dahl's method of computing the shortest path [3]. This method was successfully applied to simulation models of container terminals (see e.g. [1,2,5]).

CRITERIA FOR CLASSIFICATION OF NESTED SIMULATION MODELS

First of all we introduce the following terminological scheme. Let *M* be an external simulation model of a system *S* containing a computer *C* that simulates. It means that *M* carries an internal model *m* of a certain system *s*.

For the classification of nested simulation models three basic criteria are proposed:

- ***size*** of nesting,
- ***depth*** of nesting,
- relation of ***reflectivity***.

We can imagine that *M* contains more than one computer and all the computers simulate. It means that more than one internal model is nested in the external one. The number of nested internal models is said to be the *size of nesting*. A model without nesting (i.e. conventional simulation model) can be considered to be a model with zero size of nesting. If *M* contains just one model nested in it, the size of nesting is equal to one. An example of a model with the nesting size two is given in the Fig. 1 using so called the Mejtsky's diagram (see [10,11]).

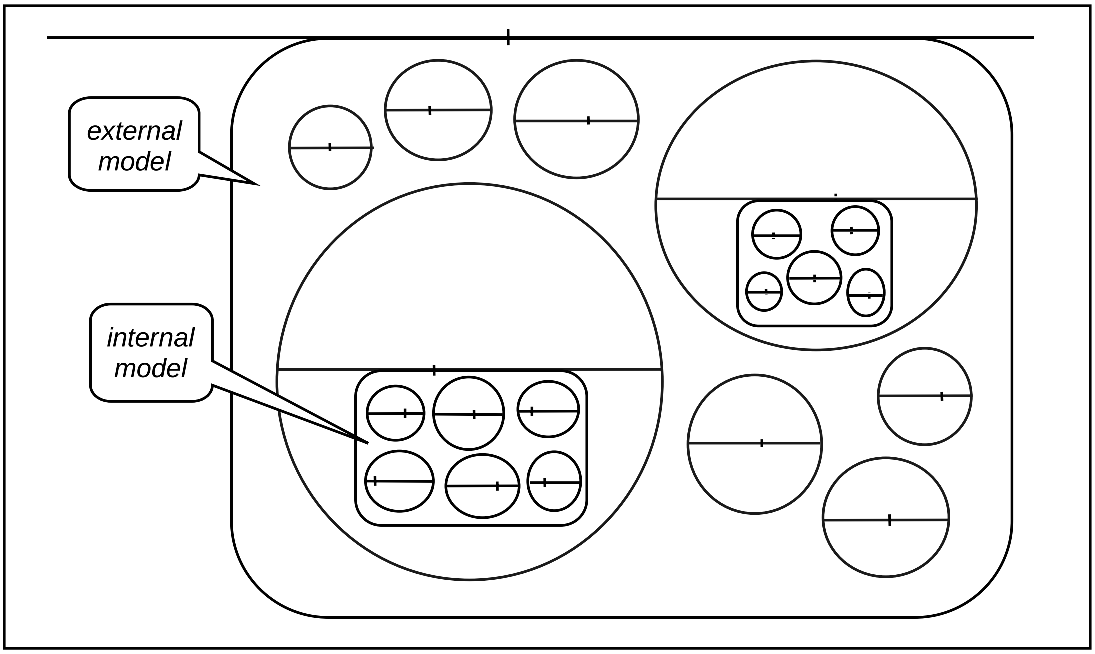

Figure 1: Nested simulation: depth 1, size 2

Not only the external system *S* can contain a simulating computer but also the internal system *s* can have a computer that simulates a system σ, i.e. that carries its

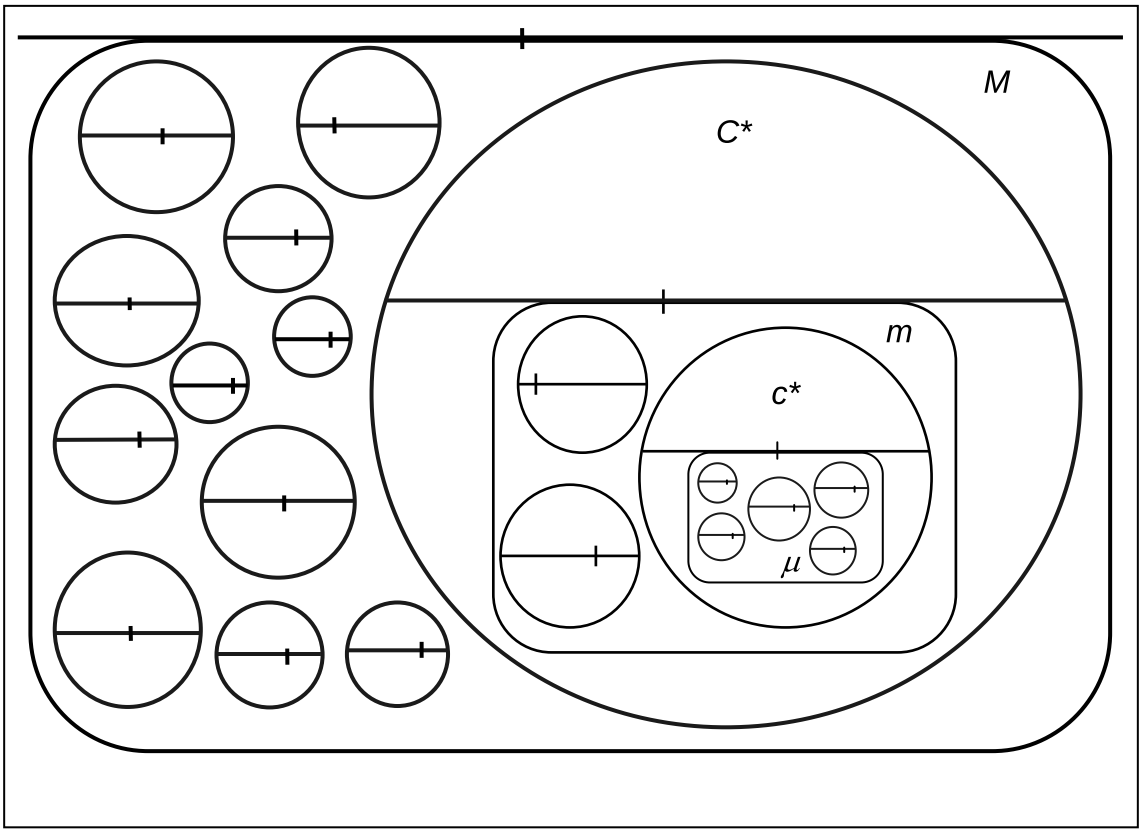

Figure 2: Nested simulation: depth 2, size 1

model μ. Therefore, we can introduce the term *depth* (*level*) *of nesting*. While a conventional simulation without nesting can be considered to be of the depth zero and the simple nesting displayed in Fig. 1 is of the depth one, the example just mentioned is of the depth one relating to *m*. This situation is illustrated in Fig. 2, the images of computers carrying models *m* and μ being denoted by *C** and *c*,* respectively. The natural demand is that since the nesting of *m* in *M* is of depth 1 and so that of μ in *m*, the nesting of μ in *M* should be viewed as that of depth 2. It is evident that such a nesting can continue in order to get the depth of nesting greater than two. And it is also evident that one can introduce the notion of "relative nesting", i.e., in the last example, nesting of *m* ***relating to*** *M* is of depth 1 and so nesting of μ ***relating to*** *m*. The generally sounding statement of nesting without the complement "relating to", e.g. that in Fig. 2 one meets nesting of depth 2, concerns generally two models *m* and μ so that μ is nested relating to *m* and *m* is nested relating to *M*, while no model is nested relating to μ.

Let *m* be of the depth one with respect to *M*. If both *M* and *m* concern the same system (model the same object in very similar manner), we can say that the nesting is *reflective* or that *m* is *reflectively nested* in *M.* (It is evident that in case *M* and *m* can be similar one to another but they can never be equivalent – see further). Otherwise, the nesting is said to be *non-reflecting*. Such a case is met for example, when *m* is a model of a fictitious system.

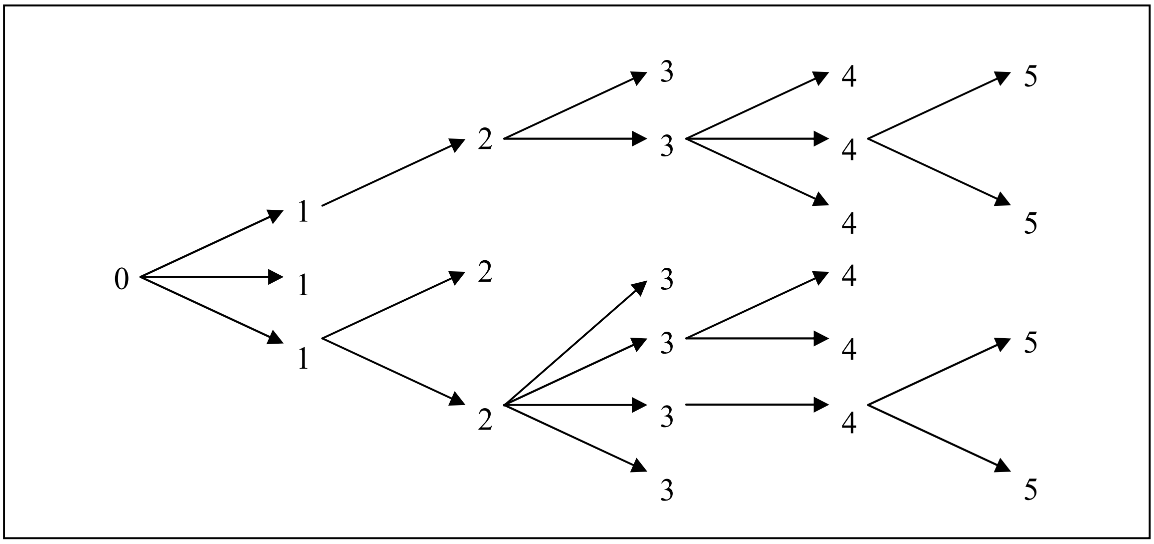

Figure 3: Nesting tree

DISSCUSSION ON THE CRITERIA

In general, the size of nesting is a dynamic value: a simulated system can contain several computers and the function of them is often not limited to simulation. So in the model of the system the number of the nested models varies from zero to a certain maximum value. Naturally the most important and frequent case is that the maximum value is equal to one. However, there are already implemented models that show the depth equal two or more (see e.g. [4]).

Therefore, it seems that beside the (dynamic) size concept it is suitable to introduce ***static*** *size criterion that could be declared as the maximum value of size.* In case the situation is clear the adjective static could be omitted.

Suppose an example illustrating a general situation, i.e. the situation of a model of the size greater than one and depth as well. Such a model should be characterized by a dynamic tree of depths and sizes like that in Fig. 3, where the natural numbers represent the depths of models to the external one.

Therefore it would be suitable to define also *the* ***static*** *depth of the model as the maximum depth occurring in it*. For the model presented in Fig. 3, the depth would be 5.

Nevertheless, the example turns our attention again to the static size of nesting. Should it be three – according to the maximum size (in time) of the models nested directly in the external model – or four (according to the maximum size of nesting in one of the models at level 2), or 22, according to the maximum number of nesting models that can contemporarily occur? A good answer, which might be valid for a long time, is difficult; maybe 4.

The concept of reflectivity is not exact. There are some cases of nesting that are evidently non-reflective (e.g. if the nested model concerns a fictitious system), but the question where are limits of reflectivity has no exact answer. The nested models must always differ from the model in which it is nested – otherwise it should contain also a

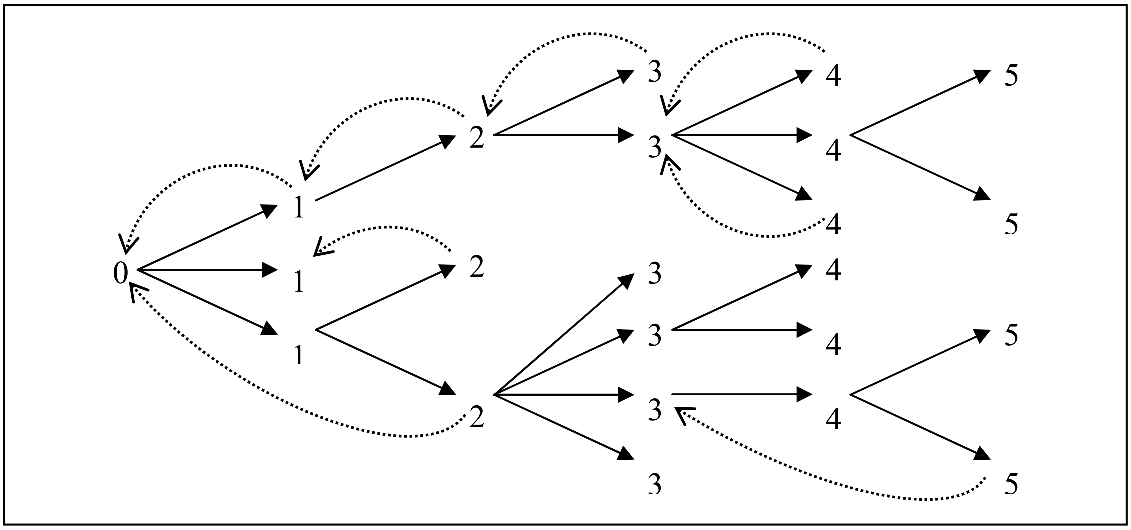

Fig. 4: Nesting tree with reflectivity

nested model that should also contain a nested model etc. until infinity.

When the reflectivity criterion is combined with the other two criteria, other problems arise, namely that a nested model m_1 can be reflectively nested to a model M_1, while another nested model m_2 can not be reflectively nested to M_1; possibly m_2 can be reflectively nested to another model M_2. If we would like to respect the reflectivity exactly, we should introduce the reflective nesting into the tree introduced above. Naturally, such a structure would not be tree more. For example, the tree presented in Fig. 3 would be completed by other – dotted – arrows that represent the reflectivity relation (see Fig. 4).

Suppose the case with the depth equal to one and the size greater than one. If we have only one class *C* of simulating elements (computers) carrying internal models, we get *homogeneous models* (*simulation*) because all the internal models belong to the same class *C*. In case the nesting is reflective, we can speak on ***homogeneous reflective*** *models* (*simulation*), otherwise the models (simulation) can be considered to be ***homogeneous non-reflective****.* Let simulation that is not homogeneous be called ***non-reflective***.

A question arises, whether a non-reflective simulation of non-zero depth has a certain importance e.g. in practical applications. An idea arises that in case we could prove that such cases are behind a horizon of a rational interest, we could omit them and our analysis could be limited to the homogeneous cases. The answer speaks against such an idea. There are "commercial" simulation models that are not homogeneous. One of them was applied in the simulation of container terminals at sea harbors [5]. In the case there are two classes *C1* and *C2* of simulating elements: *C1* makes a simulation of a fictitious system, which allows to compute the optimum trace of a ground-moving transport tool in the labyrinth of free places among the containers, and *C2* tests whether the computed trace does not cause a conflict between the given transport tool and another element of the dynamically changing structure of the container yard. While *C2* leads to reflective nesting *C1* leads to the non-reflective nesting and the alternating of the activities made by those classes must be classified as *non-homogeneous nesting* (*simulation*, *models*).

Suppose a nesting simulation of depth two and size two so that one of the nested model is reflective with respect to the external model, while the other nested model is evidently non-reflective because it concerns a fictitious system. After an extended analysis of certain special but great classes of systems it appeared that both the nested models could be joined into one model; then it is difficult to decide whether this case (see [12]) is reflective simulation or non-reflective one. The arising nesting simulation has interesting and important properties of both reflective and non-reflective simulation. We propose to consider the case to be ***reflective-plus***.

CONCLUSIONS

For classifying nesting simulation models, we applied only three criteria that seem to be sufficiently rational, easily implementable and clear at the present days. When taking into account the three basic criteria, it is rather impossible to give short names to all the cases produced by all possible combinations of the criteria. Therefore, we recommend to limit our considerations only to the cases that can be implemented (at least when using SIMULA language), i.e. to the cases with their depth one and two.

For nesting simulation of (static) depth one and (static) size one, we propose to use the term *simple nesting* that covers the terms *simple reflective nesting*, *simple non-reflective nesting* and *simple reflective-plus nesting*.

As regards nesting simulation of depth two and size $n > 1$, we recommend to use the term *nesting of size n* that covers the terms *homogeneous reflecting nesting*, *homogeneous non-reflective nesting* and *non-homogeneous nesting*.

For the cases with their depth greater than two, no special terms are so far proposed. Our opinion is that characterizing them by help of trees like in Fig. 3 or even Fig. 4 is the best method.

ACKNOWLEDGEMENT

This work was supported by the grant 201/06/0612 of the Czech Grant Agency as well as by the institutional research scheme MSM6198898701.

REFERENCES

[1] Berruet, P., T. Coudert, E. Kindler. Object-Oriented Reflective Simulation of Transitic Systems. In: The International Workshop on Harbour, Maritime and Multimodal Logistics Modelling & Simulation HMS 2003 (Y. Merkuryev, A. G. Bruzzone, G. Merkuryeva, L. Novitsky, E. Williams , Eds.). Riga : Riga Technical University, 2003, pp. 202-205.

[2] Kindler, E., T. Coudert, P. Berruet: Component-Based Simulation for a Reconfiguration Study of Transitic Systems, SIMULATION, 2004, vol. 80, no. 3, pp.153-163.

[3] Kindler, E. Simulation of Systems Containing Simulating Elements. In: Modelling and Simulation 1995, Proceedings of the 1995 European Simulation Multiconference. San Diego: Society for Computer Simulation International, pp. 609-613.

[4] Blümel, P., E. Kindler. Simulation of Antagonist Mutually Simulating Systems. In: Simulation und Animation '97(O. Deussen and P. Lorenz, Eds.). Erlangen, Ghent, Budapest, San Diego: Society for Computer Simulation International,1997,pp. 56-65.

[5] Kindler, E. Nesting Simulation of a Container Terminal Operating With its Own Simulation Model. JORBEL (Belgian Journal of Operations Research, Statistics and Computer Sciences), 2000, vol. 40, no. 3-4, pp. 169-181

[6] Kindler, E. Computer Models of Systems Containing Simulating Elements. In: Computing Anticipatory Systems CASYS 2000 (Daniel M. Dubois, Ed.). Melville, New York: AIP (American Institute of Physics), 2001, pp. 390-399.

[7] Kindler, E., Křivý, I. On the way to reflective simulation of hospitals. In: Proceedings of 4th International Conference Aplimat 2005. Part II. Bratislava: Slovak University of Technology, 2005, pp. 309-314.

[8] Křivý, I., E. Kindler. Reflective Simulation of in-Patients Dynamics. In: Proceedings of 5th International Conference Aplimat 2006. Part I. Bratislava: Slovak University of Technology, 2006, pp. 613-617.

[9] Křivý, I., E. Kindler. Synthesis of two Anticipatory Models in Design and Life-Cycle of Hospitals. To be printed in the International Journal of Computing Anticipatory Systems 2006.

[10] Mejtsky, J., E. Kindler. Diagrams for Quasi-Parallel Sequencing – Part I. SIMULA Newsletter, 1980, vol. 8, no. 3, pp. 46-49.

[11] Mejtsky, J., E. Kindler. Diagrams for Quasi-Parallel Sequencing – Part II. SIMULA Newsletter, 1981, vol. 9, no.1, pp. 17-19.

[12] Novak, P. Reflective Simulation With Simula and Java. In: Simulation und Visualization 2000 (T. Schulze, V. Hinz, P. Lorenz, Eds.). San Diego – Erlangen – Ghent – Delft:: Society for Computer Simulation International, 2000, pp. 183-196.

ABOUT THE AUTHORS

Prof. Dr. Ivan Křivý, PhD., University of Ostrava, Faculty of Science, Department of Computer Science, Phone: +420 596 160 273, E-mail: ivan.krivy@osu.cz.

Prof. Dr. Eugene Kindler, PhD., University of Ostrava, Faculty of Science, Dept. of Mathematics, Phone: +420 596 160 288, E-mail: evzen.kindler@mff.cuni.cz.