RIGA TECHNICAL UNIVERSITY

Arnis LEKTAUERS

DEVELOPMENT OF AN INTEGRATED APPROACH TO SIMULATION AND VISUALIZATION OF DISCRETE EVENT AND CONTINUOUS SYSTEMS

Summary of Doctoral Thesis

Riga 2008

RIGA TECHNICAL UNIVERSITY
Faculty of Computer Science and Information Technology
Institute of Information Technology

Arnis LEKTAUERS
Student of the Doctoral study programme
"Management Information Technology"

DEVELOPMENT OF AN INTEGRATED APPROACH TO SIMULATION AND VISUALIZATION OF DISCRETE EVENT AND CONTINUOUS SYSTEMS

Summary of Doctoral Thesis

Scientific supervisor
Dr.habil.sc.ing., professor
J.MERKURJEVS

RTU Publishing House
Riga 2008

UDK 681.511.017(043.2)
Le 414 d

Lektauers, A. Development of an integrated approach to simulation and visualization of discrete event and continuous systems. Summary of Doctoral Thesis. - R.: RTU Publishing House, 2008. - 30 p.

Printed in accordance with the decision No 08–03 of May 28, 2008 of the board of the Institute of Information Technology, Faculty of Computer Science and Information Technology, Riga Technical University.

This work has been partly supported by the European Social Fund within the National Programme "Support for the carrying out doctoral study programs and post-doctoral researches" project "Support for the development of doctoral studies at Riga Technical University".

ISBN 978-9984-32-891-1

DOCTORAL THESIS
SUBMITTED FOR THE DOCTORAL DEGREE
AT RIGA TECHNICAL UNIVERSITY

The defence of the thesis submitted for the doctoral degree in Engineering Science (Management Information Technology) will take place at an open session on December 8, 2008 in Meza Street 1/3, auditorium 202.

OFFICIAL REVIEWERS

Professor, Dr.habil.sc.ing. Aleksandrs Glazs
Riga Technical University, Latvia

Professor, Dr.habil.sc.ing. Pēteris Rivža
Latvia University of Agriculture, Latvia

Professor, Ph.D.comp.sc. Jerzy W. Rozenblit
University of Arizona, USA

APPROVAL

I confirm that I have developed this thesis submitted for the doctoral degree at Riga Technical University. This thesis has not been submitted for the doctoral degree at any other university.

Arnis Lektauers .... (signature)

Date: ....

The doctoral thesis is written in Latvian and includes introduction, 4 chapters, conclusions, 4 appendixes, bibliography, 66 figures and 16 tables in the main text, 142 pages. The bibliography contains 123 references.

TABLE OF CONTENTS

INTRODUCTION

Research motivation

The main motivation for the research is related to observable in the last years attempts to make simulation systems more accessible and friendly to the developers of simulation models and decision makers because up to now the simulation has only been used in a narrow circle of professionals, who should not only have an extended knowledge in that application area, in which a simulation model is developed, but should also be competent in the programming, probability theory and mathematical statistics. An important and perspective direction in the evolution of simulation is the use of computer graphics and visualization methods and their integration into the simulation systems [13]. Despite the broad use of visualization methods and tools in the simulation systems, there still exist several fully not solved problems of the integration of simulation and visualization.

One of the existing problems is the small number of theoretical studies and practical results in the area of simulation and visualization interaction. The vast majority of simulation software tools do not provide valuable possibilities of interacting with the model during simulation run; however, for making changes it is necessary to stop the simulation process and subsequently to continue.

The second existing problem is related to simulation in the research of discrete event and continuous systems. Attributable to it is the fact that in the traditional simulation systems of continuous processes the processing of discrete events provides large difficulties. In its turn, it is impossible or very hard to perform a discrete event simulation in discrete event systems. Although recently the number of software solutions for simulation of combined discrete event and continuous systems has been increasing, there still exists a problem that is related to the synchronization issues of simulation and visualization in the research of such systems.

The goal and the tasks of the thesis

The goal of the thesis is to develop an integrated approach and methods for simulation and visualization of discrete event and continuous systems, to practically implement them in a software system and to experimentally evaluate the developed system.

In order to achieve the goal, the following tasks are specified:

- a comparative research of visual simulation approach and methods aimed at identifying the unresolved tasks in their development and integration;
- development of integrated visual simulation approach and methods to resolve previously found unresolved tasks;
- development of architecture for interactive visual simulation of discrete event and continuous systems;
- practical implementation of system for visual simulation of discrete event and continuous systems;
- evaluation of the developed system by showing the possibilities of its practical use.

Research object and subject

Research object of the thesis is interaction process of simulation and visualization.

Research subject of the thesis is characteristic properties of simulation and visualization that determine functionality of an integrated visual simulation environment and methods that provide this appropriate functionality.

Research methods

Theoretical research in this thesis is based on systems theory and elements of discrete event system specification, set theory and computer visualization theory. Methods of software engineering, simulation and computer graphics are used for implementing the system prototype developed in the thesis.

Scientific novelty of the thesis

The main scientific novelty of the developed thesis is as follows:

- A system-theoretical formalism for interactive visual simulation is developed that is provided for development of integrated visual simulation systems using combined discrete event and continuous systems simulation approach. From typical simulation concepts it differs in a universal usability and integration of simulation and visualization functionality.
- Based on the visual simulation formalism there is developed an architecture, methods and algorithms for its practical implementation, from traditional simulation systems differing in a unified treatment of modelling, simulation and interactive elements that unifies and simplifies development and usage of simulation models.
- A concept of visualization frame for defining an integrated visual simulation context within interaction with a modelling and simulation frame is introduced.
- By implementing the developed visual simulation formalism it is established that the model-driven approach improves and simplifies the process of simulation model development and verification.
- An algorithm for integrated simulation visualization of continuous systems quantized states is developed allowing one to substantially increase the quantization step and thus to decrease the necessary computational resources without lowering the accuracy of the simulation results acquired.
- During approbation of the developed formalism in the form of practical software prototype two simulator variants are developed; it is proved that the priority queue algorithm is noticeably more efficient than the hierarchical simulator algorithm and thus more suitable for development and usage of practical visual simulation system.

Practical value of the thesis

Practical value of the thesis is related to the developed integrated simulation and visualization approach and implemented system prototype that unifies the methods of discrete event and continuous systems simulation and 2D/3D visualization. A combination of the listed techniques into a unified integrated environment allows the user to efficiently solve interactive simulation tasks.

Approbation of the obtained results

Presentations on the main results of the thesis are made at 8 international scientific conferences. The results obtained within the framework of the performed research are reflected in 7 papers published in international scientific publications approved by Latvian Council of Science:

1. Lektauers A., Merkuryev Y. 3D Visual Framework for Modelling and Simulation of Supply Chain Systems// Scientific Proceedings of the eLOGMAR-M Project. - Riga: JUMI Ltd., 2006. - 141-150 p.

2. Lektauers A. A Mixed Reality Framework for Visualization and Execution of DEVS-based Simulation Models// Proceedings of the 19^{th} European Conference on Modelling and Simulation "Simulation in Wider Europe", ECMS2005. - Riga: Publishing House of Riga Technical University, 2005. - 271-276 p.

3. Lektauers A. Developing a 3D Graphical User Interface for Object-Oriented Simulation Modelling// Rīgas Tehniskās universitātes zinātniskie raksti. 5.sēr., Datorzinātne. - Rīga: RTU, 2003. - 14.sēj., - 36-42 lpp.

4. Lektauers A. A Virtual 3D Environment for Logistics Modelling// Proceedings of the International Workshop on Harbour, Maritime and Multimodal Logistics Modelling & Simulation (HMS2003). - Riga: RTU, 2003. - 242-248 p.

5. Lektauers A. 3D-Projektierung von Simulationsmodellen in einer VR-Umgebung// Proceedings der Tagung "Simulation und Visualisierung '2003" der Otto-von-Guericke Universität Magdeburg (SimVis 2003). - Magdeburg: SCS Publishing House, 2003. - 313-322 S.

6. Lektauers A. Imitācijas modeļu izveide virtuālā 3D vidē// Rīgas Tehniskās universitātes zinātniskie raksti. 5.sēr., Datorzinātne. - Rīga: RTU, 2002. - 10.sēj., - 107-113 lpp.

7. Lektauers A. Imitācijas modeļu vizualizācijas tehnoloģija// Rīgas Tehniskās universitātes zinātniskie raksti. 5.sēr., Datorzinātne. - Rīga: RTU, 2001. - 5.sēj., - 103-109 lpp.

Structure of the thesis

The thesis includes introduction, four chapters, conclusions, four appendixes and bibliography. The formation and logical structure of the thesis is shown in figure 1.

The introduction motivates the research and defines research goals and tasks. The scientific methods applied, the novelty, practical value of the thesis, as well as approbation of the main results is described.

Chapter 1 "Integrated approach to simulation and visualization" is a theoretical chapter that contains an analysis and justification of usage and advantages of the discrete event system specification based on the system approach in order to determine drawbacks of the existing visual simulation systems and to determine general requirements for an integrated visual interactive discrete event and continuous systems simulation environment. A comparative analysis of visualization approaches and methods for their integration in the simulation is performed as well.

Chapter 2 "V-DEVS formalism" proposes an integrated visual simulation approach, which is described by the developed system-theoretical interactive visual discrete event and continuous systems simulation formalism called V-DEVS. On the basis of it, an interpolation method for visual simulation of quantized state systems is developed, the concepts of model-driven V-DEVS modelling are explored, as well as the V-DEVS visualization conveyor is developed.

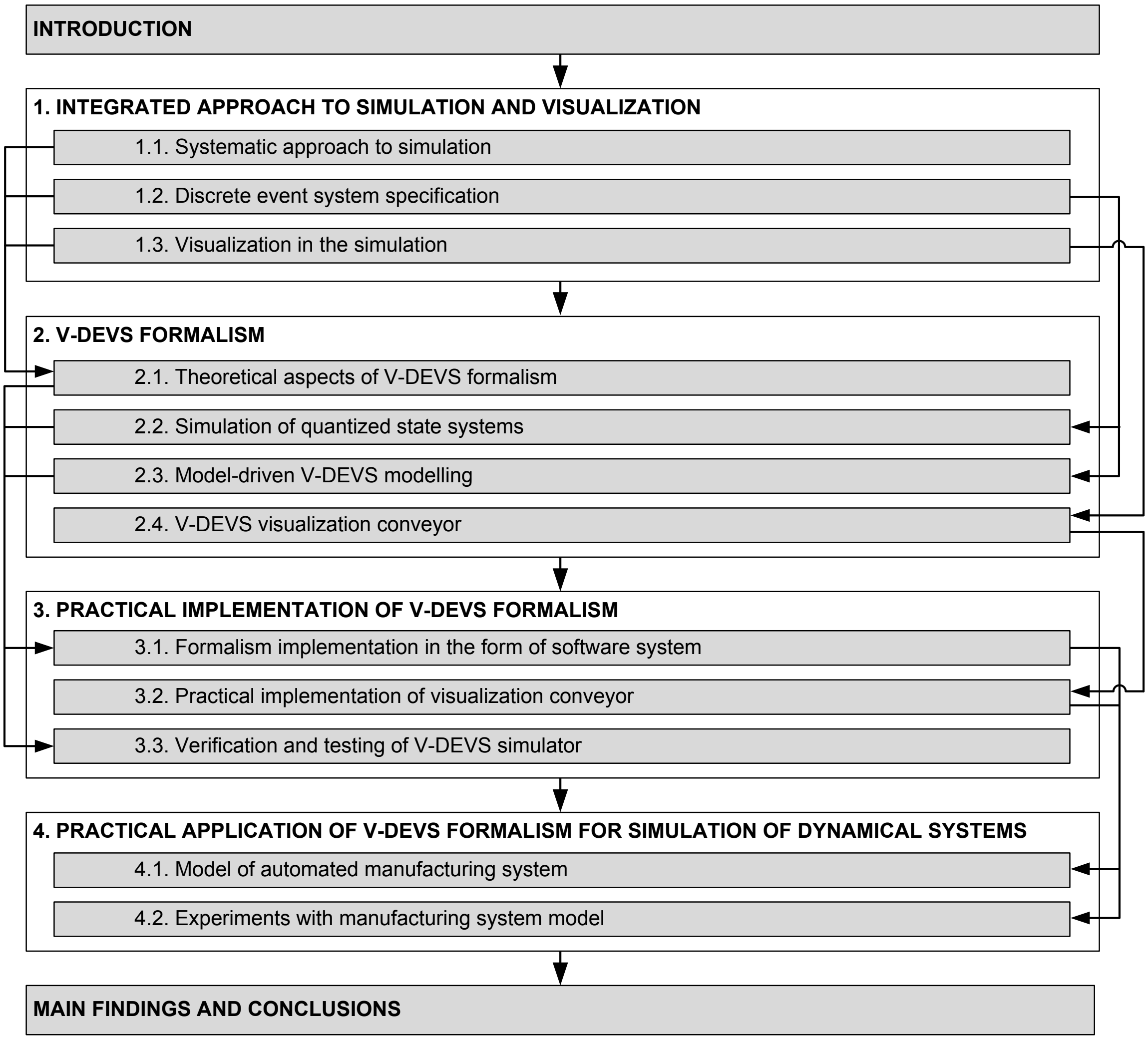


Fig. 1. The formation and logical structure of the thesis

Chapter 3 “Practical implementation of V-DEVS formalism” concretizes general requirements for an interactive simulation environment defined in Chapter 1 and describes practical implementation of V-DEVS formalism in the form of software system that includes elements of simulator, model-driven approach and visualization conveyor. Using the software prototype developed practical experiments for verification of its subjacent concepts and implementation details are performed.

In Chapter 4 “Practical application of V-DEVS formalism for simulation of dynamical systems” an V-DEVS model implementation of automated manufacturing system is laid out and the obtained experimental results showing the practical application possibilities of the developed formalism for interactive simulation of dynamical systems are analyzed.

Main findings of the research and conclusions are presented in the last part of the thesis.

1. INTEGRATED APPROACH TO SIMULATION AND VISUALIZATION

During the second half of the 20th century in many practical and theoretical scientific fields a wide use of the term complex systems began by applying it to complicated control systems of dynamical objects [31, 32] that consist of many differently built, variable functioning mutually coupled components. Practically any complex system can generally be treated as s combined system. With the term *combined system* are labelled systems that have observable both continuous and discrete activity aspects [29].

1.1. System approach to simulation

In the analysis and synthesis of complex (large-scale) systems there is widely used a system approach that differs from the classical (inductive) approach in the transition from the general to a detailed level with a subjacent goal by extracting the explorable object from the environment [30].

The terms *model* and *simulation* are often used as synonyms although they have strictly determined and different meaning that is defined by the *Framework for Modelling and Simulation* [8]. Based on the systems theory, the framework for modelling and simulation provides base concepts for simulation environments.

1.2. Discrete event system specification

In order to provide conceptualization and specification of simulation models of complex systems, various available paradigms, formalisms, modelling methodologies and simulation methods can be used. The discrete event system specification is a general universal formalism of this kind that is provided for the description and definition of discrete event systems dynamics, but is applicable for the representation of traditional formalisable systems too, for example, of differential equations (continuous systems) and difference equations (discrete time systems). DEVS (*Discrete Event System Specification*) formalism was developed by the American scientist Bernard Zeigler [24, 25, 26] in the early 1970s in order to ensure the development of discrete event simulation models in a hierarchical and modular manner.

Although DEVS formalism is a universal framework for discrete dynamical systems modelling there exist several dedicated DEVS formalism varieties that conform to certain system classes and problem domains.

1.3. Visualization in the simulation

For association of computer graphics and visualization terms with modelling and simulation concepts, there have to be mentioned such terms as *Visual Interactive Simulation* [2] and *Visual Interactive Modelling* [23]; they, however, are generally related not to simulation in its classical sense but to such information technology areas as scientific visualization, computer aided design, computer games and learning systems.

According to [27], visualization is the transformation process of data, information and knowledge into a graphical depiction with the goal to provide data analysis, information exploration, information interpretation, trend forecasting, pattern recognition etc. In its turn, the computer graphics includes all aspects of graphical image synthesis, rendering and manipulation [3].

Since the term *model* depending on its usage context has many meanings [10], from the viewpoint of an integrated simulation and visualization it is necessary to more precisely define the model term too. The model can be both an abstraction of the logic and dynamics of the modelled source system and a depiction of the physical characteristics and spatial appearance in two or three dimensions, therefore the author of the thesis proposes a classification that has decidable terms of logical simulation model and visual model (Figure 1.1). In its turn, the visual model depending on representation context can be divided into the subtypes of logical visual model and physical visual model.

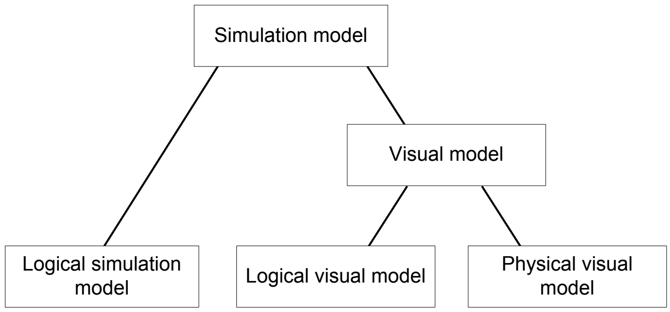


Fig. 1.1. Classification of simulation models from an integrated view

The main motivation for the visual interactive simulation is an integration of visualization and interactive control tools allowing the user to observe the operation process of model and to interact with it during the experimentation time. It is possible to modify the parameters and variables of the model, as well as the visual depiction during the experimentation time with an immediate influence on the simulation process. Depending on the degree of the interaction between user and model there can be distinguished three interaction classes: *Post-processing*, *Control* and *Steering* [17]. On that ground it is possible to create a classification of simulation and visualization integration forms (see Figure 1.2).

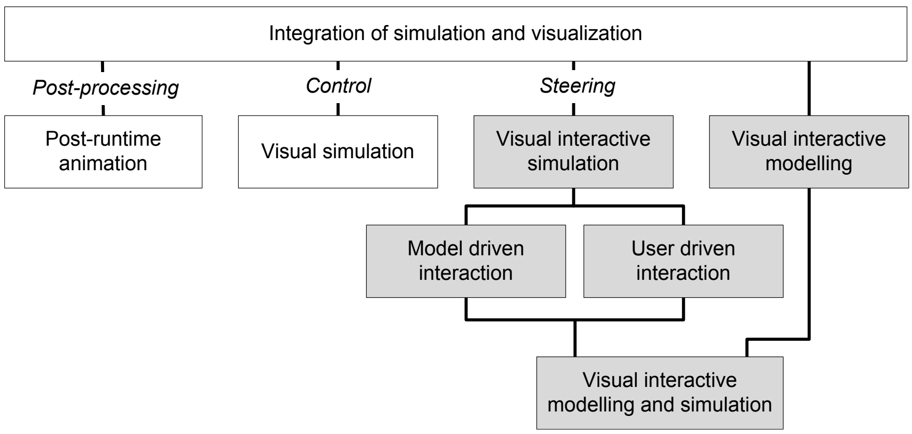


Fig. 1.2. Classification of simulation and visualization integration forms

The dynamical visualization or animation enables the user to visually observe the dynamics of the simulation model behaviour and increases the effect of model credibility by giving a more complete review about the simulated system therefore it can serve as an effective simulation supplementary aid in different related areas and aspects of simulation [21]:

- verification and validation;
- understanding of results;
- communication of results;

- getting by-in from nonbelievers;
- achieving credibility for the simulation.

The existence of an interactive funding during the simulation processing opens relevant advantages to the model user - the possibilities of changing model parameters and behaviour without the need to stop the simulation cycle and then to restart it. Such a possibility is the most useful during the model verification and validation when the tuning of model parameters and structure is performed. Thus the user interaction with the model during the simulation run can be considered as a supplementary aid for model verification and validation.

1.4. Conclusions

In this chapter obtained results are the following:

- a visualization classification is developed that is oriented towards use in the theoretical research and practical implementation of simulation systems;
- the advantages provided by a user interaction during the simulation run, are identified;
- requirements for an interactive simulation system are generally formulated.

The main conclusions are as follows:

- The system approach based DEVS formalism provides a universal formal basis and multi-formalism approach for simulation of combined discrete event and continuous systems.
- The visualization depending on its application goal and audience proves to be an effective supplementary aid for different simulation steps:
  - visualization as an information mining tool;
  - a supplementary verification aid of simulation models during the simulation run;
  - a supplementary validation aid of simulation models;
  - analysis tool for explanation of different simulation runtime processes;
  - communication tool for simulation experts, planners or users;
  - presentation tool;
  - training tool.
- The interactive visual modelling enables the user to change model parameters and to affect its behaviour during the simulation run allowing to speed up the verification and validation process of simulation models.

2. V-DEVS FORMALISM

Grounding on a research of integrated approaches in simulation and visualization it is possible to come to the conclusion that is necessary to create a unified system-theoretical framework that meets both the simulation, and interactive visualization requirements. Therefore in this thesis a new formalism is proposed - *Visual Discrete Event System Specification* (V-DEVS) that is provided for an integration of discrete event and continuous systems modelling and 2D/3D visualization. The basic idea and the discrete event modelling structure is borrowed from the classical DEVS formalism, but the continuous state simulation elements are borrowed from DEVS formalism extensions - DESS [26], DEV&DESS [19] and GK-DEVS [12] (Figure 2.1).

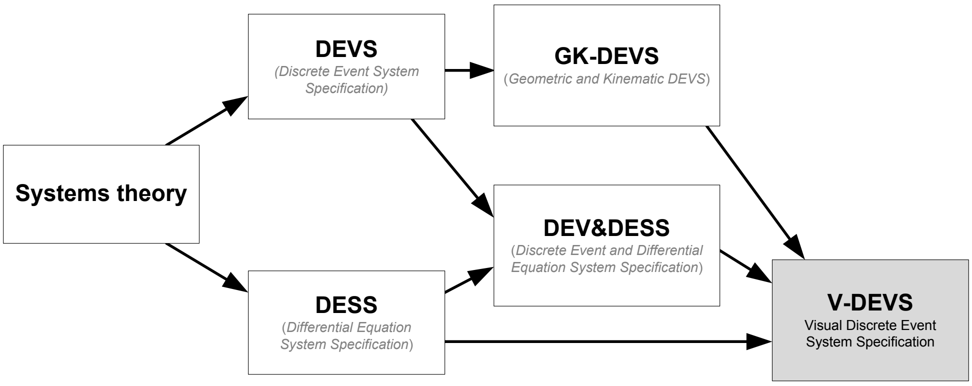


Fig. 2.1. The linkage of V-DEVS formalism with other extensions of DEVS formalism

In Table 2.1 a comparison of the possibilities of V-DEVS formalism with other linked DEVS extensions is given.

Table 2.1

Comparison of DEVS formalisms ("+" - the possibility is provided, "-" - the possibility is not provided)

Possibilities of the formalism	Classic DEVS	DEV&DESS	DESS	GK-DEVS	V-DEVS
Discrete system modelling	+	+	+	+	+
Continuous system modelling	-	+	-	-	+
Animation of object kinematics	-	-	-	+	+
Integration of simulation and visualization	-	-	-	-	+
Interactive simulation	-	-	-	-	+

2.1. Theoretical aspects of V-DEVS formalism

In the same way as an experimental frame for assigning observations made within the simulation framework of the modelled system or experiment conditions is defined, it is necessary to introduce a term of visualization frame for defining experimental conditions of the modelled system in relation to the graphical process of simulation. The *Visualization Frame* is a simulation structure that defines conditions by which a creation and execution of the visual model of the modelled system, as well as the user interaction during the simulation run occurs. The advantage of the visualization frame in the structure of simulation and visualization is the possibilities of separating the simulation process from the graphical depiction by defining a formal basis for visual modelling and simulation.

An atomic V-DEVS model can be defined as the following cortege:

$$AM_{V-DEVS} = \left\langle X, Y, S, \delta_{ext}, \delta_{int}^{discr}, \lambda^{discr}, ta^{discr}, \delta_{int}^{cont}, \lambda^{cont}, ta^{cont} \right\rangle, \tag{2.1}$$

where $X = \langle X^{discr}, X^{cont} \rangle$ - a set of discrete and continuous inputs;

$Y = \langle Y^{discr}, Y^{cont} \rangle$ - a set of discrete and continuous outputs;

$S = S^{discr} \times S^{cont}$ - a set of sequential states as a Descartes multiplication of discrete states S^{discr} and continuous states S^{cont};

$\delta_{ext} : Q \times X \to S$ - external transition function, where

$Q = \{(s,e) \mid s \in S, 0 \le e \le ta^{discr}(s)\}$ - total state set;

$s \in S$ - the state where system is located since the last state transition;

e - elapsed time since the last discrete state transition;

$\delta_{int}^{discr} : S \to S$ - discrete event internal transition function;

$\lambda^{discr} : S \to Y^{discr}$ - discrete event output function;

$ta^{discr} : S \to \mathbb{R}_{0,\infty}^{+}$- discrete event time advance function;

$\delta_{int}^{cont} : S^{cont} \to S^{cont}$ - continuous internal transition function;

$\lambda^{cont} : S \to Y^{cont}$ - continuous output function;

$ta^{cont} : S \to \mathbb{R}_{0,\infty}^{+}$ - continuous time advance function.

In Figure 2.2 a general dynamics of V-DEVS atomic model activity is depicted.

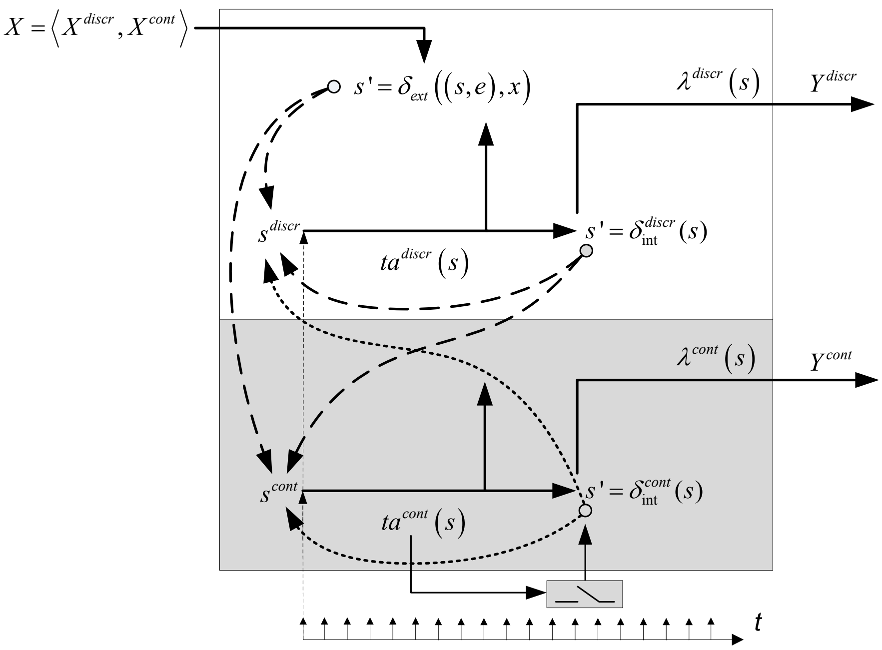


Fig. 2.2. The dynamics of V-DEVS atomic model activity

2.2. Simulation of quantized state systems

DEVS formalism provides a formal basis for simulation of continuous systems using discrete tools by utilizing quantized state algorithms [5]. The application of quantized state algorithms makes it possible to decrease the necessary simulation step count, providing at the same time that all essential events in the modelled system are taken into account. However, in developing the V-DEVS visualization conveyor and performing practical experiments in the integration of quantized state simulation models a problem can be observed that not always the quantization step $\triangle q$ that is suitable for the experimentation needs is equally suitable for execution of dynamical visualization or animation, because the model output values can change in a relatively large range. For the animation purposes it is necessary that model output values change uniformly creating an effect of simulation continuity. For solving the given problem this work proposes a concept of V-DEVS interpolator that allows an effective coupling of quantized state simulation and visualization.

Figure 2.3 illustrates a working concept of V-DEVS interpolator: on the left-hand side a quantized state system is reproduced, but on the right-hand side - a V-DEVS interpolator attached at the output of this system.

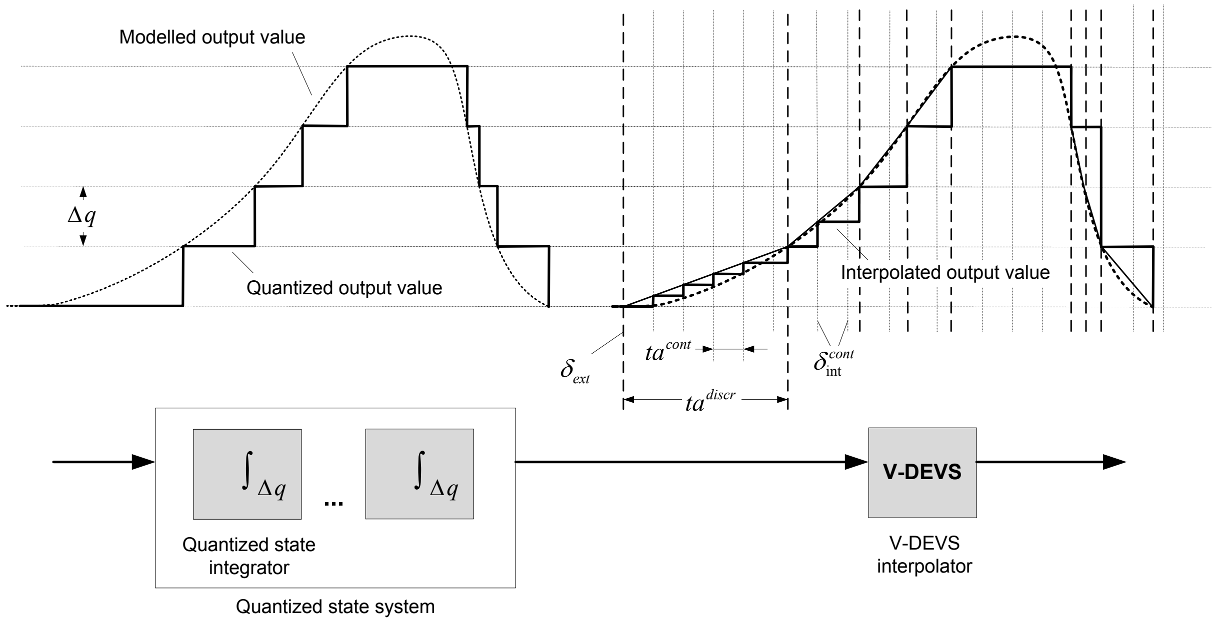


Fig. 2.3. Working diagram of the V-DEVS interpolator

The output values of quantized state system simulation process change with a discrete variable value size step $\triangle q$ during time ta^{discr}. The task of the interpolator is to transform these discrete changes into a regular discrete time ta^{cont} flow of changes.

2.3. Model-driven V-DEVS modelling

Development of a simulation model at an early system development phase is a complicated task, because it requires an extended knowledge of modelling methods, system problem domain and model execution paradigms. Here the collaboration between system experts and modelling experts, which usually have diametrically opposite competence and experience, is necessary. Therefore it is necessary to have a practical and effective way to use modelling and simulation framework in simulation even at early development step. There exist various techniques and tools for solving that problem; one of most widely used concepts and tools is the unified modelling language UML [20, 28] and the *Model-Driven Architecure* (MDA) [11]. The underlying idea of the model-driven architecture is the idea of system development by using high level graphical notation allowing the developer to concentrate on the problem nature less heed giving to smaller details.

For the model-driven approach in V-DEVS formalism there should be defined the following requirements:

- the metamodel should provide a syntactical structure for the description of the modelled system;
- the existence of modelling element graphical depiction is necessary;
- the proposed metamodel should provide a flexible enough mechanism for the consistent description of the modelled system.

In Figure 2.4 a general V-DEVS simulation metamodelling architecture is shown. The V-DEVS metamodel defines a syntactical and semantic structure of the developed V-DEVS simulation model, and it is a basis for the description of logical and visual models. The V-DEVS metametamodel defines a syntax and semantics of V-DEVS metamodel, and it is provided for the description of model library elements.

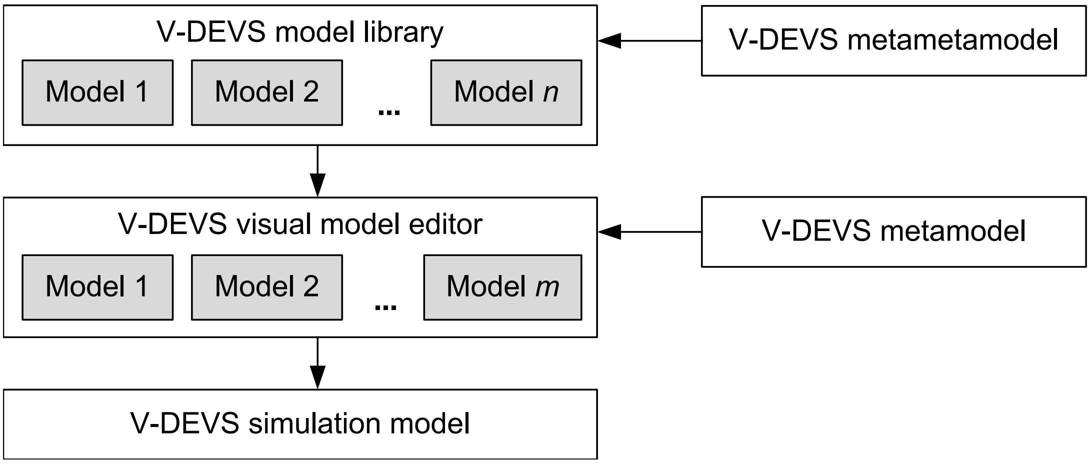


Fig. 2.4. A general architecture of V-DEVS simulation metamodelling

2.4. V-DEVS visualization conveyor

Based on the V-DEVS formalism proposed in the work it is possible not only to develop the concept of integration of simulation and visualization of discrete and continuous systems, but also to apply this formalism to implement to the visualization subjacent computer graphics systems. Since the concept of visualization conveyor is based on the use of mutually coupled hierarchically structured components, there arises a possibility of applying the V-DEVS mathematical appliance therefore merging into a unified context not only the aspects of simulation, but also of computer graphics. In the work, two forms of V-DEVS conveyor are proposed - event driven and demand driven variant of the conveyor.

The advantage of event driven visualization conveyor is its relatively simple structure (Figure 2.5). However every change in any element input causes a state updates in all coupled elements. The necessary count of input/output ports N_P for the information exchange between the elements of event driven conveyor can be determined by the following equation:

$$N_P = S_O + 2F + S_I, \quad (2.2)$$

where S_O - source element count;

F - filter element count;

S_I - output element count.

In Figure 2.6 a general structure of demand driven V-DEVS visualization conveyor is presented. The advantage of this variant is that processed are only those elements that directly affect the visualization result.

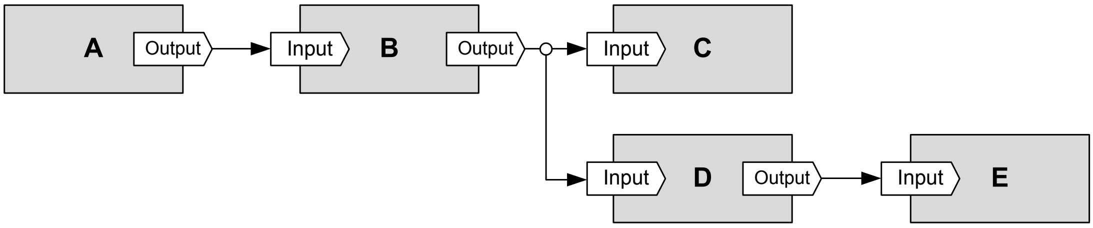


Fig. 2.5. Structure of event driven V-DEVS conveyor

The necessary count of input/output ports N_P for the information exchange between demand driven conveyor elements can be determined by this equation:

$$N_P = 2S_O + 4F + 2S_I. \quad (2.3)$$

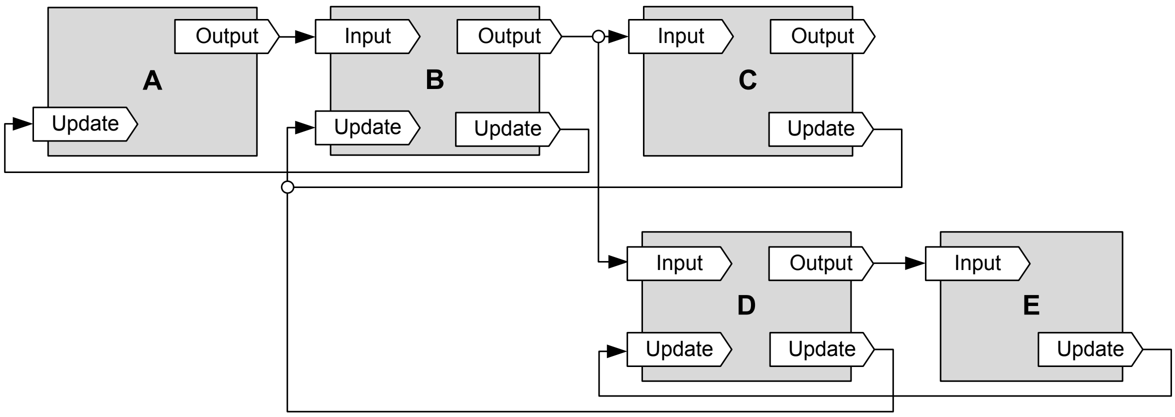


Fig. 2.6. Structure of demand driven V-DEVS conveyor

2.5. Conclusions

In this chapter the achieved results are the following:

- a concept of visualization frame for defining an integrated visual simulation context by the interaction with modelling and simulation framework is introduced;
- a V-DEVS formalism that is based on systems theory is developed;
- a methodology for integrating simulation and visualization into simulation of quantized state systems by applying interpolation of quantized states is created;
- a model-driven approach to an implementation of V-DEVS formalism is developed;

- a V-DEVS visualization conveyor for the integration of simulation and visualization processes is elaborated.

The main conclusions are the following:

- The developed V-DEVS formalism is a combined concept of discrete event and continuous systems simulation because it contains simulation components of both discrete event and continuous processes:
  - the discrete component is equivalent to atomic model defined by classical DEVS formalism used for modelling discrete events;
  - the continuous component defines modelling of continuous processes, and it is universally applicable to tasks of both simulation and visualization.
- V-DEVS formalism provides a formal basis for interactive visual modelling of combined systems.
- The advantages of the V-DEVS visualization conveyor are as follows:
  - system approach to providing visualization process;
  - a unified principle of inputs, outputs and synchronization with V-DEVS;
  - event driven or demand driven processing of visualization flow;
  - the possibility of merging the well-known in the computer graphics scene graph architecture and traditional visualization pipeline.

3. PRACTICAL IMPLEMENTATION OF V-DEVS FORMALISM

This chapter is devoted to the practical implementation of the system-theoretical V-DEVS formalism described in the previous chapter and to solving related problems by practically implementing the previously obtained theoretical results in the form of an interactive 3D V-DEVS simulation system prototype.

3.1. Formalism implementation in the form of a software system

In Figure 3.1 the architecture of the V-DEVS practical implementation developed by the author of this work [15, 16] is shown that consists of three subsystems - modelling system, visualization system and V-DEVS simulator.

For the development of system architecture there is employed an integrated approach [9], in the last time ever more used in simulation systems, by merging different source and form inputs, for example, using the processing of statistical and graphical information in a unified platform.

Within the framework of this thesis, two V-DEVS simulator versions are developed - a hierarchical simulator algorithm and priority queue algorithm. Both algorithms are grounded on a common base algorithm (Figure 3.2).

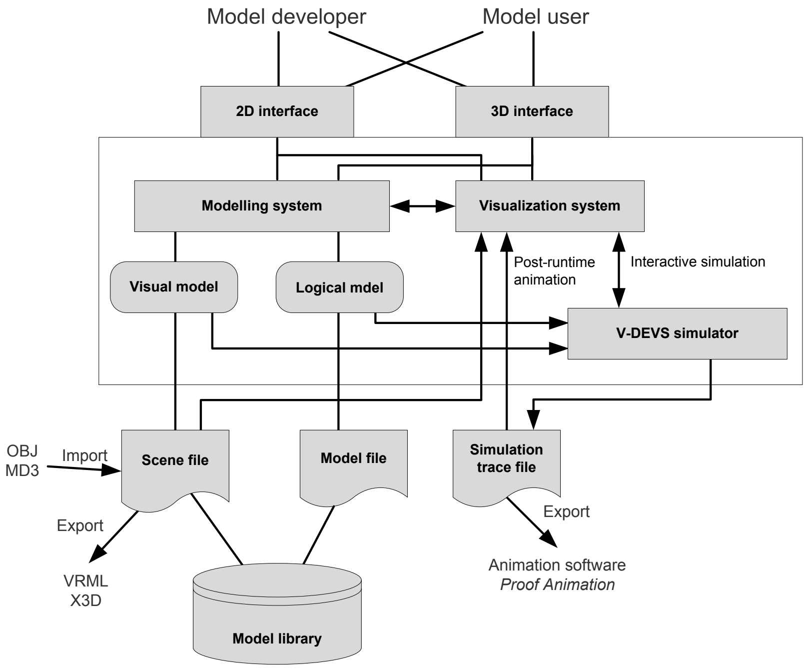


Fig. 3.1. Implementation architecture of V-DEVS system

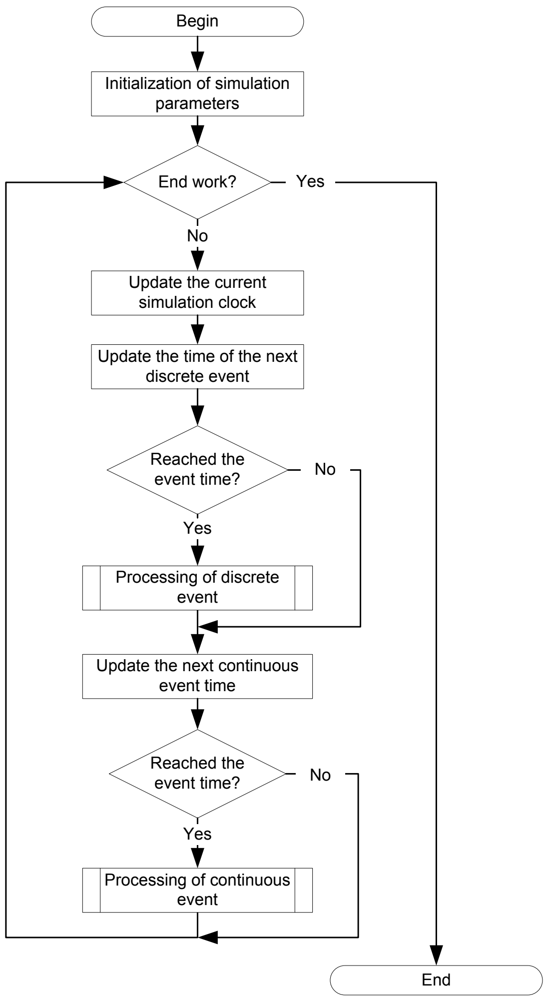


Fig. 3.2. Base algorithm of V-DEVS simulator

The hierarchical and modular structure of the V-DEVS model is based on the classical specification of DEVS simulators [26]. Each atomic model has an associated simulator object. In its turn, each coupled model has an associated coordinator object, and therefore the count of coupled models is equal to the count of coordinators, but the count of atomic models is equal to simulator count. The simulator performs an execution of atomic model, but the coordinator - the execution of coupled model.

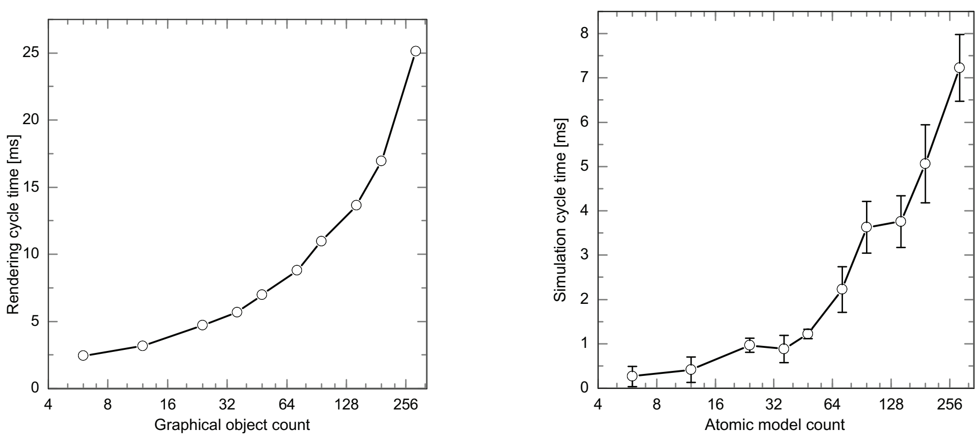
The implementation of priority queue V-DEVS algorithm is built in such way that the simulation efficiency can be improved with respect to the following aspects:

- abandonment of unnecessary use of simulator and coordinator objects;
- speed up of event scheduling by processing only active models;
- abandonment of unnecessary use of internal synchronization messages;
- abandonment of unnecessary use of event routing messages.

3.2. Practical implementation of visualization conveyor

The practical implementation of the V-DEVS visualization conveyor is based on the previously described theoretical statements of the visualization conveyor. Within this work the V-DEVS visualization conveyor is implemented by integrating V-DEVS principles with existing computer graphics and visualization software systems. The V-DEVS formalism as a universal paradigm of discrete event and continuous systems modelling that is independent of a concrete realization is relatively simple to integrate with both existing scene graph and data flow visualization systems. In the implemented prototype an integration of V-DEVS simulator with an open source scene graph library JME (JMonkeyEngine) and open source visualization library VTK [22] is performed.

For clarifying the performance of V-DEVS simulator in collaboration with a scene graph system, during the development of software prototype several experimental tests are carried out, which show (Figure 3.3) that this is the used scene graph and visualization library that most constitutively affects the total simulator performance.



(a) The dependence of scene graph rendering time on the graphical object count

(b) The dependence of V-DEVS simulation cycle time on the scene graph atomic model count

Fig. 3.3. The performance of V-DEVS simulator in collaboration with a scene graph library

3.3. Verification and testing of V-DEVS simulator

By implementing of V-DEVS formalism in a practical simulation system prototype it is necessary to verify that the used algorithms are correct - consequently it is necessary to perform a validation and testing of the developed system.

A typical simulation example of combined discrete event and continuous system is characterized by a model of ball bouncing down-stairs [5, 14]. The given model is realized with a goal to perform a verification of the practical implementation of V-DEVS quantized state interpolator and to estimate its working efficiency in solving simulation tasks.

The V-DEVS simulation model whose structure is shown in Figure 3.4 implements a mathematical formulation of the movement of a ball bouncing down-stairs. For providing a steady animation process during the simulation run there is used V-DEVS 3D quantized state interpolator.

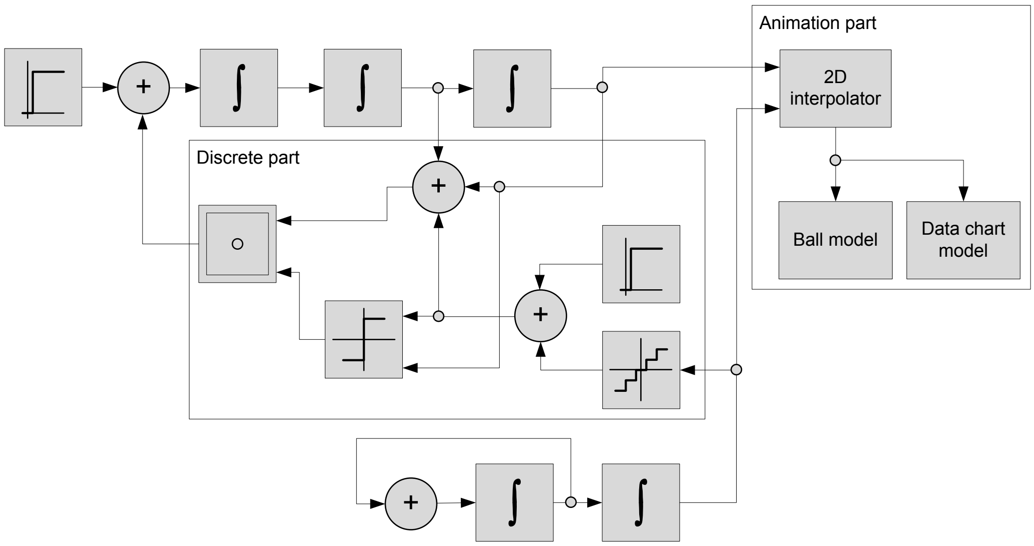
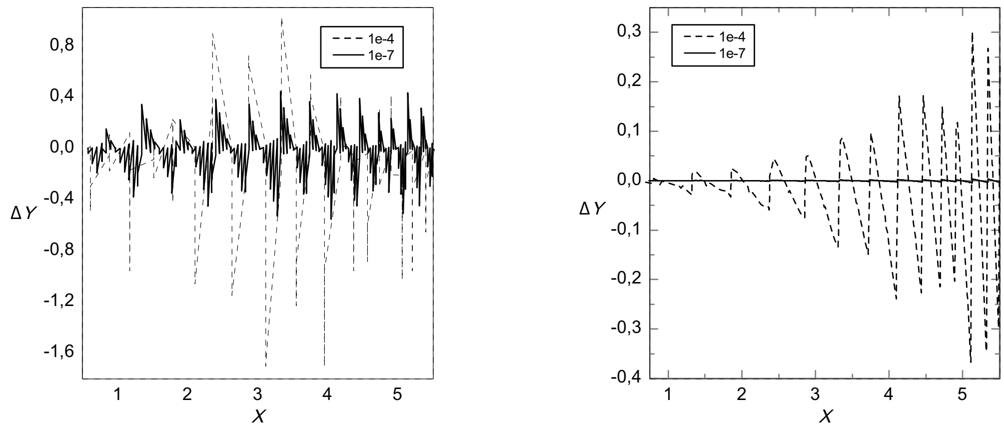


Fig. 3.4. Model structure of ball bouncing down-stairs

The ball bouncing down-stairs moves in two dimensions thereby the rolling conditions are dependent on two variables (x and y). In using a visual V-DEVS simulation of quantized states for the accomplishment of this problem a problem arises that the available quantization step Δq that is used for obtaining sufficiently accurate modelling results is not suitable for providing a regular animation process. Therefore for solving this problem there is used a quantized state interpolator introduced in this work.

To estimate the efficiency of the proposed V-DEVS quantized state interpolator with the given simulation model, a number of experiments are performed by comparing the trajectory data of the ball movement that are obtained with and without using the quantized state interpolator. In Figure 3.5 the obtained residual data are shown by comparing the chosen as a standard data ($\triangle q = 1,0 \times 10^{-7}$) with the data that are obtained at a greater value of the parameter $\triangle q$.

As can be seen from Figure 3.5, the smaller the chosen value of parameter $\triangle q$ (a greater accuracy of the simulation model), the smaller the difference between obtained trajectory data; and thus the data of the comparable trajectories better approximate the standard trajectory. From the simulation results seen in Figure 3.5 is possible to conclude that a use of V-DEVS interpolator allows improving the accuracy of the model output approximately 5 times using a fixed value of the parameter Δq.



(a) Residual data not using an interpolation

(b) Residual data using an interpolation

Fig. 3.5. Residual data $\Delta Y = Y_{1,0\times 10^{-7}} - Y_{\triangle q}$ of the ball movement trajectory obtained by different values of the quantization step $\triangle q$

3.4. Conclusions

The achieved results are the following:

- a practical architecture of the V-DEVS formalism is developed;
- two versions of V-DEVS algorithm implementation are developed; the first algorithm is based on the classical specification of V-DEVS simulators, in its turn the second algorithm is based on the principles of use of one universal simulator and priority queues;
- a model-driven approach to development of V-DEVS simulation models is developed;
- a V-DEVS visualization conveyor for an interaction with a simulation model is developed.

The main conclusions are the following:

- the practical implementation of the V-DEVS formalism and the performed experimental tests show that the proposed theoretical V-DEVS concepts are valid and on their basis it is possible to develop a real simulation system;
- the advantage of the realized simulator algorithms is that they are not directly dependent on the used visualization system because the visualization task is realized in the level of atomic models;
- the application of the developed quantized state interpolator for simulation of continuous systems allows the user to substantially increase the visualization accuracy by a fixed consumption of the computational resources;
- the model-driven approach improves and simplifies the development and verification processes of simulation models.

4. PRACTICAL APPLICATION OF V-DEVS FORMALISM FOR SIMULATION OF DYNAMICAL SYSTEMS

In this chapter a practical application of the V-DEVS formalism for simulation of dynamic systems is described by examining a simulation model of an automated manufacturing system. The main goal of the simulation model development is to show the application possibilities of the realized V-DEVS formalism and to experimentally verify the performance of the realized algorithms. The description and analysis of the examined manufacturing system model are performed grounding on the V-DEVS theory and concepts by characterizing model architecture, identifying components, their mutual interaction and influence, as well as the type of the required specification.

4.1. Model of automated manufacturing system

In order to be able to compete within the circumstances of the progress and competition of the global market manufacturing companies must on an ongoing basis to expand and improve their manufacturing capabilities by introducing modern automated manufacturing systems [6]. DEVS based application research in the analysis of automated manufacturing systems are realized previously too [7], though a simulation environment that is based on V-DEVS formalism includes both discrete and continuous processes, which doubtless occur in automated manufacturing systems.

The model basic structure (Figure 4.1) consists of generators, terminator of simulation entities, interactive elements (slider and check box), workstations of manufacturing production assembly (painting station, assembly robot, assembly station or assembler) and conveyors for production transportation between workstations.

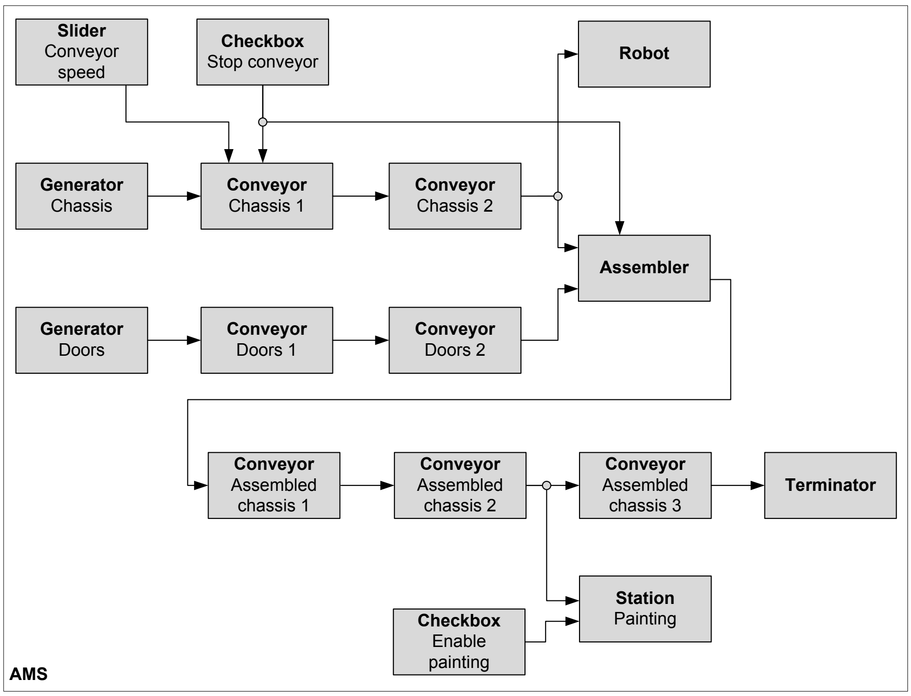


Fig. 4.1. Model structure of automated manufacturing system

Generator models generate the material flow incoming in the system - car chassis and doors.

The car chassis and doors go into an assembler model through two different conveyors. The assembler simulates the mounting process of doors into a cars chassis. Further through other conveyor the mounted chassis moves forward the painting station until it reaches a terminator. The terminator model is an element that performs the terminating of received entities - mounted car chassis that corresponds to a leaving of material flow in the modelled system.

In order to verify and demonstrate the implementation possibilities of the discussed manufacturing system in an existing commercial simulation system, as well as to compare the realized prototype of V-DEVS simulation environment there is performed development and experimental tests of the simulation model in the Flexsim environment. The realized experiments show the suitability of the system conceptual model, however, it is not possible to directly compare the Flexsim environment with the performance test results of V-DEVS system prototype because the algorithmic implementation and architecure of Flexsim is not open and available for exploration.

In Figure 4.2 3D image of the simulation model is shown that is developed with the V-DEVS simulation prototype. From the user's point of view, the main differences and advantages comparing to typical commercial simulation systems are the possibilities of simulating combined discrete event and continuous processes and modeless interacting with a simulation model during a simulation run.

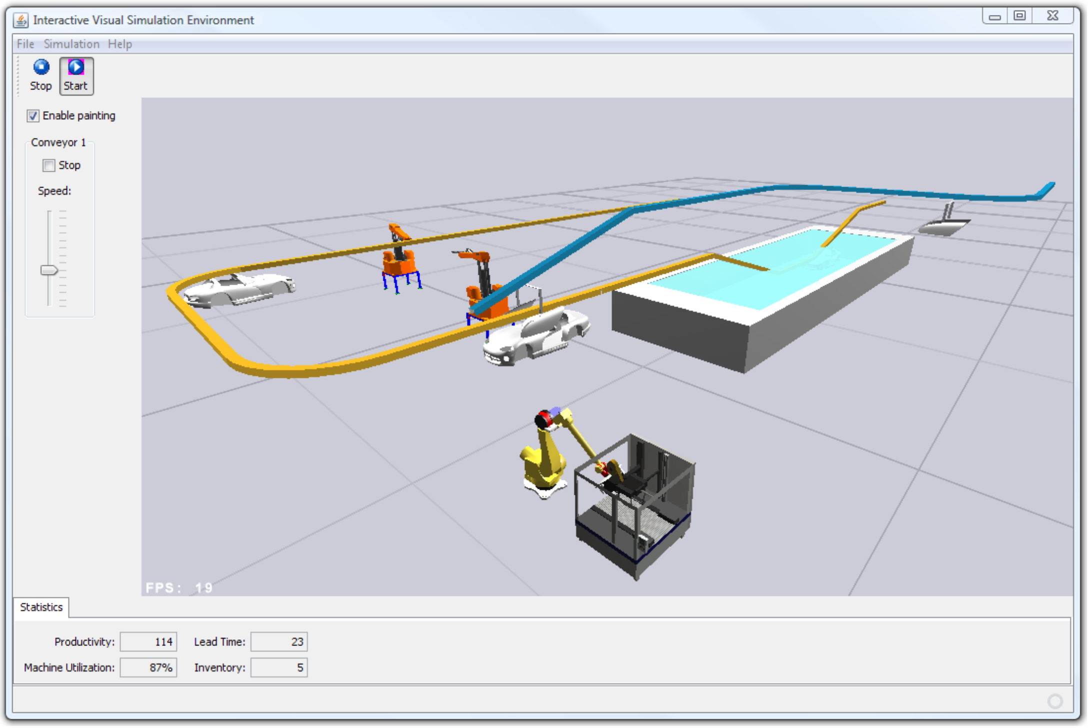


Fig. 4.2. Model of automated manufacturing system in V-DEVS visual simulation environment

4.2. Experiments with manufacturing system model

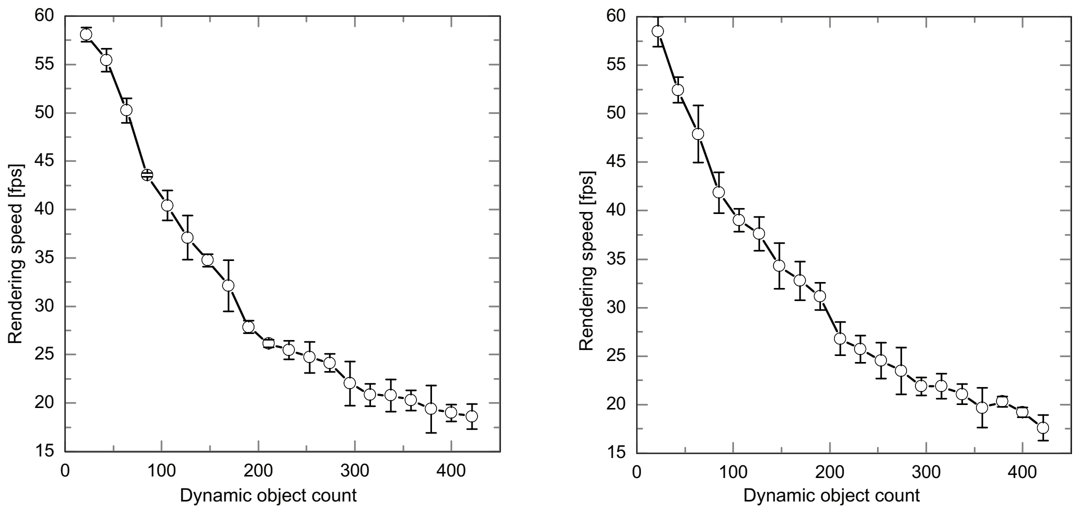
The simulation process is integrated with real time visualization; therefore a question arises: what is the influence of the visualization on the performance of simulation process? The first cycle of experiments is performed to obtain an answer exactly to this question by changing for this purpose the count of conveyors and generators in a range [6; 25]. Table 4.1 shows the dependence of generated simulation entities on the count of generators and conveyors in the model.

Table 4.1

Dependence of generated simulation entities on the count of conveyors in the system

Conveyor count	6	7	8	9	10	11	12	13	14	15
Count of simulation entities	22	43	64	85	106	127	148	169	190	211
Conveyor count	16	17	18	19	20	21	22	23	24	25
Count of simulation entities	232	253	274	295	316	337	358	379	400	421

In Figure 4.3 the dependence of simulation rendering speed on the generated simulation entities or dynamic objects is shown by using hierarchical and priority queue algorithms. By increasing the count of dynamic objects in the model both the hierarchical and priority queue based rendering performance of V-DEVS simulator decreases, which is basically determined by the computer graphics software and hardware used.

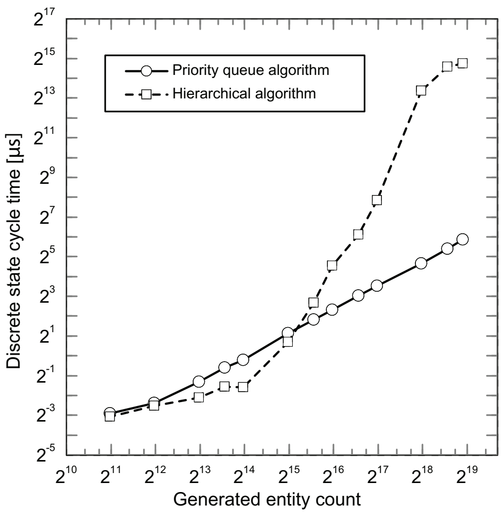


(a) Hierarchical algorithm (b) Priority queue scheduling algorithm

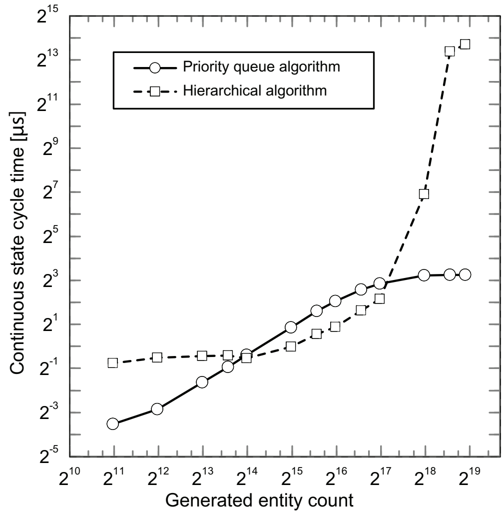
Fig. 4.3. Dependence of simulation model rendering speed on the count of dynamic objects in the model

In Figure 4.4 the dependence of discrete state S^{discr} and continuous state S^{cont} processing time on the count of generated simulation entities in the model is shown. By increasing the count of generated entities in the model, the performance of V-DEVS hierarchical simulator algorithm decreases faster than by using a priority queue algorithm.

Figure 4.5 depicts the dependence of discrete and continuous state events on the count of generated simulation entities in the model. The count of discrete and continuous state events that are processed by the hierarchical simulator noticeably increases as the count of generated entities increases, until a saturation point is reached where the simulation execution system has no more possibilities to process all the generated events. As can be seen from Figure 4.5, such saturation point of discrete and continuous state event processing by a hierarchical simulator appears when the count of generated entities reaches a value 2^{14}. After this point the count of processed events begins to decrease. That means that in this way obtained results are not credible anymore because the system is not able to process all events.



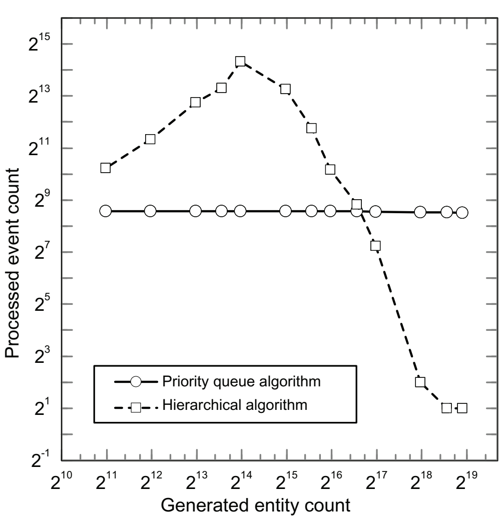
(a) Discrete event simulation



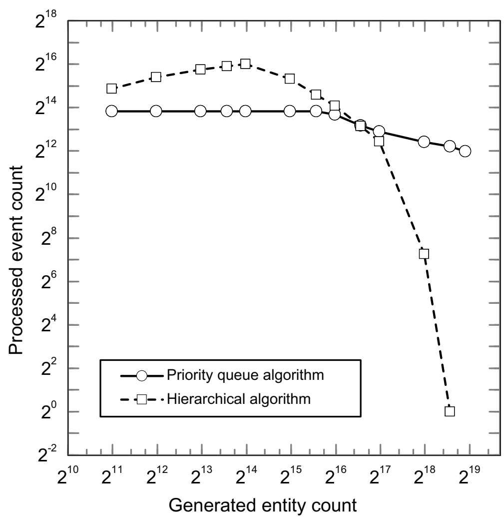
(b) Continuous state simulation

Fig. 4.4. Dependence of discrete state S^{discr} and continuous state S^{cont} processing cycle time on the count of generated simulation entities in the model

In its turn, the discrete event processing performance of priority queue algorithm in the framework of performed experiments is practically independent of the count of generated entities. But the continuous event processing performance of this algorithm begins to decrease when the count of generated entities reaches a value 2^{16}. It means that in the processing of continuous events the performance of priority queue algorithm is approximately $2^{16}/2^{14} = 4$ times greater than that of hierarchical simulator algorithm.



(a) Discrete event simulation



(b) Continuous state simulation

Fig. 4.5. Dependence of discrete and continuous state event count on the count of generated simulation entities in the model

Statistical performance indices of the discussed automated manufacturing system are determined with a goal to show that the realized V-DEVS formalism and based on it prototype is applicable for simulation of practical systems, and the obtained statistical results are adequate. For testing the suitability of the obtained results there is performed a comparison with similar results that are obtained with the commercial simulation tool Flexsim.

For obtaining statistical indices of the performance there are performed 5 experiments by changing conveyor speed in a range $[0,1; 0,5]$ m/s. Each experiment is repeated 10 times, and the length of each simulation cycle is selected corresponding to 720 hours or 30 days of real time manufacturing. The average statistical performance indices obtained with a prototype of interactive visual simulation environment are shown in Table 4.2. For statistical estimation of the modelled automated manufacturing system performance there is used a methodology discussed in the works [1, 4, 18].

The obtained simulation results using V-DEVS simulation prototype and Flexsim tool are similar showing that the realized interactive simulation prototype allows one to achieve adequate simulation results.

Table 4.2

Average statistical indices of the automated manufacturing system performance obtained in the V-DEVS visual simulation environment

			Assembler	
Conveyor speed (m/s)	Assembled chassis count	Manufacturing lead time	Chassis queue length	Doors queue length
0,1	18097	68,86	0,38	0,47
0,2	23379	56,25	0,5	0,26
0,3	23294	51,6	0,55	0,23
0,4	22989	49,1	0,58	0,19
0,5	23747	47,83	0,7	0,26

4.3. Conclusions

The achieved results are the following:

- a comparison of V-DEVS software prototype and commercial system Flexsim grounding on the realized model of automated manufacturing system is made;
- the performance measurements of the simulation system prototype are obtained;
- the statistical simulation results of the automated manufacturing system are obtained.

The main conclusions are the following:

- it is possible to use the V-DEVS formalism for the development of practical interactive simulation systems, which is proved by the experimental results of the automated manufacturing system model developed with the simulation software prototype;
- interactive visual simulation system that is based on the V-DEVS formalism can be effectively applied and can compete with commercial software systems for the simulation of complex dynamic systems;

- despite a simple conceptual architecture of the hierarchical algorithm, it has a noticeably lesser performance than the improved priority queue algorithm. From this it follows that as the count of model elements and their relations grows, the V-DEVS based priority queue simulation becomes more suitable than the hierarchical algorithm.

MAIN FINDINGS AND CONCLUSIONS

The main goal of the thesis was to develop an integrated approach to simulation and visualization of discrete event and continuous systems on the basis of comparative analysis of simulation and visualization integration and identification of unresolved tasks, to implement the developed methods in the software system and to evaluate the realized system.

In order to achieve the goal, the following tasks have been solved:

- research of simulation and visualization approaches, methods and systems, which allowed to identify the unresolved tasks in their development and integration, as well as to define general requirements to an interactive visual simulation environment;
- development of an integrated visual simulation approach and methods for the elimination of the previously identified problems;
- development and practical implementation of the discrete event and continuous systems interactive visual simulation architecture on the basis of the proposed approach;
- experimental verification of the developed system for estimating possibilities for its practical use.

The development of the thesis made it possible to achieve the following most important and novel theoretical results:

- the integration necessity of discrete event and continuous systems simulation and visualization is substantiated;
- V-DEVS formalism for visual interactive modelling of discrete event and continuous systems is developed;
- a concept of visualization frame for defining an integrated context in the integration of visual simulation with the framework of modelling and simulation is introduced;
- a combined approach to integration and synchronization of quantized state system simulation and visualization is proposed;
- a model-driven approach to V-DEVS modelling is developed.

The following practical results are achieved:

- architecture, methodology and algorithms for a practical implementation of V-DEVS formalism are developed that differ from the traditional simulation systems in a unified treatment of modelling, simulation, visualization and interactive elements by unifying and simplifying the development and use of simulation models;
- a prototype of V-DEVS visual interactive simulation system is implemented, and the possibilities of this system are tested and presented.

The development of the V-DEVS simulation environment prototype and its evaluation allows drawing the following conclusions:

- the proposed V-DEVS visualization conveyor enables one to adapt both scene graph and classical visualization pipeline systems for simulation tasks by unifying those architectures into a unified system;
- the developed priority queue V-DEVS simulator algorithm has a better efficiency than the hierarchical V-DEVS simulator that is implemented on the basis of classical DEVS simulation algorithm, thus the priority queue algorithm is more suitable for solving simulation tasks of complex systems;
- experimental verification of V-DEVS system prototype confirms the usage efficiency of quantized state interpolator for visualization purposes by simulation of quantized state continuous systems;
- the model-driven approach improves and simplifies the process of simulation models development and verification.

Main possible directions for future research are:

- adaptation of V-DEVS formalism for modelling dynamic structure systems that were useful for the implementation of direct execution V-DEVS visualization conveyor;
- implementation of distributed and parallel data processing principles for support of V-DEVS based systems in multi-processor and/or web environments;
- support of a multi-user mode in an interactive simulation environment allowing different access rights for different user groups and synchronization between users;
- research of performance improvement possibilities of V-DEVS simulator by simulation of continuous systems;
- research of quantized states V-DEVS simulation stability in simulation of nonlinear continuous systems;
- V-DEVS integration with virtual / mixed reality technologies.

LIST OF REFERENCES

[1] *Discrete-Event System Simulation*, Banks J., Carson J.S., L. N.B., Nicol D.M. - Upper Saddle River, NJ: Pearson Prentice Hall, 2005. - 4th ed. - 624 p.

[2] Bell P., O'Keefe R. *Visual interactive simulation - history, recent developments, and major issues*// Simulation. - 1987. - No. 49. - Vol. 3. - 109–116 p.

[3] Blundell B.G. *An Introduction to Computer Graphics and Creative 3-D Enviroments*. - London: Springer Verlag, 2008. - 501 p.

[4] Borenstein D. *A visual interactive multicriteria decision analysis model for FMS design*// The International Journal of Advanced Manufacturing Technology. - 1998. - No. 11. - Vol. 14. - 848–857 p.

[5] Cellier F.E., Kofman E. *Continuous System Simulation*. - New York: Springer, 2006. - 643 p.

[6] Choi B.K., Han K., Park T. *Object-oriented graphical modeling of FMSs//* International Journal of Flexible Manufacturing Systems. - 1996. - Vol. 8. - 159–182 p.

[7] Choi B.K., Park B.C., Park J.H. *A formal model conversion approach to developing a DEVS-based factory simulator//* Simulation. - 2003. - No. 8. - Vol. 79. - 440–461 p.

[8] Cloud D., Rainer L. *Applied Modelling & Simulation: An Integrated Approach to Development and Operation*. - New York: AIAA American Institute of Aeronautics and Astronautics, 1998. - 736 p.

[9] Donald D.L. *A tutorial on ergonomic and process modeling using QUEST and IGRIP// WSC '98: Proceedings of the 30th conference on Winter simulation*. - Los Alamitos, CA, USA: IEEE Computer Society Press, 1998. - 297–302 p.

[10] Fishwick P.A. *A visual object-oriented multimodeling design for physical modeling*. - Techn. rep., ACM Transactions on Modeling and Computer Simulation, 1997. - 22 p.

[11] Frankel D.S. *Model Driven Architecture: Applying MDA to Enterprise Computing*. - Indianapolis: Wiley Publishing, 2003. - 352 p.

[12] Hwang M.H., Choi B.K. *GK-DEVS: Geometric and kinematic DEVS formalism for simulation modeling of 3-dimensional multi-component systems//* Transactions of the Society for Modeling and Simulation International. - 2001. - No. 3. - Vol. 18. - 159–173 p.

[13] Jain S. *Simulation in the next millenium// Proceedings of the 1999 Winter Simulation Conference*. - Phoenix, Australia, AZ, 1999. - December 5-8. - 1478–1484 p.

[14] Kofman E. *Discrete Event Based Simulation and Control of Continuous Systems*. - PhD thesis, Universidad Nacional de Rosario, Argentina, 2003. - 167 p.

[15] Lektauers A. *A mixed reality framework for visualization and execution of DEVS-based simulation models// Proceedings of the* 19^{th} *European Conference on Modelling and Simulation "Simulation in Wider Europe", ECMS2005*. - Riga: Publishing House of Riga Technical University, 2005. - June 1-4. - 271–276 p.

[16] Lektauers A., Merkuryev Y. *3D visual framework for modelling and simulation of supply chain systems// Scientific Proceedings of the eLOGMAR-M Project*. - Riga, 2006. - 141–150 p.

[17] Marshall R., Kempf J., Dyer S., Yen C. *Visualization Methods and Simulation Steering for a 3D Turbulence Model of Lake Erie//* Riesenfeld R., Sequin C. (eds.): *Computer Graphics: 1990 Symposium on Interactive 3D Graphics*, vol. 24. - 1990. - 89–97 p.

[18] *Sistēmu imitācijas modelēšanas tehnoloģija*, Merkurjevs J., Merkurjeva G., Pečerska J., Tolujevs J. - Rīga: Rīgas Tehniskās universitātes Datorzinātnes un informācijas tehnoloģijas fakultātes Informācijas tehnoloģijas institūts, 2008. - 120 p.

[19] Praehofer H. *System Theoretic Foundation for Combined Discrete-Continuous System Simulation*. - PhD thesis, Department of Systems Theory, University of Linz, Austria, 1991. - 173 p.

[20] Risco-Martín J.L., Mittal S., Zeigler B.P., de la Cruz J.M. *From UML state charts to DEVS state machines using XML// Proceedings of the Workshop on Multi-Paradigm Modeling: Concepts and Tools*, vol. 1 of *BME-DAAI Tech. Rep. Series*. - Budapest Univ. of Tech. and Economics Dep. Automation and Applied Informatics, 2007. - 35–48 p.

[21] Rohrer M.W. *Seeing is believing: the importance of visualization in manufacturing simulation// WSC '00: Proceedings of the 32nd conference on Winter simulation*, vol. 2. - San Diego, CA, USA: Society for Computer Simulation International, 2000. - 1211–1216 p.

[22] Schroeder W., Martin K., Lorensen B. *The Visualization Toolkit: An Object-Oriented Approach to 3D Graphics*. - Upper Saddle River, N.J.: Pearson Education, Inc., 2004. - 3rd ed. - 474 p.

[23] Wagner P., Freitas C., Wagner F. *A new paradigm for visual interactive modeling and simulation// 8th European Simulation Symposium - ESS'96*, vol. 1. - Genova, Italy: Society for Computer Simulation, 1996. - October 24-26. - 142–145 p.

[24] Zeigler B.P. *Theory of Modelling and Simulation.* - New York: John Wiley & Sons, 1976. - 435 p.

[25] Zeigler B.P. *Multifacetted Modelling and Discrete Event Simulation.* - London: Academic Press, 1984. - 372 p.

[26] Zeigler B.P., Praehofer H., Kim T.G. *Theory of Modeling and Simulation: Integrating Discrete Event and Continuous Complex Dynamic Systems.* - San Diego, California: Academic Press, 2000. - 2nd ed. - 510 p.

[27] Zhang J. *Visualization for Information Retrieval*. The Informational Retrieval Series. - Berlin Heidelberg: Springer-Verlag, 2008. - 292 p.

[28] Zinoviev D. *Mapping DEVS models onto UML models// Proceedings of the 2005 DEVS Integrative M&S Symposium*. - San Diego, CA, 2005. - April. - 101–106 p.

[29] Бенкович Е., Колесов Ю., Сениченков Ю. *Практическое моделирование динамических систем*. - Санкт-Петербург: БХВ-Петербург, 2002. - 464 стр.

[30] Советов Б.Я., Яковлев С.А. *Моделирование систем*. - Москва: Высшая школа, 2001. - 343 стр.

[31] Бусленко, Н.П. *Моделирование сложных систем*. - Москва: Наука, 1978. - 400 стр.

[32] Растригин, Л.А. *Адаптация сложных систем: методы и приложения*. - Рига: Зинатне, 1981. - 375 стр.