Journal of the Serbian Society for Computational Mechanics / Vol. 3 / No. 1, 2009 / pp. 250-272
UDC: 626.82:519.876.5

Discrete Events Simulation Model Applied to Large-Scale Hydro-Systems

N. Milivojević[1*], N. Grujović[2], B. Stojanović[3], D. Divac[4], V. Milivojević[5]

Institute for Development of Water Resources "Jaroslav Černi", 80 Jaroslava Černog St., 11226 Beli Potok, Serbia; e-mail: [1]nikola.milivojevic@gmail.com, [4]ddivac@eunet.rs, [5]vladimir.milivojevic@gmail.com
[2]Faculty of Mechanical Engineering, University of Kragujevac, 6 Sestre Janjić St., 34000 Kragujevac, Serbia; e-mail: gruja@kg.ac.rs
[3]Faculty of Science, University of Kragujevac, 12 Radoja Domanovića St., 34000 Kragujevac, Serbia; e-mail: bobi@kg.ac.rs

**Corresponding author*

Abstract

This paper presents the principles of simulation model developed with DEVS formalism for large-scale hydro-systems management and planning. This simulation model, being DEVS-based, is natively object-oriented, open for communication with other models and is suitable for parallel execution in multiprocessor and distributed environments. It is designed as a part of the hydro-information system, whose strategic objective is to create environment for the optimum management of water resources and to address and resolve the existing and potential conflicts of interest in the region, related to multi-purpose use of water, and the misalignment of interests of the various stakeholders in the river basin. The functional objective of the simulation model is to support water management decision-making (i.e. to aid users in the assessment of the consequences of various management scenarios and to support planning within various hydrologic, climatic, economic, regulatory and political constraints). The combined use of the rainfall/runoff model and DEVS-based simulation for system elements represents a new approach when it comes to water and hydropower management. The model and simulation core make a solid base for the application of optimization algorithms, in order to provide more automated decision support system. A case study of River Drina Basin simulation model, named "Drina" Hydro-Information System is also presented.

Keywords: Simulation, modeling, discrete event, hydro-system

1. Introduction

Hydro-system modeling is a very important element of the contemporary water resources management (Divac et al. 2009). The roots of the hydro-system modeling are in more then hundred years old hydrological models. Some of the first models were used were the model used for forecasting of peak flow prediction (Mulvaney 1850), the storage volume analysis model (Rippl 1883) and the unit hydrograph method for transformation of rainfall into runoff in the catchment (Sherman 1932). New analytical and numerical models had appeared over the years, but the major progress was achieved after the computers were invented. Present development in the domain of modeling and simulation in the hydro-information technology was unimaginable just 50 years ago.

Today, the progress in the field of hydro information systems is closely related to modeling and simulation (Divac et al. 1999, Grujović and Divac 2004). Although the disciplines like, for example, experimental hydrology are very important, new hypotheses and theories are created only in conjunction with modeling and simulation. Modeling and simulation in hydro-information technology became the testing platforms for new theoretical assumptions and hypotheses that are helping the further understanding of the processes, phenomena and methods of their interaction.

Mathematical models relevant for simulation in the application of hydro-information system can be diverse in terms of all modeling aspects, both in regard to concept and applicable simulation methods. Various hydrological models divisions were presented in the specialized literature. Usual division is based on opposed concepts, such as deterministic and stochastic, lumped and distributed, constant and variable time-wise, etc. (Singh 1995, Abbott and Refsgaard 1996, Refsgaard and Knudsen 1996). General issue in regard to design of simulation software packages is what information set should the packages provide to the user for him/her to make correct decisions or conclusions in the shortest possible time. Principally, the most general division of models is on the models oriented towards the use of measured states of the real systems and on physically-based models, focused on mathematical modeling of the simulated process. These two model types represent two extremes. A broad range of “hybrid” solutions, dependent upon available data, definition of the process itself, application of the model and similar, lies between these two extremes.

A fine division inside the possible set of models is not possible, because, practically, all levels of application of final solutions are allowed (completely oriented towards the use of the measured states of the real process and completely physically-based models, oriented towards the mathematic modeling of the process) without predefined limitations. However, a somewhat rougher division is often made, on neuron networks, hybrid models, numerical models with statistics elements and deterministic numeric models.

Neuron networks are based on calibration of parameter of correlation between input and output sets. Hybrid models, in addition to the correlations that neuron networks are based on, introduce certain physically-based rules, supplemented by the knowledge generated on the basis of the observed data. Numerical models with statistical elements represent the group of numerical models which are, at the certain level, supplemented with the statistical processing of measured values that are not describable by deterministic equations. Finally, deterministic mathematical models assume good knowledge of the process involved and the capacity to describe it in detail by the systems of equation.

Several fundamental decisions are to be made on the basis of model spatial and time scales. In terms of time characteristics, the model can be event-based (the unit hydrograph method), or it can be formed for the purpose of simulation of processes continuous in time. Also, the objective of the model can be the description of a certain steady state and tracking of the distribution of state variables in the given system, and it can refer exactly to the time dynamics of the process on individual system locations. If the simulated process lasts for more than one year, seasonal parameters should be often taken into consideration. Different answers to these questions result in different concepts of the hydrological models.

There are also many options in the spatial domain, ranging from the simplest centralized models, which are analyzing processes inside the parts of discretized space (elements), to the complex 1D, 2D and 3D models, which are created by merging an unlimited number of elementary units.

Finally, the most important thing to do before the start of development of a hydrologic model is to comprehend the model purpose and the conditions of its use. The model will sometimes work with input values, which are in the range of data used for its calibration, and sometimes the practice will result in significantly different input data. Since certain models will be used for designing and some for system operational management in real time, the concepts of such models will significantly differ.

The modern approach to water resources management imposes the integrated approach to problem solution (European Parliament, 2000), which is bringing the spotlight on the problem of formation of complex catchment models. The main challenge in formation of complex basin models is the generation of interaction between the models treating only individual phenomena in the basin area. Development of a single model that implicitly treats all phenomena in the catchment area is not optimum and, maybe, even not realistic. Above all, the existing models are not used in that case. Huge efforts were invested in development of these models, which are stable, reliable and tested in detail in real operation. The only realistic approach to modeling for the purpose of integrated basin management is the linking of different models and their coupled execution with a complete interaction during the simulation. This form of interaction, which implies the exchange of information and a large amount of data in real time, can be achieved in several ways. One of the generally accepted approaches to this is the implementation of mechanisms such as OpenMI (OpenMI Document Series 2007), used to perform the transformation from modeling in the single problem domain to simultaneous modeling of several problems that are coupled, but not belonging to the same domain. Of course, this mechanism is not limited only to problems related to water flow but it can also involve economic, social and similar models. OpenMI itself defines the interfaces providing for time-dependent models to perform data exchange during the simulation. Each model implementing OpenMI standard in the form of interface can be included in the simulation of a complex system, whereby the models can exchange required data in each simulation step. In this way, the experts can test numerous options with different combinations of temporal and spatial dimensions of the model, as well as the level of detail.

The conclusion is that there are certain defined standards providing for creation of complex basin models which were already tested and adopted in practice. However, these standards, even though they do not limit the models to water resources domain, are specialized to a certain extant and it is not possible to treat them at the level of general systems, model specifications and simulations, as in the case of time-discrete models, continuous models or event-based simulations. Many models based on these most general principles must therefore be introduced subsequently into the system of standards such as OpenMI, what requires new experts and time. This is often not justified if models of individual basin parts are not overly complex. Therefore, the use of generally accepted modeling and simulation platforms can be often considered as an alternative. Due to the outstanding heterogeneity of subject models in terms of temporal and spatial discretization, it is necessary to overcome the limitations of continuous and time-discrete model specification, what is pointing the way to the specification of discrete event models.

After many years of development of simulation tools and progress in information technology, the compatibility of development of these systems with the object-oriented paradigm came into light. Therefore, special attention was paid to the use of the object-oriented paradigm in development and implementation of models and simulation platforms. Special importance in development of real simulation solutions in object-oriented languages of general purpose has the fact that this paradigm is also used in modeling and in programming, making these two processes more efficient.

Since the hydrologic modeling involves diverse objects, which can be physically-based by nature, as well as defined by the means of abstract mathematical descriptions, one can come across the completely different modeling principles. The expected final model is most often an integral system, comprised of heterogeneous models operating as a whole.

Present paper will present the basics of the simulation model developed on the basis of DEVS formalism for the purpose of management and planning of large-scale hydro-systems. This simulation model, being DEVS-based, is natively object-oriented, open for communication with other models and is suitable for parallel execution in multiprocessor and distributed environments. The model was applied on real problems, which is also presented here.

2. Simulation of a system with discrete events

Several approaches can be used for the analysis of simulation models. The most general division of model definitions is on informal description and formal description of the model.

Informal model description provides the basic notions about the model and, although the main aspiration is the completeness and preciseness of the model that is not achieved in most cases. Division on objects, descriptive variables and rules of interaction between the objects is performed in formation of the informal description, exactly to eliminate the mentioned deficiencies (Zeigler et al. 2000).

Informal model description can be prepared rather fast and easy. However, very often it is not consistent and clear, especially in case of complex models. The anomalies of informal description can be often described as an incomplete model description, an inconsistent model description or an unclear model description.

If the model does not contain all situations that can occur, then the description is incomplete. If the model description uses two or more rules for the same situation, application of which results in contradictory actions, than the description is inconsistent. Finally, if one situation requires two or more actions, and their order is not defined that the model description is not clear. On the other hand, modern modeling methodology is largely based on certain conventions, called formalisms. Formalism specifies the class of observed objects in an unambiguous and general manner, using the conventions and rules (Delaney and Vaccari 1989).

Formal model description should provide higher accuracy and completeness in model description, and sometimes it makes possible the formalization of testing of incompleteness, inconsistency and vagueness. Nevertheless, the most important fact here is that the introduction of formalism in the methodology provides for focusing of attention on these characteristics of the object that are of the greatest importance for the research; that is, to use abstractions.

In contrast to analytical models treating the overall system behavior directly, simulation models collect data on changes of the state of the system and the output, by focusing on the behavior of individual system components (Zeigler et al. 2000). The importance of simulation models originates from the fact that only a small number of complex real systems can be described appropriately through analytical equations. The application of simulation modeling can not result in the solution in analytical form wherein dependent variables are the functions of independent variables, but the problem solutions has to be the result of experiments performed on the model.

Two basic divisions of the simulation models can be distinguished: the first one, according to the type of variables in the model, and the second one, according to the way the state in the model is changing in time.

According to the first division, simulation models can be deterministic and stochastic, and according to the second one, they can be continuous and discrete. Three main system specifications originate from the second division (Zeigler et al. 2000):

- Differential Equation System Specification - DESS
- Discrete-Time System Specification – DTSS and
- Discrete-Event System Specification – DEVS.

2.1. Formal description of the system with discrete events

Set theory (Enderton 1977) provides for construction of formalisms that are used for describing the model objects. Each class of the object can be presented by the respective formalism that is describing its parameters and limitations. In order to define an individual object of the class within the certain formalism, formalism parameters are assigned with values meeting the limitations.

One object of the class assigned with the formalism is defined by assigning the specific values to the parameters which meet the limitations. Formalism structure refers to both parameters and limitations. Object classes are usually related in a way that facilitates formalization of the links as replications from one class to another. Three types of replications are particularly interesting: abstraction, association and specification.

Abstraction is the conformation that includes controlled details inclusion during the model description. This process sets aside certain object related details that is, important features are separated from unimportant ones, for the purpose of identifying the substance of the object. Therefore, the abstraction can be characterized as the replication of the object of one class to another, less complex class. The significance of the abstraction lies in the fact that a large number of original objects can be presented by the same target abstractions.

Association is the type of replication from lower to the higher level in the hierarchy of system specification. Inverse replication is called realization or implementation.

Subclasses can be defined for the given class of objects by introduction of the new formalism for each subclass. Subclass objects can be, after this, used within the new formalisms. However, to establish relations between the objects of different related subclasses, it is necessary to define the replication which translates one special formalism into another, a more general one. If such replication exists, all concepts and actions applicable to objects defined in the general formalism are also applicable to objects defined in the special formalism.

As mentioned above, mathematical models relevant for simulation of the problems in the hydro-information systems can be very diverse in terms of all modeling aspects, both in concept and the applicable simulation methods. But, since these processes are by default based on natural phenomena, their continuous nature has to be bear in mind. On the other hand, human factor acts in time in a discrete manner and its actions are often based on certain events. Simulation of continuous system by utilization of some of the classic methods can be represented by equations:

$$x\left(t_{k+1}\right)=f\left(x\left(t_k\right),t_k\right) \tag{1}$$

where the time step $t_{k+1}-t_k$ can be both constant and variable, and function f can be implicit and explicit. Programs that perform simulations of these models contain an iterative code which is processing model states that are changing in time at exactly set steps that is, these models are discrete in time.

Regardless of all the similarities, there is an apparent difference between these systems and the systems with discrete events. Simulation of discrete events is the method of system modeling where discrete changes of the state in the system or its environment are discontinuous in time. The notion of a “discrete event” is often related to the well-known formalisms such as Petri nets, event graphs, state charts, etc. Unfortunately, none of the methods provides the general description of the notion of discrete event in full. These graphic languages are limited only to systems with a certain number of possible combinations of system states what in case of simulation of a continuous system does not represent a sufficiently general approach. Discrete Event System Specification – DEVS is an existing specification which is sufficiently general to be used for solving this type of problems.

DEVS formalism was developed by Bernard Zeigler in mid-seventies of the last century. So far, it was widely used for simulation of computer systems and networks, and as a method for simulation of continuous physical systems it is still not known well. DEVS provides for presentation of all systems whose input/output behavior can be described by sequences of events under the condition that systems states have a finite number of changes in any time interval.

An event is a momentary change in a certain part of the system and it can be described by a value and the point in time it occurred in. The value can be a number, array, word or, in general case, an element of an arbitrary set. The model of a system with discrete events processes the sequences of events by assuming that the value of input is ϕ (or without an event) in time, except for the instants when the events are taking place, as shown in Figure 1.

Fig. 1. Input/output behavior of the system

Since the model specification defines the set of rules, instructions, parameters and similar, it is necessary to provide an “agent” capable of behavior harmonized with the formalism and, thus, capable of generate the model “behavior”. These agents are called simulators. Therefore, a simulator is an agent capable of execution of model instructions and generation of its behavior (Zeigler et al. 2000). This separation of the modeling process (resulting in model specification) and the simulation process (defining the simulator capable of execution of model instructions) creates many advantages in comparison to the classical monolith approach to development. Firstly, a formal model can be simulated by several different simulators what creates a better interoperability. Also, it is possible with certain types of formalisms to clearly define simulation algorithms and determine with success whether they are correct, accurate and suitable to the subject model.

Since the formal descriptions of the DEVS atomic model and the coupled DEVS model have been defined, it is necessary to analyze their simulators. These simulators are realized in the concrete program environment in later stages, in line with object-oriented concept.

2.2. DEVS Model specification

DEVS model processes input series of events and, in line with the given series and own initial state, produces a series of events on its outputs. Input/output characteristic of DEVS model is described in the manner which is quite common in the automata theory (Hopcroft et al. 2000).

Formal description of DEVS atomic model is defined as structure

$$M = \left(X, Y, S, \delta_{\text{int}}, \delta_{ext}, \lambda, ta\right) \tag{2}$$

where:

- X - is the set of input events,
- Y - is the set of output events,
- S - is the set of the states of the system and
- δ_{int} - is the internal transition function,
- δ_{ext} - is the external transition function,
- λ - is the output function,
- ta - is the set of positive real numbers.

All possible states $s\left(s \in S\right)$ have the associated time advance calculated by the time advance function $ta\left(s\right)\left(ta\left(s\right): S \rightarrow \mathbb{R}_0^+\right)$. The result of the time advance function is a non-negative real number describing for how long the system will stay in unchanged state in absence of an external events.

If the state takes the value s_1 at time t_1, after $ta\left(s_1\right)$ time units (i.e. at the time $ta\left(s_1\right) + t_1$), the system will execute an internal transition to the new state s_2. The new state is as follows $s_2 = \delta_{\text{int}}\left(s_1\right)$. Function δ_{int} ($\delta_{\text{int}} : S \rightarrow S$) is called the internal transition function.

When the system transits from the state s_1 to the state s_2, an output event with the value $y_1 = \lambda\left(s_1\right)$ is generated. The function $\lambda\left(\lambda : S \rightarrow Y\right)$ is called the output function. Functions ta, δ_{int} and λ define the autonomous behavior of the model.

If at the certain time an input event occurs, system state is changed momentarily. The new state does not depend only on input events, but also on the previous state, as well as on the time elapsed from the previous transition. If the system transits to the state s_3 at the time t_3 and the input event occurs at the time $t_3 + e$, with the value x_1, the new state is $s_4 = \delta_{ext}\left(s_3, e, x_1\right)$, wherein $ta\left(s_3\right) > e$ was assumed. In this case, it can be stated that an external transition has occurred. Function $\delta_{ext}\left(\delta_{ext} : S \times \mathbb{R}_0^+ \times X \rightarrow S\right)$ is called the external transition function. Output events are not generated during an external transition.

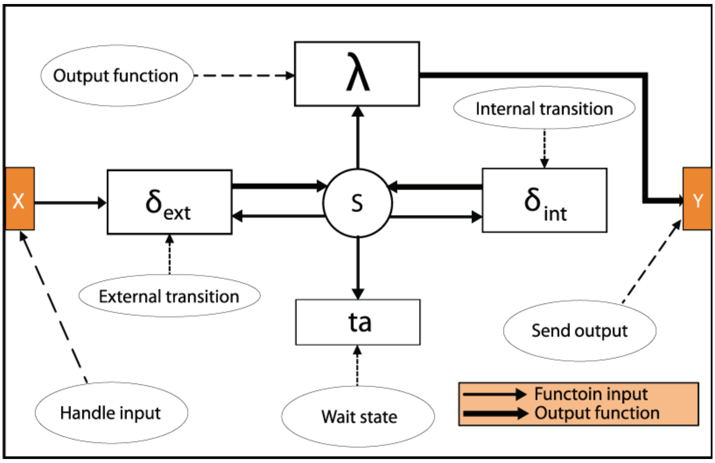

Fig. 2. Schematic illustration of functioning of DEVS simulator

2.3. DEVS coupled models specification

DEVS is a general formalism and it can be used to describe very complex systems. However, complex system description only by the transition and the time advance function might be too complex. Difficulties occur because only several functions are used to predict and describe all possible cases encountered during the simulation. Naturally, complex systems can be treated as a large number of coupled simple elements. After the coupling, output elements of one subsystem become the input events of another subsystem it is coupled with. Theoretic assumptions ensure that the coupled system will in a given environment completely behave as an atomic model; hence, complex models can be created hierarchically.

There are two different ways to couple two DEVS models. The first one is more general and it is based on the introduction of the translation function between the coupled models. The second way introduces new notions – the input-output ports.

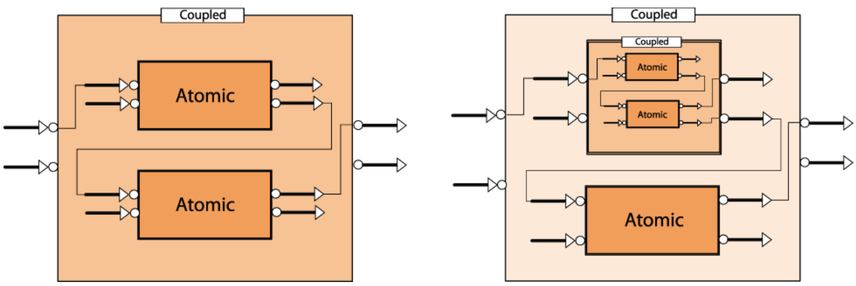

Fig. 3. Hierarchical coupling

Coupling of a system with discrete events by the means of ports is more often used for simulation of continuous and hybrid systems and it is simpler for implementation. DEVS formalism of a model with ports can be described in the following form:

$$M = \left(X, Y, S, \delta_{\text{int}}, \delta_{ext}, \lambda, ta\right) \quad (3)$$

where:

- $X = \left\{(p,v) \middle| p \in InputPorts, v \in X_p\right\}$ - set of input ports and values of input events,
- $Y = \left\{(p,v) \middle| p \in OutputPorts, v \in Y_p\right\}$ - set of output ports and values of output events,
- S - set of sequential states,
- $\delta_{ext} : Q \times X \to S$ - external transition function,
- $\delta_{\text{int}} : S \to S$ - internal transition function,
- $\lambda : S \to Y$ - function for generation of output events,
- $ta : S \to \mathbb{R}_0^+$ - time advance function, and
- $Q : \left\{(s,e) \middle| s \in S, 0 \le e \le ta(s)\right\}$ - set of total states.

2.4. Simulators of DEVS models

In order to implement the simulation platform of DEVS atomic and coupled models on a computer, it is necessary to define their simulators and the hierarchy of the simulation environment itself. These simulators can be developed in a programming environment in line with the object-oriented concept.

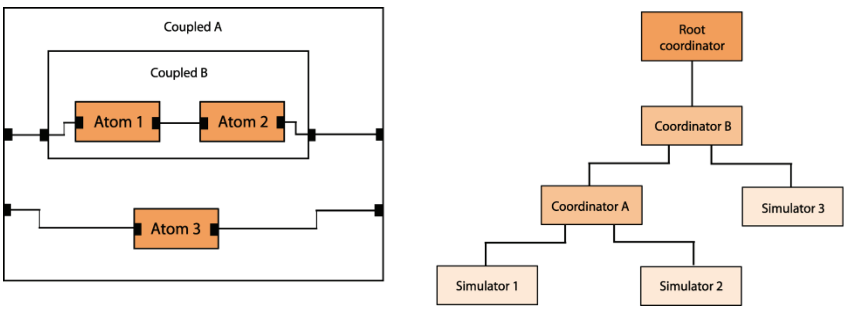

Fig. 4. Basic Structure of DEVS simulation Environment

Extendable generic basis of a DEVS simulation environment is founded on the following concepts:

- Basic structure in each formalism is a class of atomic models,
- Each atomic model must possess the corresponding simulator,
- Structure of the coupled model is represented by the class of coupled models,
- Each class of the coupled model possesses its own corresponding simulator, also called the coordinator,
- Simulators are based on the generic protocol that makes possible to use the messages to perform communication required for simulation,

- Assigning of simulator to the atomic models, as well as the coordinators to the coupled models and simulators to their coupled atomic models is performed hierarchically until the base of the hierarchical tree is reached.

Figure 4 shows the mapping between the hierarchical model and its abstract simulator. Atomic simulators are attached to the ending elements of the tree comprising the atomic model, the coordinators are attached to the coupled models and internal tree branches, while the root coordinator, in charge of the coordination of the simulation process, is on the top.

2.5. Simulator of atomic DEVS model

Basic DEVS simulator uses two time variables t_l and t_n. The first variable keeps the simulation time of the occurrence of the last event and the second variable is the time of occurrence of the following event. The following expression results from the definition of the time advance function:

$$t_n = t_l + t_a(s) \tag{4}$$

Also, if t is global simulation time, the simulator can calculate the time elapsed from the last event:

$$e = t - t_l \tag{5}$$

as well as the time up to the following event

$$\sigma = t_n - t = t_a(s) - e \tag{6}$$

The time of the following event t_n is forwarded to the upper simulator of the in order to ensure a correct synchronization of the course of the simulation.

As seen in the given algorithm, a correct initialization requires that the simulator has received the message (i,t) at the start of each initiation of the simulation. When the simulator has received the message, time $t_l = t - e$ is initialized. The time of occurrence of the next event is computed by adding the value of the time advance function to the time of occurrence of the last event. The resulting time t_n is forwarded to the simulator, which is then aware of the time of the next internal event and on the basis of that it performs synchronization.

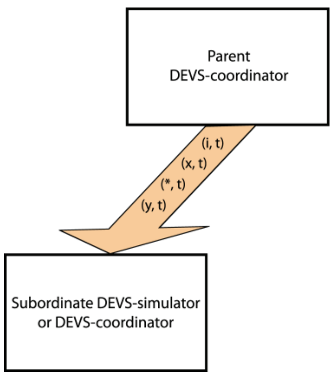

Fig. 5. Protocol of the DEVS simulator

Synchronization message $(*,t)$ requires from the simulator to calculate output values and execute the internal transition function. Output values are sent to the parent simulator in the form of output messages (y,t). Finally, the time of the last event t_n is set to the value of global simulation time and the time of the next event as the global simulation time increased by time advance function value $t_a(s)$.

Input message (x,t) informs the simulator on the occurrence of the event x at the simulator input at time t. This event causes the execution of the external transition function, taking into consideration x and elapsed time e. Same as for the internal transition, the time of the last event t_n is set to the value of the global simulation time and the time of the next event as the global simulation time, increased by the time advance function value $t_a(s)$.

2.6. Simulator of the coupled DEVS model

In the case of coupled models, individual simulators (which are assumed to perform correct model simulation) are responsible for each internal model. The coordinator, in charge of the coupled model, performs the synchronization of all simulators belonging to it, as well as the delivery of messages received at the coupled model input ports. To meet the condition of full consistency, coordinator plays the role of simulator in relation to its parent coordinator by implementing all protocols.

Inside, the coordinator implements algorithms for processing of the list of events in order to perform a correct synchronization of its components. Model simulators, which are coupled, are on the list that contains of the couples consisting of the simulator and the time of the next event. This list is sorted by times t_n, and if two or more simulators contain the same time, the choice is performed by the relevant function of the coupled model (most often called Select) which is used to solve the order of message processing. Time t_n of the first simulator (or coordinator) on the list is the time of occurrence of the following event for the entire coupled model. This minimum time:

$$t_n = \min\left\{t_{n_d} \middle| d \in D\right\} \tag{7}$$

is forwarded to the parent coordinator as the time of occurrence of the next event. Similarly, the time of occurrence of the last event is as follows:

$$t_l = \max\left\{t_{n_d} \middle| d \in D\right\} \tag{8}$$

the last event time on some of the associated simulators. Also, the coordinator is in charge of forwarding the input messages in the coupled model.

As mentioned, the coordinator uses the same message types as the simulator and implements the same processing mechanisms. The presented algorithm shows that the coordinator receives the messages for initialization (i,t), synchronizing messages $(*,t)$, as well as the external messages (x,t). Its self-generated messages are sent to its parent coordinator in the form (y,t) and the same messages (generated on their related simulators) are processed.

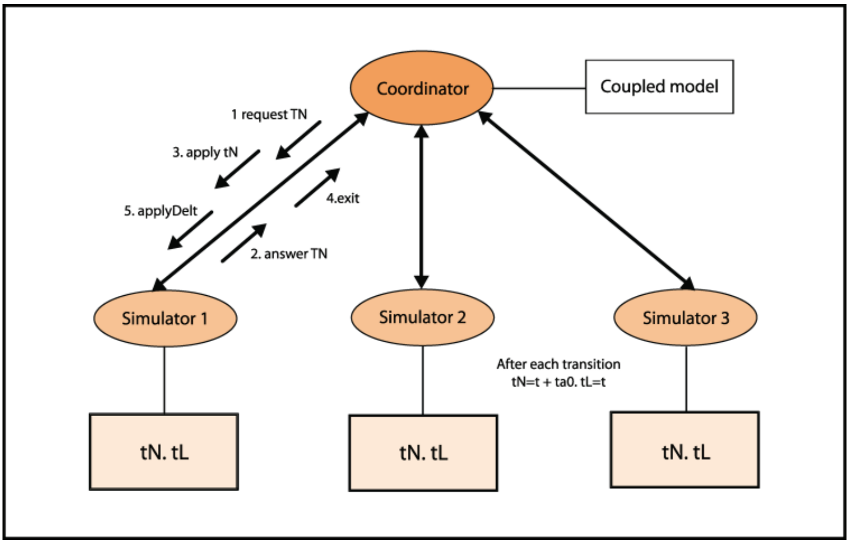

Fig. 6. Coordinator algorithm for processing of the events.

After the coordinator has received the message (i,t), the message is forwarded to the associated simulator. After all simulators have processed this message, the coordinator sets the resulting times t_l and t_n .

After the coordinator has received the message $(*,t)$, the message is forwarded to the first simulator on the list, which is then executing its transition function and, if needed, generates the output message (y,t). After the completion of the internal transition and processing of output messages (generated by the simulator), the next event time is set, as a minimum value of the set of times of all associate simulators.

After the coordinator has received the output message (y_d,t), created on the simulator listed as first, the coordinator firstly tests whether this message is leaving the coupled model or it is forwarded to the input of the associated simulator. In the first case, the coordinator forwards the message to its parent coordinator as (y_N,t). In the second case, the message (y_{d^*},t) is converted into the input message (x_{r^*},t) by $x_r = Z_{d^*,r}(y_{d^*})$, which appears on a respective simulator r as a new input message.

After the coordinator has received the input message (x,t) from its master coordinator, all associated simulators, affected by this message, are determined and input messages $(Z_{N,r}(x),t)$ are generated and forwarded to the given simulators.

The role of the root coordinator is to manage subordinate coordinators and thus perform the simulation in a synchronized manner. Firstly, coordinator are sent the message (i,t), which is initializing simulators, and then the synchronizing message $(*,t)$ is periodically generated with the time t_n and its task is to induce relevant simulators and, thus, execute the simulation until some of the interruption conditions is reached.

3. Platform for modeling and simulation of complex hydro-systems

As show in Divac et al. (2009), modeling and simulation play a very important role in development of systems for support in hydropower systems management. However, a large number of different problems to be solved make development of universal models difficult. These models would have to solve problems of different complexity, spatial and time discretization, and often with simultaneous observance of the interaction between essentially different domains (hydraulics, hydrology, ecology, economy, social effects and alike).

The conditions of utilization of hydropower potential are very complex because the associated storages are often also used for water supply, irrigation, fulfilling the needs for water of the industry and similar. Also, water is a scarce resource directly connected to living creatures and environment. Successful management of hydropower potential requires a modeling and simulation platform that, above all, would facilitate implementation of the model of water flow through the complex system objects and transformation of the hydropower potential into electricity on hydropower plants. Additionally, the platform must be extendable by different models which are about concerned with all phenomena associated with the utilization of water resources and directly related to exploitation of the hydropower potential. It is necessary to provide models of water supply, irrigation and similar, as well as to create conditions for platform extension by new models. In addition to all things mentioned, models developed on the basis of the platform for modeling and simulation for complex hydro-systems must be flexible and the parts of the complex models must be easily manipulated to define variant solution.

These requirements make the development and implementation of the platform for modeling and simulation of complex hydro-systems, because it has to include models different in terms of formalisms, degree of complexity, purpose and similar.

As mentioned above, DEVS is a general formalism which can describe highly complex system. Therefore, DEVS is a natural solution for the definition of the platform for modeling and simulation of complex hydro-systems. However, description of complex systems based on strict physical principles, in the seemingly rather simplified formalism, is often an overly complex procedure. Since DEVS formalism provides for treating of complex systems as a large number of coupled simple elements, the basic goal is the development of a library of elementary models that can be coupled so that more complex models can be created (if the storage and the associated power plant are coupled into one coupled system, then the output event of the storage, such as data on the headwater level, current balance, outlet to power plant and similar, becomes the input event for the power plant, based on which it calculates its own state variable, time and similar). Theoretical assumptions of the simulation platform ensure that the coupled system will behave as an atomic model so that complex models can be created hierarchically.

3.1. Examples of models with discrete events

For the purpose of understanding the application of the described models, it will be necessary to analyze certain elementary examples of formal DEVS model description applied on physical objects. In order to clarify the application of the DEVS simulation platform, three cases of idealized storages shall be analyzed: with a spillway and a constant inflow, with a spillway and with a variable inflow, and with a spillway with the variable crest elevation.

The first example presents the model setup for a quite simple storage with a linear characteristic of volume-level dependency, constant elevation of spillway threshold and a continuous inflow. The only discontinuity (event) in this case is the spilling. For the sake of simplicity, the inflow was expressed in dimensions if volume per unit length. Since the initial condition is an empty storage, then the application time is expressed as $t_a = H/q$. Therefore, the first following moment to perform model simulation is $t = t_a$ and then the change of the outflow will occur and it will be equal to the inflow and after that application time will be infinite.

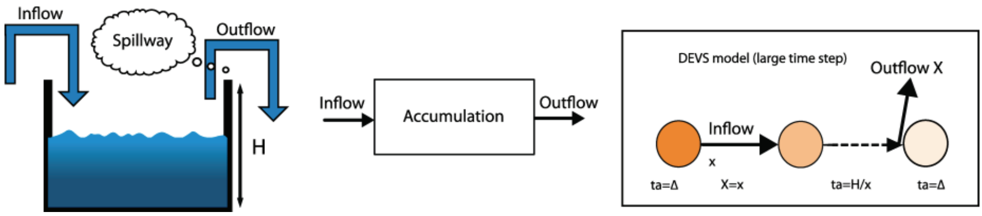

Fig. 7. Modeling of discrete events on the example of a continuous process with a limitation – an example of the storage with spillway

Previous example shows that the entire simulation was comprised of one calculation of the model at the time of model transition from the no spilling state to spilling state. This course of events is possible only if the inflow is constant.

As a somewhat more complex example, one can observe the option of inflow variation during the simulation period. The maximum simplification leads to the situation when the initial inflow q_0 at certain time has changed into q_1. The time of occurrence of this event is designated with t_e and it meets the condition $t_e < H/q_0$. The trajectory of system movement has one state more than in the previous example. System transition takes place at the time t_e when the inflow change occurs and that is changing the speed of change of the storage water level. Therefore, the new lifespan time is $t_a = h/q_1$, wherein $h = H - t_e \cdot q_0$. When spilling occurs at time $t = t_e + t_a$, the runoff will be equal to the new value of the inflow q_1.

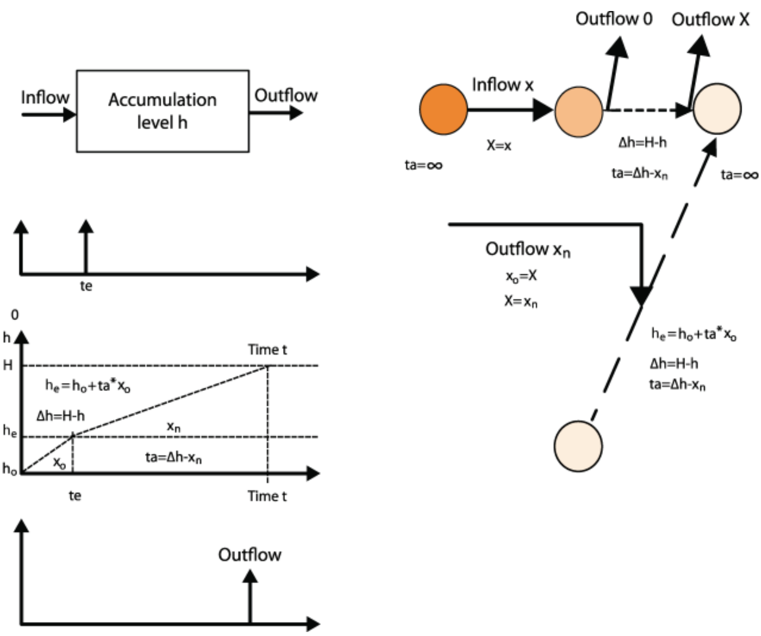

Fig. 8. Modeling of inflow change in DEVS simulation

Finally, one can analyze the case where the model state changes due to the control of certain parameters. For example, one can observe modeling of the spillway field control at the constant inflow. As in the previous two cases, the inflow was given in dimensions of volume per unit length. Initial inflow q_{in} is constant in time of simulation, so the application time in the first step is $t_a = H/q_{\text{in}}$. For the time t_e, where $t_e < H/q_0$ holds true, the elevation of the spillway threshold changes from the value H to value H_e. At that time model transition is performed into the one of the two possible states depending on the sign of the difference $h = H - H_e$. If the assigned spillway opening is such that the elevation of spillway is below the current water level in the storage, the spilling will occur, which means that the outflow was changed to the value q_{out}, and that the new lifespan time is $t_a = h/q_{out}$. In the second case, where $h > 0$, and the state remains unchanged, the lifespan time changes to $t_a = h/q_{in}$. In both cases, after the lifespan time has elapsed, the model will move to the final state defined by $t_a = \infty$, wherein $q_{out} = q_{in}$ and $H = H_e$.

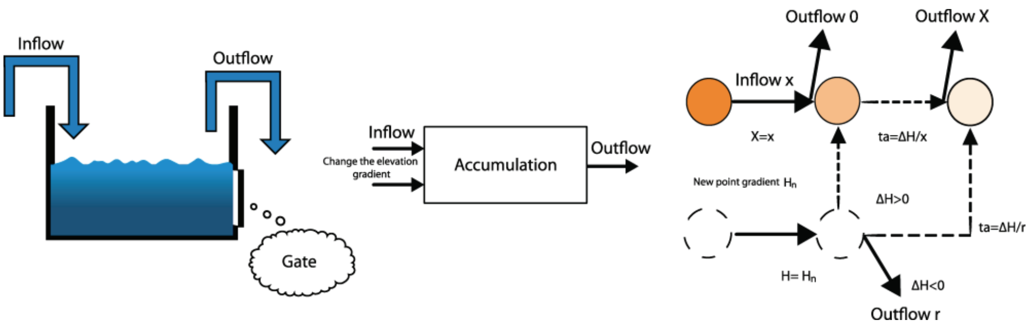

Fig. 9. Modeling of the spillway field opening by DEVS

Similarly, the behavior of other models in the system (power plants, users, flows, etc.) is defined, which creates the building elements for the construction of complex models of hydro-systems. Naturally, the analyzed storage model is simplified, but the model of a real object is not essentially different from the presented model. The main difference between the real object model and the presented model is the nonlinear characteristic of the water level-volume dependency for the storage, as well as in the associated phenomena of evaporation, leakage etc. that are included in the balance equation. The nonlinear dependency of the volume upon water level is solved by dividing the storage volume curve into a required number of sections (quantization), either according to water level or the volume. This is used to limit the lifespan time as dependency on the storage state and to ensure the balance throughout the simulation.

3.2. Structure of the library of models for simulation of hydro-systems

Mapping between the hierarchical models and their abstract simulators shown in Figure 4 implies the required structure of the library of models for simulation of hydro-system. Mapping implies that atomic simulators are attached to the ending elements of the trees that consists of atomic models, coordinators are attached to coupled models and internal tree branches, while the main coordinator in charge of coordination of the simulation process is on the top. Given structure provides for the identification of the main classes that the simulation environment is comprised of, which are related to the models and simulation entities. In addition to the classes of the atomic model, the simulator, complex models, the coordinator and the main coordinator, it is necessary to ensure the functioning of the protocols between the given entities; hence, one can refer to this as to the classes linked to ports, coupling and messages used for the communication.

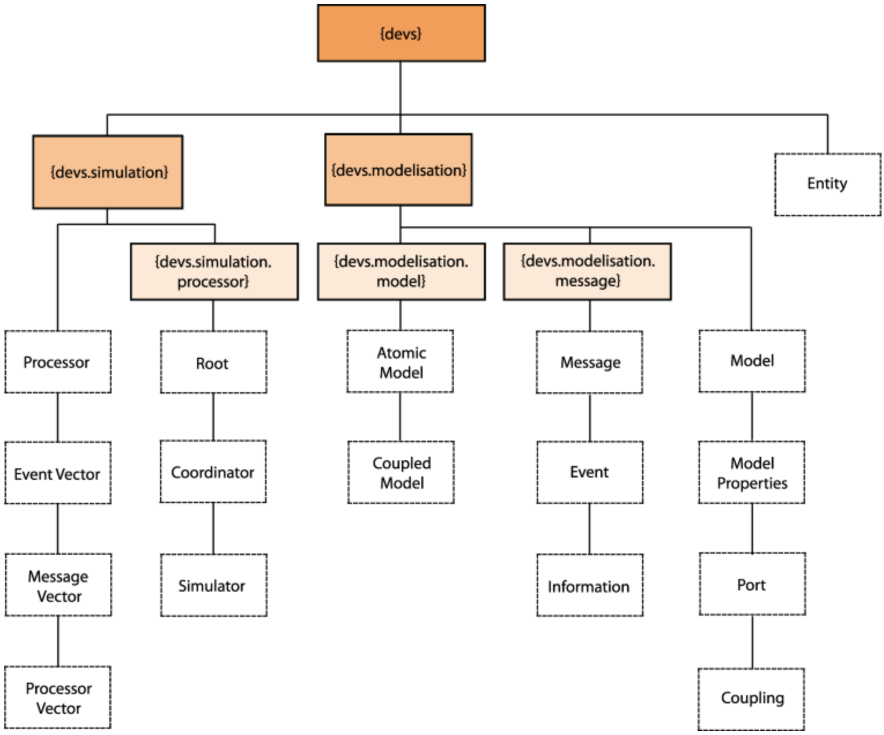

Fig. 10. Structure of DEVS environment according to name spaces

The basic elements required for the functioning of the DEVS simulation environment are shown in Figure 10. This structure has been defined according to "namespaces" and it represents the "organizational" structure of the environment where the classes have been grouped inside the certain namespaces. One can notice that the first division was performed according to the classes related to modeling and classes related to the simulation of the given models. These two groups are connected because one simulator is joined to each atomic model during the simulation, and one coordinator to each coupled model.

3.3. Objects in the model library

In order to design the library of models for simulation of complex hydro-systems, it is necessary to perform an analysis of real systems, as to comprehend the required model types, parameters, methods of their connections, operational rules and similar. By conforming to the mentioned spatial and functional complexity of such real hydro-systems, model spatial decomposition has been performed with introduction of different elements that can simulate various water flows (natural or artificial, uncontrolled or manageable, through the built-up, or through the possible future objects). This set of objects that are modeled individually represents the model library where the model coupling can be used to create complex models of hydro-systems of any size.

This library is made of models that describe natural processes of transformation of rainfall into runoff (sub-catchment), flow in the network of open channels (hydro-profile, open flow),

flow through the artificial structures (flow in closed conduits) and user objects (hydropower plant, pump station, reversible power plant and the user).

The model that describes the sub-catchment runoff is a spatial element determined by the upstream and downstream hydro-profiles and watersheds of the river network. If the downstream hydro-profile is inactive, the sub-catchment of that inactive hydro-profile is added to the first downstream storage or a flow-through hydro-profile. Merging of several sub-catchments results in the catchments that lead up to the respective active hydro-profile. The following forms of modeling are possible on the element of the sub-catchment-type: rainfall formation on catchment areas (water entry into the system), transformation of rainfall into the surface and sub-surface runoff in catchment areas and water loss in catchment areas (outflow of water from the system). A hydro-dynamic, physically based model was adopted (Simić et al. 2009).

Hydro-profile is the model element envisaged on all locations of built-up and planned dam profiles, water-measuring stations, water intakes for any type of use, points of water returning back to the system by the user, and confluences. Hydro-profile is located solely on the river (natural watercourse) and its existence determines the control profile of the associated sub-catchment. Hydro-profile in the model may exist in one of the following four states: as the storage hydro-profile, as the runoff hydro-profile, as the inactive hydro-profile or as the entry hydro-profile.

The most complex form of the hydro-profile form is the storage hydro-profile that, as one of the options, is foresee on all locations of built-up and planned dams. The storage hydro-profile is used for modeling of operation of storage pool and objects on the dam. This is done by conforming to all natural and artificial phenomena associated with water flow, which are described by relevant mathematical equations. These equations describe transformations in the storage, manageable and uncontrolled spilling over the spilling objects on dams, manageable water release through foundation outlets on dams, uncontrolled leakage through the dam and dam profile, uncontrolled evaporation from the water surface (exiting of water out of the system), formation of the aggregate flow downstream the dam including the definition of guaranteed discharge downstream from the hydro-profile.

The flow in a closed conduit is an element used to perform linear modeling of the flow through a tunnel or a pipeline whereby certain loss of potential energy occurs, but with preservation of equality between the inlet and outlet hydrographs. This element forms the link between the storage profile and hydro-profiles and the hydropower elements (hydropower plant, pump stations and reversible power plant), as well as the link between the storage hydro-profiles.

Open channel flow is an element used to perform linear modeling of the flow in the rivers in accordance with the morphological performances of the riverbed with the transformation of the inlet hydrograph into the outlet hydrograph according to equations of the Muskingum, i.e. Muskingum-Cunge model. This element is used to form the link between two active hydro-profiles. In the case of an inactive hydro-profile, the open-conduit flow links the upstream and the first active downstream hydro-profile. An entity of the open-conduit flow-type is determined automatically (generation-regeneration) based on the states of active hydro-profiles.

Hydropower plant is an element used to perform modeling of the electricity generation management and relevant water flow management. Hydropower plant in this model can be with the powerhouse at the base of the dam (water is taken in and returned into the watercourse within the same storage hydro-profile) or of the diversion-type (water intake is performed on one hydro-profile and returned back into another one). Tailwater of the hydropower plant may be the tailrace of the storage hydro-profile, flow-through hydro-profile or storage. Hydropower plant model of operation is based on the use of turbine exploitation chart (power – net head – discharge), including the losses in the inlet–outlet system of the hydropower plant.

As an option (depending on time discretization and problem type), the model uses several possible hydropower plant operational modes. In general, hydropower plant operation can be modeled with defined demand for power or electricity generation as the function of time, or hydropower plant operation depending on the inflow. There is a huge set of possibilities regarding the modeling of different management scenarios in terms of power and discharge distribution and discharges over units.

Pump station is an element used to perform modeling of energy consumption and the management of water flow (pumping) from the tail to the head storage in the pumping regime, as well as the modeling of modeling of electricity generation and the management of water flow from the head to the tail storage in the turbine mode of operation. It is also possible to model pump station operation in various modes: in general, pump station can operate either according to the demand for electricity generation or according to the level in the intake storage.

Reversible hydropower plant is an element used to perform modeling of energy consumption modeling and the management of water flow (pumping) from the lower to the upper storage in the pumping regime, as well as modeling of energy consumption and the management of water flow from the upper to lower storage in the turbine regime. Only two modes of reversible hydropower plant operation are envisaged by now: operation according to assigned energy in both regimes (turbine and pump), or the operation according to demand for power in turbine regime and operation according to the program in pump regime.

The user is an element for modeling of the managed water intake from the storage or the flow-through hydro-profiles by the user (water supply, irrigation) with a partial returning of water back to the system of into downstream hydro-profiles. The user defines the demand for water in the form of a hydrograph. Demands are met according to the priority defined during the configuration. Examples of water use are as follows: water supply for residential areas and industrial facilities, irrigation of agricultural areas, cooling of the facilities in thermo-power plants, etc.

The most important factor for development of the library of individual objects of hydro-systems is the name space devs.modelisation.model that contains the class of the atomic and the class of the coupled model.

3.4. Building of the complex simulation model based on DEVS model library

The class of each of mentioned objects is derived from the atomic model class through adaptation of standard functions of the generic DEVS atomic model. In the case of the storage hydro-profile, the input ports (natural and artificial), defined to represent the inflow and information on requirements assigned to the storage, as well as the output ports that transfer water to the objects (conditionally and unconditionally). Also, the time advance function is defined according to the quantified level of headwater, since the major part of storage functionality is related to the current water level. Quantification is performed on the completely arbitrary basis, but the intersection set must include important points such as elevation of the spilling threshold, minimum water level for intake and similar.

A similar procedure is performed for all other objects (flows, power plants, etc.). This is the way to form the library of classes of hydro-system objects for the hydro-systems which can be coupled in a modular fashion. This library can be extended with certain common systems, such as the storage coupled with the power plant, which facilitates to the user the formation of the complex model. To achieve this, the new class is derived from the class of the coupled model, and it is implementing all mechanisms required for coupling of the two mentioned atomic models. By the further hierarchical coupling of the model of mentioned entities it is possible to form the simulation model of the part or of the entire River Drina basin for different levels of construction and different assumptions regarding object performances (applicable both to the existing and future objects).

4. Example of application of DEVS model in simulation of complex hydro-systems

Based on the presented model library, simulation model has been developed and applied on the River Drina basin within a broader IT solution called "Drina" HIS. "Drina" Hydro-information system is a distributed hydro-information system used as a support to management of water of the River Drina basin (Divac et al. 2009). Simulation model is the fundamental part of the complex software and it represents the core of the distributed system for support to the integrated water management in the River Drina basin. The model handles water use and flow in a large and complex space which covers the entire River Drina basin (around 20.000 km^2). Since the water entry into the system is in the form of rainfall, and there exists a system of user requirements (demand for electricity generation as a function of time or demand for a certain water intake as a function of time), the model covers the formation of runoff from the rainfall by including the effects of snow, relief and soil, as well as all relevant forms of flow: flow through natural watercourses in line with morphological performances, flow through objects (dam spillways and outlets, hydropower plant, tunnels, channels, pipelines and other). Change of flow conditions is modeled as a function of the time due to management decisions (delivery, priorities and limitations in line with demands for electricity generation and water, as a function of the system state parameters). The model has been developed for calculations in daily and hourly time base as a part of decision-making support system.

4.1. Complex model of the River Drina basin

Full spatial decomposition of the entire River Drina basin (with consideration of the possible variants of future system development) forms the system configuration with the total of 127 hydro-profiles, 127 sub-catchments, 127 flows in open conduits, 27 flows in open conduits, 64 hydropower plants, 2 pump stations, 2 reversible hydropower plants and 43 users (water supply, irrigation). Any other state of construction of the River Drina hydro-system can be treated as a subsystem of the completed construction state.

All data used in "Drina" HIS are categorized and entered into a single database, primarily aimed at serving the model needs. This database includes the following information: system configuration data, performances of the existing and future objects (storages, spillways, outlets, hydropower plants, pump stations and similar), basin data (relief, vegetation, soil and similar), hydrographic network, watercourse characteristics, hydro-meteorological stations, hydrologic and metrological data, user data and other.

In addition to data on model objects (127 hydro-profiles, 64 hydropower plants and other), "Drina" HIS database includes data on 23654 HRUs with 118270 outflow functions, data on 10 vegetation types, 8 soil types and 6 hydrogeological structures found on the River Drina basin. Hydrographic network is made of 1957 nodes and 1955 river sections.

"Drina" HIS database includes historic data on mean daily values of water levels and discharges on 54 water-measuring stations in the River Drina basin, as well as the historical meteorological data (rainfall and temperature) on 54 measurement stations, making a data fund of more than 6 million daily values. All data is accessed through the HIS user interface, formed to interact with the model and the database (Figure 11).

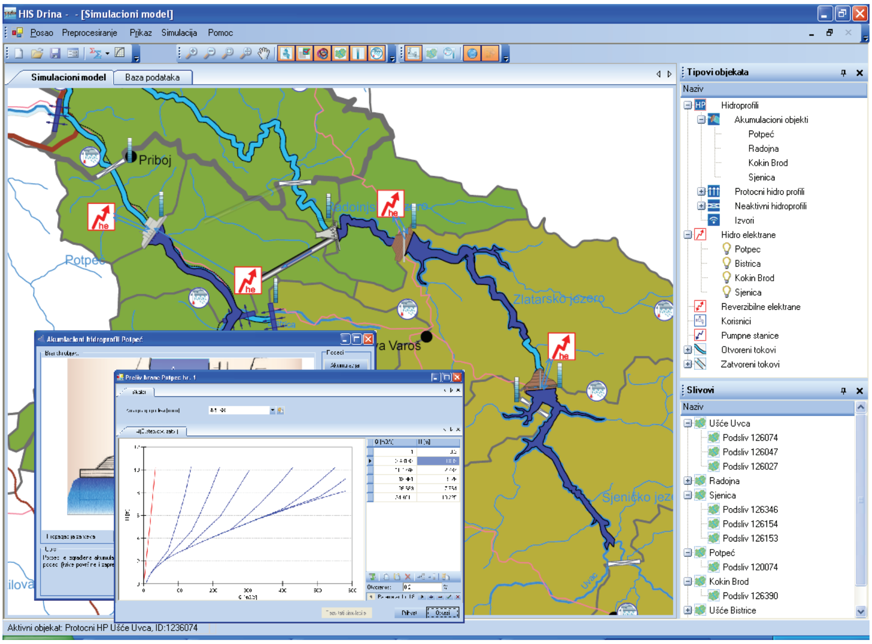

Fig. 11. One of the possible configurations

Information in the database is the foundation for formation of the simulation model. All object descriptions in the database are mapped into the object derived from the corresponding class (storage into the storage atomic model, power plant into the power plant atomic model and similar), parameters of which are transferred from the tables of hydro-object performances. This is a method to form the set of possible basic models that are used as the foundation for the formulation of a complex model. The configuration placed in the database and subsequent interaction with the user, who is making decisions on object states (active, inactive, future states and similar), constitutes the base of hierarchical compositions of atomic models into the complex coupled models until the desired configuration is reached.

4.2. Model application

Full control of model formation, simulation and result analysis is placed in the interaction between the user and the software package. Major portion of the process of object creation, automatic formation of coupled models and adjustment of parameters and inputs is automatic and no user action is required. Nevertheless, the user can influence on the whole range of different parameters and thus, analyze the problem in an interactive manner. The unique user interface was developed to provide efficient and convenient use. It serves as the intermediary between the user and simulation software. This is modern, graphically-oriented software that uses a series of windows and dialog boxes to lead the user though the simulation process in an interactive and intuitive manner.

Figure 12 shows the model configurations variants in terms of the present and future system development states. Due to the high flexibility of the simulation platform such structural changes of the complex model can be effectuated with no limitations.

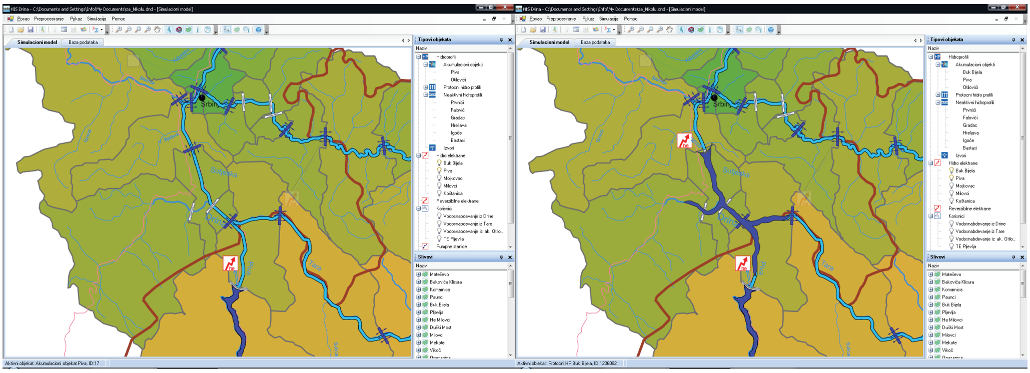

Fig. 12. Variants of model configuration

In addition to complex model structure, all parameters of real models can be changed at any time what allows for the deviation from the predefined values. Predefined parameter values facilitate the creation of initial models, because the objects are given the performances that belong to the existing hydro-objects that is, the planned objects if they describe the future state.

4.3. Simulation results

Results of the simulation are the hydrographs and water levels on dam profiles, hydrologic stations and other profiles (i.e. on all hydro-profiles), hydrographs of discharge through dam elements intended for water evacuation, realized electricity generation, number of committed power units, specific energy and discharge on hydropower plants (on all active units of the hydropower plant: generated electricity, efficiency, discharge through the turbine), water delivered to users, realized discharge through watercourses (open and closed), electricity consumed for pumping. Results are the time series in the selected discretization given in downloadable graphical and numerical forms (Figure 13 shows some of the simulation results).

5. Conclusions

A very important step in integral water management on a certain basin is the introduction of the hydro-information system for support to basin management, as an IT, technical and expert support to decision-making. The hydro-information system for support to water management in the River Drina basin is the way to establish a more dynamic and efficient dialog between all relevant subject in the basin in all decision-making phases, from strategic investment planning to operational management of exploitation, as well as at all levels of engagement, from measurement and gathering of information complex formulation of evidence in legal proceedings. The core of the system for support to decision-making is a complex simulation model of the entire basin, which covers numerous processes and provides simple extendibility in terms of all phenomena relevant to basin management. Composing a definition of this model type is made difficult by the fact that an integral simulation of heterogeneous models is required.

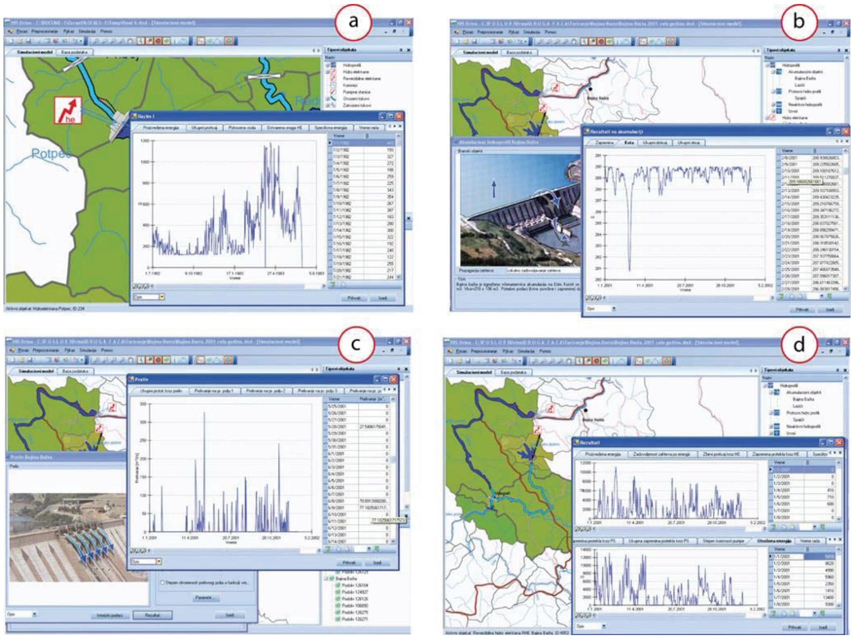

Fig. 13. Simulation Results (a – electricity generation on "Potpeć" HPP, b – realized water level in the "Bajina Bašta" storage, c – water spilling on the "Bajina Bašta" storage and d – reversible power plant operation).

One of the main deficiencies of the classic approach to modeling and simulation (solving of practical engineering problems) is the isolated approach to finding a solution. On the other hand, it is beyond doubt that modeling and simulation are assuming an interdisciplinary character more and more often, because the subject of their implementation are mainly more and more complex systems with diverse processes and their mutual interactions.

The presented simulation environment overcomes the boundaries set by the model types, simulation algorithms, operating systems and locations of computer resources. Presented system specifications and the corresponding simulation algorithms can be used for modeling and simulation of a wide range of real systems.

References

Abbott MB, Refsgaard JC (1996), Distributed Hydrological Modelling, Kluwer Academic Publishers, 1996.

Delaney W and Vaccari E (1989), Dynamic Models and Discrete Event Simulation, Marcel Dekker Inc., 1989.

Divac D, Grujović N, Milivojević N, Stojanović Z, Simić Z (2009), Hydro-Information Systems and Management of Hydropower Resources in Serbia, *Journal of the Serbian Society for Computational Mechanics*, Vol. 3, No. 1

Divac D, Grujović N, Milovanovic M (1999), Nov simulacioni model za bilansnu analizu vodoprivrednih sistema - Metodologija, softver i primena, Monografija: Upravljanje vodnim resursima Srbije. Beograd: Institut za vodoprivredu "Jaroslav Černi", 1999.

Enderton HB (1977), Elements of Set Theory, Academic Press, 1977.

European Parliament (2000), Directive 2000/60/EC of the European Parliament and of the Council of 23 October 2000 establishing a framework for Community action in the field of water policy, Official Journal L 327 , 22/12/2000. pp. 0001 - 0073

Grujović N, Divac D (2004), Numeričke procedure i algoritmi simulacionog modela sliva Drine, Časopis Vodoprivreda. Beograd: Vodoprivreda, 2004. pp. 123-130. Vol. 36.

HR Wallingford Report (2001), State of the Art Review, Work Package 1, Harmonit, IT Frameworks, Contract EVK1-CT-2001-00090.

Hopcroft JE, Motwani R, Ullman JD (2000), Introduction to Automata Theory, Languages, and Computation (2nd Edition). Pearson Education. ISBN 0-201-44124-1.

Mulvaney TJ (1850), On the use of self?registering rain and flood gauges in making observations of the relations of rainfall and of flood discharges in a give catchment, Proceedings of the Institution of Civil Engineers Ireland, 1850. pp. 18-31. No. 4.

OpenMI Document Series (2007), Part A - Scope for the OpenMI, The OpenMI Association, 2007.

Refsgaard J C, Knudsen J (1996), Operational validation and intercomparison of different hydrological models, Water Resources Research, 1996. pp. 2189-2202. No. 32.

Rippl W (1883), The capacity of storage reservoirs for water supply. Proceedings of the Institution of Civil Engineers, 1883. pp. 270-278. No. 71.

Sherman LK (1932), Streamflow from rainfall by the unit-graph method, Engineering News-Record, 1932. pp. 501-505. No. 205.

Simić Z, Milivojević N, Prodanović D, Milivojević V, Perović N (2009), SWAT-Based Runoff Modeling in Complex Catchment Areas – Theoretical Background and Numerical Procedures. *Journal of the Serbian Society for Computational Mechanics*, Vol. 3, No. 1

Singh VP (1995), Computer Models of Watershed Hydrology, Water Resources Publications, 1995.

Zeigler BP, Praehofer H, Tag GK (2000), Theory of Modeling and Simulation. New York: Academic Press, 2000.