

COMBINING MICRO AND MACRO-MODELING IN DEVS FOR COMPUTATIONAL BIOLOGY

Adeline M. Uhrmacher
Roland Ewald
Mathias John
Carsten Maus
Matthias Jeschke
Susanne Biermann

Modeling and Simulation Laboratory
University of Rostock
18059 Rostock, GERMANY

ABSTRACT

In computational biology there is an increasing need to combine micro and macro views of the system of interest. Therefore, explicit means to describe micro and macro level and the downward and upward causation that link both are required. Multi-Level-DEVS (or *ml-DEVS*) supports an explicit description of macro and micro level, information at macro level can be accessed from micro level and vice versa, micro models can be synchronously activated by the macro model and also the micro models can trigger the dynamics at macro level. To link both levels, different methods are combined, to those belong, value coupling, synchronous activations, variable ports, and invariants. The execution semantic of the formalism is given by an abstract simulator and its use is illustrated based on an small extract of the Wnt pathway.

1 INTRODUCTION

Modeling is about abstraction. Therefore, combining different levels of abstraction has been of interest in the community of modeling and simulation for a long time. Hybrid modeling approaches combine continuous and discrete perspectives on systems under study (Alur et al. 2000), qualitative scales are combined with quantitative scales (Fishwick and Zeigler 1992, Berleant and Kuipers 1997), and a macro perception of systems is combined with a micro perception. The latter we find traditionally in the sociological (Knorr-Cetina and Cicourel 1981), and in the biological area (Campbell 1974). Individual actors with their attributes and dynamics are described at micro level, whereas the macro level holds aggregated variables and functions that describe high level dynamics. In micro-macro modeling one of the central questions has always been how to relate the micro and the

macro level of a system (Alexander et al. 1987). This question typically takes the form of defining *upward* and *downward causation*, i.e. how parts of a system influence the system as a whole and how the system as a whole influences its parts (Campbell 1974). Despite this interest in micro-macro modeling, only little work has been done so far to directly support the micro-macro link explicitly in a modeling formalism (Köhler, Martens, and Rölke 2003, Uhrmacher 1995).

Computational biology is an application area that has attracted a lot of attention during the last years (Kitano 2002). Several tools exist to describe the flux and change of concentrations on a macro level, e.g. (Hoops et al. 2006, Klamt et al. 2003). There are also different strategies to describe biological processes individual based on a micro view level (Le Novère and Shimizu 2001, Henson, Müller, and Reuss 2002, Ewald et al. 2007). The combination of these two levels of detail in the same model can provide some advantages. In a biological system with strongly deviating component quantities, it could be appropriate to simulate only the processes with low concentrations in a high resolution (micro level) and the rest on a higher level with less detailed information. With multi-level models it is possible to focus on certain biological processes of interest, but one can also have a less detailed view on the overall behavior of the system. The interesting question from the point of modeling is how to link these two perspectives and how this is supported by the different modeling formalisms.

Diverse formalisms are used for modeling biological systems (de Jong 2002); among them traditional continuous, deterministic, macro models can be found as well as a variety of discrete event modeling formalisms (Ewald et al. 2007), like stochastic pi, stochastic petri nets, state charts, or DEVS. All of them provide specific views on the system of interest focusing either on a macro or on a micro view

(Uhrmacher and Kuttler 2006). For the combination of both, these formalisms need to be extended. Our exploration how to extend the formalism will be based on DEVS (Discrete Event Systems Specification) (Zeigler, Praehofer, and Kim 2000). As any other modular modeling formalism DEVS allows to describe macro and micro levels of a system and to connect those, e.g. to describe the Tryptophan Synthase (Uhrmacher, Degenring, and Zeigler 2004). However, it does not directly support it, as we will discuss in the next section. Thereafter, our modeling approach is described in section 3 followed by its operational semantics and an example.

2 MICRO- AND MACRO-MODELING IN DEVS

DEVS distinguishes between atomic and coupled models. Models communicate asynchronously via sending events over ports. Each coupled model comprises atomic or coupled models as components and defines their interaction via explicit couplings between input and output ports. Although a hierarchical composition of models is accomplished by coupling models, the behavior of coupled models is completely determined by their constituents and how they are coupled with each other. The formalism's inherent reductionism sometimes burdens modeling. This can be the case when certain events within a system depend on the overall state of the system, e.g. chemical reactions may depend on pressure, temperature, and the concentrations of the involved species. In DEVS no direct way exists of conveniently letting a higher-level property, i.e. a property of the whole system, influence the state of a sub-model.

As a workaround, coupled models are often equipped with sub-models that serve to manage the "high-level" properties of the whole coupled model, e.g., the *BulkSolution* model in the Tryptophan model handles the number of existing enzymes (Degenring, Röhl, and Uhrmacher 2004). However, there are several problems with this approach:

- Downward and upward causation are realized by exchanging events asynchronously. Communication by asynchronous events is entirely appropriate between independent subsystems which react to these external perturbations according to their internal rules of behavior. However, whether downward causation and upward causation between macro and micro level of a system is best described by exchanging events appears doubtful.
- Hierarchical composition of DEVS models and the metaphors of coupled and atomic model are misleading. Although the coupling of models might be associated with a compositional hierarchy of systems (Salthe 1985), coupled models serve merely as a container for their sub-models. They do not have a state nor a dynamic of their own.

- If micro and macro levels shall be supported, atomic models have to function as macro models and interact with all micro models. For realizing upward and downward causation, their ports have to be coupled with the ports of all micro models. Additionally, protocols for exchanging information from micro level to macro level and vice versa have to be realized. This has to be done for each macro and all micro models, which burdens modeling and simulation significantly.

Since there is no satisfactory way to circumvent the resulting difficulties with standard DEVS, it might be worth considering another possibility: to integrate a high-level model within the coupled DEVS model itself. This is not a new idea. Faced with the problem to introduce variable structures (changing composition and interaction patterns) into the DEVS formalism, the dynamic structure system network DSSN equips the coupled model with a behavior of its own (Barros 1997). The introduced executive of a coupled model is a hybrid between atomic and coupled model, and can access the structure of the entire model. Also in RUISEM (Pidd and Castro 1998) an executive is introduced which is able to change the couplings between components on demand and thus to realize a Jib-shop model efficiently in DEVS. However, in both downward and upward causation are only rudimentary supported (see section 6).

3 PROPOSED SOLUTION

Our approach of a Multi-Level-DEVS is based on two ideas. The first is to equip the coupled model with a state and a behavior of its own, such that the macro level does not appear as a separate unit (an executive) of the coupled model. Secondly, we have to explicitly define how the macro level affects the micro level and vice versa. Both tasks are closely interrelated.

Obviously, one means to propagate information from macro to micro level is to exchange events between models. However, this is rather tedious, e.g. in case the dynamics of a micro model has to take the global state into consideration. Therefore, we will adopt the idea of value couplings. Information at macro level is mapped to specific port names. Each micro model may access macro variables by defining input ports with corresponding names.

In the opposite direction, the macro level needs access to crucial information at the micro level. For this purpose, we equip micro models with the ability to change their ports and to thereby signalize crucial state changes to the outside world. Upward causation is supported, as the macro model has an overview of the number of micro models being in a particular state – i.e., exhibiting a particular set of ports – and to take this into account when updating the state at macro level.

Another question is, how are models “activated”? We assume that at macro level as well as at micro level models are still triggered by the flow of time and the arrival of events. In addition, the macro level can directly activate its components by sending them events – thereby, it becomes possible to synchronously let several micro models interact, which is of particular interest when modeling chemical reactions. Additionally, the dynamics at macro level can be activated by the dynamics at micro level, e.g. if the number of components being in a certain state (signalized by their ports) surpasses a certain threshold. Therefore, a form of invariant is defined at macro level, whose violation initiates a transition at macro level. This is inspired by the ideas of hybrid state automata, where the discrete state changes are triggered at the moment the continuous dynamics lead to threshold crossing.

Now let us have a look at the *ml*-DEVS formalism, which comprises a definition for atomic models, MICRO-DEVS, and coupled models, MACRO-DEVS.

Let $X, Y, S = (V; S_1, \dots, S_n)$ be structured sets with $V = \{v_1, \dots, v_n\}$ (see Zeigler, Praehofer, and Kim 2000, p. 124). The input ports of the structured set X may now also hold information handed down via value coupling (see definition of the MACRO-DEVS model). Furthermore, let \mathcal{P} be the set of available port names.

MICRO-DEVS models are different from the typical atomic models as they do no longer consist of internal, external and confluent transition functions, only one transition function δ exists. The main reason for this is the clarity of the formalism. Moreover, the distinction between internal and external state transitions is partly rendered meaningless by the fact that δ_{int} may now rely on macro variables, which are accessed over value-coupled ports. With a single state transitions function, one may now decide which different kinds of transitions have to be distinguished for the model at hand. Since the elapsed time and the inputs at all ports are accessible, the modeler can define what to do under which circumstances.

As in regular DEVS, the λ function is invoked just before an internal or confluent event happens. This situation is recognized by the simulator. A time advance function ta is given that associates a duration with each state. Additionally, a MICRO-DEVS model has a function $p: S \rightarrow 2^{\mathcal{P}}$, which defines the ports the model exhibits in a given state.

Definition 1 A MICRO-DEVS is defined as a structure

$$\langle X, Y, S, s_{init}, p, \delta, \lambda, ta \rangle$$

where

- X : the structured set of inputs
- Y : the structured set of outputs
- S : the structured set of states
- $s_{init} \in S$: the start state

- p : $S \rightarrow 2^{\mathcal{P}}$ selects the ports available in a given state
- δ : $X \times Q \rightarrow S$ state transition function
- λ : $S \rightarrow Y$ output function
- ta : $S \rightarrow \mathcal{R}^{\geq 0} \cup \{\infty\}$ time advance function

The definition of a MACRO-DEVS model is based on the definition of a MICRO-DEVS model. Similar to a MICRO-DEVS model, a MACRO-DEVS model has structured input and output sets X and Y and a state set S . The input ports might hold events or information that was handed down via value coupling from its superordinate MACRO-DEVS model. A λ function produces outputs for the output ports. Similar to coupled models in DEVS, a set C of components is defined.

Unlike DEVS, which realizes an extensional definition of couplings between individual ports, we adopt the idea of multi-couplings that realize, in conjunction with variable ports, an elegant mechanism of dynamic coupling (see ρ -DEVS, Uhrmacher et al. 2006). Multi-couplings are intensionally defined as a set MC of functions that map output ports to input ports, typically by referring to certain types (here: names). The transition function δ at macro level takes the state, the information about the model’s components and multi-couplings into account when calculating the new state. Again, the function p associates a set of ports with each state. The structural change function sc defines the set of components and multi-couplings for the MACRO-DEVS model’s current state.

In what follows, we describe how upward and downward causation are realized in *ml*-DEVS.

Downward information: The downward causation at information level is realized by the function v_{down} , which couples state variables of the MACRO-DEVS model to input ports of its MICRO-DEVS models. Thereby, each relevant variable at macro level is directly accessible by the micro models via their input ports. This implies that the input ports of the MICRO-DEVS models are never “empty,” because value-coupled information is always accessible.

Downward activation: The downward activation is done by the λ_{down} function, which allows to synchronously trigger a number of micro models by sending them events. This does not require a coupling between macro and micro models, as a MACRO-DEVS model may directly access the ports of its components.

Upward information: The information propagation from micro to macro level is realized by changing ports. The macro model can access the information which ports are available and does so in its δ function to determine the next state at macro level. Each model can change its ports via the function p and thus can signalize important information to other micro models (Uhrmacher et al. 2006) and the macro level.

Upward activation: Changes at the micro level can initiate changes at the macro level. The activation constraint act_{up} guards that the invariants at macro level are fulfilled, otherwise the invocation of the MACRO-DEVS model's δ function is triggered. This leads to a new state with a possibly new set of components and multi-couplings.

A MACRO-DEVS model is formally defined as follows:

Definition 2 A MACRO-DEVS is defined as a structure

$$\langle X, Y, S, s_{init}, p, C, MC, \delta, \lambda_{down}, v_{down}, sc, act, \lambda, ta \rangle$$

where

$$\begin{aligned} C: & \text{ set of sub-models which are of type} \\ & \text{MICRO-DEVS or MACRO-DEVS} \\ MC: & \text{ set of multi-couplings, } \{m \mid m: 2^{\mathcal{P}} \rightarrow 2^{\mathcal{P}}\} \\ \delta: & X \times Q \times 2^{C \times \mathcal{P}} \rightarrow S \text{ state transition function} \\ \lambda_{down}: & S \rightarrow 2^{Y \times C \times \mathcal{P}} \text{ downward output function} \\ v_{down}: & V_S \rightarrow \mathcal{P} \text{ value coupling downward} \\ sc: & S \rightarrow 2^C \times 2^{MC} \text{ structural change function} \\ act_{up}: & S \times 2^{C \times \mathcal{P}} \rightarrow \{true, false\} \text{ activation} \\ & \text{function} \end{aligned}$$

Moreover, the following has to hold:

- If a port is an output port, it cannot be an input port:
 $\forall p \in \mathcal{P} : (\exists m \in MC \wedge P \in 2^{\mathcal{P}} : p \in P \wedge m(P) \neq \emptyset) \implies (\nexists m' \in MC \wedge P' \in 2^{\mathcal{P}} : p \in m'(P'))$
- Value coupling is defined on ports that are no input ports:
 $\forall v_S \in V_S : (\nexists m \in MC \wedge P \in 2^{\mathcal{P}} : v_{down}(v_S) \in m(P))$

All other elements of the tuple are defined as for MICRO-DEVS models. The dependencies between macro and micro level could easily lead to an algebraic loop. This is prevented by the simulator, which defines the execution semantics of *ml-DEVS*.

4 SIMULATION

Following the tradition of all DEVS formalisms (Zeigler, Praehofer, and Kim 2000), the execution semantics of the model formalism is given by an abstract simulator. The abstract simulator is structured into simulators which execute the atomic, i.e. MICRO-DEVS models, and coordinators, which are responsible for executing MACRO-DEVS models.

4.1 Simulator

The simulator is similar to those that have been realized in JAMES II (Himmelspach and Uhrmacher 2004). The difference is that only one transition function, δ , is invoked

```

1  when receive * message
2    y = m.λ(m.state)
3    send y message
4
5  when receive x message
6    m.state = δ(x, m.state, t - tole)
7    tole = t
8    tonie = t + m.ta(m.state);
9    ports = m.p(m.state)
10   send done message with ports and tonie

```

and that the ports are changed by invoking the function p . In the *done message*, the time of next internal event (tonie) as well as the model's ports are sent to the parent.

4.2 Coordinator

The coordinator combines functionalities of other DEVS coordinators and DEVS simulators. All imminents receive a * message which will lead to the activation of their lambda function (line 9). In addition, the time could be expired for the macro model, i.e. $t=tonie(m)$. In this case, the λ function of the macro model has to be invoked too (12). Besides the regular λ function, a downward activation will be triggered by invoking the function λ_{down} (13). The events produced by the regular λ functions at macro and micro level will be sent to the parent via the associated output ports (15).

As in other abstract DEVS simulators, the coordinator will wait for its inputs from outside (16). The events will be propagated as in the regular coordinator (20-28). Additionally, all somehow activated components have to be informed about significant state changes at macro level. The simulator will simply fill the ports with the required information. This could be avoided by using a shared variable, but as most algorithms in JAMES II are built for being distributed at some point, it is more convenient to update the information each time it might be needed. Please recall, this is not a traditional external event, because additional information about the state of the macro model is added according to the defined value coupling (23). The availability of this information is granted and does not trigger the transition function at micro level. Then, the coordinator keeps track of its sub-model's ports (31-34) and finally checks whether it has to invoke its macro model's transition function (38). This is triggered either by inputs that are reaching the macro model from the outside, by the flow of time, or by fulfilling the activation constraint.

5 EXAMPLE

We would like to illustrate the idea with an example. As the Wnt signaling pathway is crucial for cell differentiation, it is an important topic for many researchers. Here we will focus on the activities in the nucleus. The protein Groucho

```

1 //Created during initialization:
2 ports = map from sub-models to their current
   ports (C×P)
3
4 when receive * or x message with time t
5
6 //Querying all imminent models and
   distributing their output
7 xyd = 0
8 if message is * message
9   send * to d ∈ IMM = {c ∈ m.C | tonie(c)=t}
10  wait for xyd messages from all children ∈
   IMM
11  if t = tonie(m)
12    xymacro = m.λ(m.state)
13    xdown = m.λdown(m.state)
14  fi
15  send xymacro ∪ toParent(xyd) message to
   parent
16  wait for xyfromParent message from parent
17 fi
18
19 //Executing all imminent or influenced models
20 msg = 0
21 for all d ∈ IMM ∪ INF
22   msg = msg ∪ create msg from xyd ∪ xyfromParent
   bound for d as def. in m.mc
23   msg = msg ∪ create msg for d as def. in m.
   vdown
24 endfor
25 if t = tonie(m)
26   msg = msg ∪ xdown
27 fi
28 send messages in msg
29
30 //Register port changes
31 ports = 0
32 wait for done msg from all d ∈ IMM ∪ INF:
33   add (d, done.ports) to ports
34   store done.tonie
35
36 //Execute macro model
37 μ = getMsgForMe(xyfromParent)
38 if μ ≠ 0 ∨ t=tonie(m) ∨ actup(m.state, ports)
39   m.state = m.δ(μ, m.state, t - tole(m),
   ports)
40   (m.comp, m.mc) = m.sc(m.state)
41   myports = p(m.state)
42 fi
43
44 send done msg with myports and tonie =
45   minc∈m.comp∪{m} tonie(c)

```

interacts together with the T-cell specific transcription factor (TCF) as a transcription repressor. β -catenin can also bind to TCF and acts as a transcriptional coactivator. Therefore, β -catenin activates the transcription initiation by replacing the protein Groucho from TCF (Daniels and Weis 2005). In absence of a Wnt signal the activity of a huge list of target genes is downregulated because of the lack of β -catenin in the nucleus. A Wnt signal results in a higher nuclear concentration of β -catenin and increases the expression of different gene products by transcription activation. The

```

1 X = {(betaIn, grouchoIn); N, N}
2 Y = {(betaOut, grouchoOut); N, N}
3 S = {σ, #beta, #grou, #betaTcf, #grouTcf,
4       tBeta, tGroucho, tGeneAct, tGeneRep} |
5       σ ∈ {geneOcc, geneFree},
6       #beta, #grou, #betaTcf, #grouTcf ∈ N,
7       tBeta, tGroucho, tGeneAct, tGeneRep ∈ ℙ0+}
8 sinit = (geneFree, #betainit, ...)
9
10 C = {TCF1 ... TCFn, Gene}
11
12 MC = {m} with m({bGene}) = {bSite} else 0
13
14 p = {betaIn, grouchoIn, betaOut, grouchoOut}
15
16 δ = if elapsedTime == ta(σ) then
17   #beta -= #betaMigrate(#beta)
18   #grou -= #grouchoMigrate(#grou)
19   if elapsedTime ∈ {tGeneAct, tGeneRep} then
20     σ = geneOcc
21   if betaIn ≠ 0 then #beta += betaIn
22   if grouchoIn ≠ 0 then #grou += grouchoIn
23   if actup then
24     #beta += #betaTCF - count(C, beta)
25     #betaTCF = count(C, beta)
26     #grou += #grouTCF - count(C, groucho)
27     #grouTCF = count(C, groucho)
28     if count(C, bGene) == 0 then σ = geneFree
29     tBeta = nextBetaBind(#beta, count(C, free))
30     tGroucho = nextGrouBind(#grou, count(C, free))
31     tGeneAct = nextGeneAct(#betaTCF)
32     tGeneRep = nextGeneRep(#grouTCF)
33
34 sc = (C, {m})
35
36 actup = if (#betaTcf ≠ count(C, beta)) ∨
37   (#grouTcf ≠ count(C, groucho)) ∨
38   (σ == geneOcc ∧ count(C, bGene) == 0)
39   then true
40   else false
41
42 λ = betaOut = #betaMigrate(#beta)
43   grouchoOut = #grouchoMigrate(#grou)
44
45 λdown = time = min(tBeta, tGroucho, tGeneAct, tGeneRep)
46 case time of
47   tBeta: ("beta!", pick(C, free), free)
48   tGroucho: ("grou!", pick(C, free), free)
49   tGeneAct: ("bind!", pick(C, beta), beta),
50     ("activate!", Gene, bSite)
51   tGeneRep: ("bind!", pick(C, groucho),
52     groucho), ("repress!", Gene, bSite)
53
54 vdown = 0
55
56 ta = min(tBeta, tGroucho, tGeneAct, tGeneRep)

```

Figure 1: The nucleus as a MACRO-DEVS model.

activities in the nucleus can be described as a set of rather simple processes:

- β -catenin and Groucho migrates into and from the nucleus.


```

1 X = {(free, beta, groucho); ...}
2 Y = {(bGene); ...}
3 S = {free, bGroucho, bBeta, act, rep}
4 sinit = (free)
5
6 p = case s of
7     bBeta: {beta}
8     bGroucho: {groucho}
9     act: {beta, bGene}
10    rep: {groucho, bGene}
11    free: {free}
12
13 δ = case s of
14     free: if free == "beta!" then bBeta
15           else bGroucho
16     bBeta: if beta == "bind!" then act
17            else free
18     bGroucho: if groucho == "bind!" then rep
19                else free
20     act: bBeta
21     rep: bGroucho
22
23 λ = if s ∈ {act, rep} then "unbind!"
24
25 ta = case s of
26     free: ∞
27     bBeta: expRand(rateBeta)
28     bGroucho: expRand(rateGroucho)
29     act: expRand(rateTrans)
30     rep: expRand(rateRep)
    
```

Figure 2: TCF model as a MICRO-DEVS model.

- In the nucleus, β -catenin and Groucho compete in interacting with TCF.
- Interaction with TCF can lead to either activation (in combination with β -catenin) or repression (in combination with Groucho) of the gene transcription.

We describe TCF and the gene as micro models and the concentration of β -catenin and Groucho as part of the macro model. To keep our example model small, we assume that β -catenin cannot displace Groucho directly from TCF, although there is some evidence for it (Daniels and Weis 2005). We also assume that TCF can only bind to the gene if it is bound to β -catenin or Groucho but not in its free state.

The interesting question is how upward and downward causation are realized.

Downward causation: the macro level influences the micro level

- Macro-variables could be accessed from the micro level, e.g. the temperature, however we did not make use of this in the example.
- The macro-level sends events to the micro level, e.g. to facilitate synchronous interactions between two or more partners, in this case it means that TCF/ β -catenin or TCF/Groucho binds to the gene whose

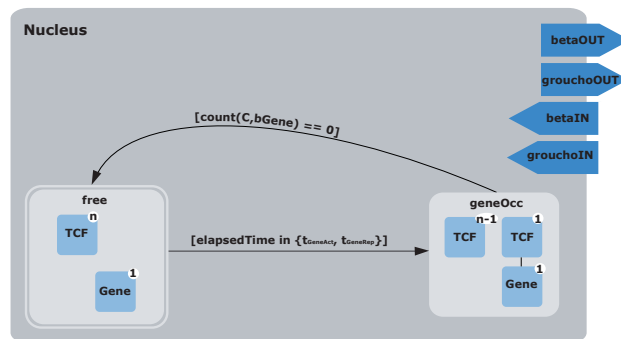


Figure 3: Visualization of the nucleus model.

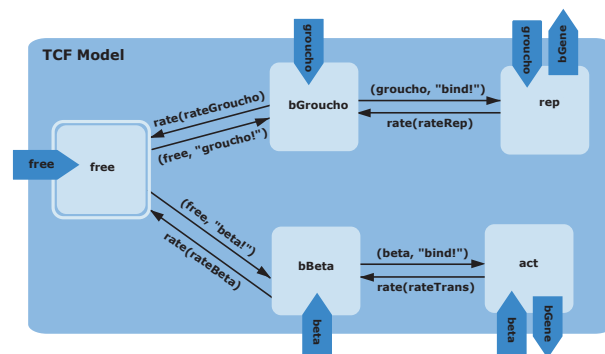


Figure 4: Visualization of the TCF model.

transcription process becomes now activated, or repressed.

Upward causation: the micro-level influences the macro-level

- Ports signalize crucial state changes at micro-level, e.g. TCF is bound to β -catenin or Groucho
- Multi-couplings make use of this information to install connections automatically, e.g. between a TCF that has the “bGene” port available.
- Crucial changes at micro level trigger events at macro-level, e.g. TCF falling off the gene.

The nucleus model (Figure 1) has only static ports (1-2). Also no downward value coupling is required (54). If we had described the binding of β -catenin or Groucho to TCF as part of the dynamics of TCF, a value coupling downward that links the number of β -catenin and Groucho molecules at macro level with the input ports of the TCF at micro level would have been necessary. However, due to the fact that β -catenin and Groucho changes frequently at macro level and requires re-calculating the probability of TCF binding with β -catenin or Groucho, and thus a re-scheduling of events, we kept scheduling the event of a TCF binding to β -catenin or Groucho at macro level (47-48). In the model only little need for usual couplings exists. The multi-coupling links the ports “bGene” and “bSite” (12). All other ports are not connected.

```

1 X = {(bSite); ...}
2 Y = {}
3 S = {free, trans, rep}
4 sinit = (free)
5
6 p = {bSite}
7
8 δ = if s == free then
9     if bSite == "activate!" then trans
10    else rep
11    else free
12
13 λ = {}
14
15 ta = ∞

```

Figure 5: Gene model as a MICRO-DEVS model.

As soon as a TCF exposes the port with name “bGene” a coupling to the “bSite” port of the gene will be realized. As in ρ -DEVS multi-couplings are defined intensionally and not extensionally. In the transition function we have to distinguish between external events, meaning that β -catenin and Groucho are entering the nucleus (21-22) and internal or confluent events when β -catenin and Groucho molecules will leave the nucleus (17-18). If the transition has been triggered by the invariant (36-40), either the numbers of β -catenin and Groucho have to be updated (24-27), or a TCF has fallen from the gene (28). The MACRO-DEVS model realizes a Gillespie algorithm (Gillespie 1977), recalculating the propensities for certain reactions to take place (29-32). Having defined the downward and upward causation in a consistent manner, it now would be easy to add further reactions to the macro level.

The MICRO-DEVS models are comparatively simple. Each individual TCF follows its own rules. The dynamics for which TCF signs responsible are: reacting to the incoming message that β -catenin and Groucho binds to it (14-15), that β -catenin and Groucho fall off (17,19), that TCF dock to the gene, activating (16) or repressing (18) the transcription, and that it falls off the gene (20-21,23). With each state ports can change (6-11). The gene is also modeled at micro level, and is currently rather simple.

As the formalism is not easy to read, we are currently developing a visualization (Figures 3, 4 and 6) that adopts ideas from State Charts (Harel 1987) and hybrid automata, i.e. Charon (Alur et al. 2000), to visualize the structure of the model and its crucial changes, e.g. the port changes of the TCF model (Figure 4).

6 RELATED WORK

In developing *ml*-DEVS we followed an idea put forward in EMSY (Uhrmacher 1993, Uhrmacher 1995) and applied it to DEVS. As stated above, the idea to combine a compositional hierarchy with means to equip the macro level with an

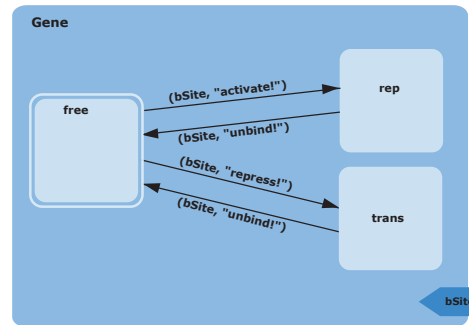


Figure 6: Visualization of the gene model.

explicit state and rules of behavior is not new, e.g. (Barros 1997, Pidd and Castro 1998). The definition of DSSN is based on the vision of an executive that resides as a kind of all-mighty atomic model in the coupled model (Barros 1997). For keeping track of important changes within its components, “events” are exchanged between the executive and the other components of the model which are coupled to the executive. The purpose of the executive is to support the initiation of variable structures in a top down manner. Other approaches avoid the need for an executive by realizing additional software layers (Mittal, Mak, and Nutaro 2006). In contrast to these approaches, in *ml*-DEVS state and behavior are directly associated globally to the coupled model. Therefore, neither couplings between micro and macro level exist nor information is exchanged via events. Also RUISEM (Pidd and Castro 1998) introduces a central unit into the coupled model, an executive which can access its components, however, the components cannot access the executive. For information flow between macro and micro RUISEM relies on couplings and the exchange of events.

ml-DEVS utilizes a set of different methods to realize the downward and upward causation. For the downward information flow we adopted the idea of value couplings, which are proposed for combining continuous and discrete DEVS models (Zeigler, Praehofer, and Kim 2000). They are often used as a basis for coupling continuous models, if a modular modeling shall be supported, e.g. in Modelica (Elmqvist, Mattsson, and Otter 2001). In the opposite direction, information is propagated from micro to macro level as the macro level observes the significant state changes at the micro level due to changing ports. In the context of DEVS introducing variable ports has been suggested in (Hu, Zeigler, and Mittal 2005) and ρ -DEVS (Uhrmacher et al. 2006). By changing ports individual entities signalize significant changes to their environment. In addition, multi-couplings from ρ -DEVS are used to install couplings depending on the availability of ports. These ideas in ρ -DEVS are inspired by work done in the area of process algebras most notably stochastic π (Priami 1995) and Beta binders (Priami and Quaglia 2005).

The macro model is now able to synchronously activate an arbitrary number of different micro models (that they are being in the correct state is signaled by their ports), to send them the required information, and to induce a synchronous interaction between them. Thereby, the modeling of chemical reactions has been facilitated a lot. In addition, neither a sending and receiving part of the interaction has to be distinguished nor is the number of actors within a starting reaction restricted to two. Both is the case in Stochastic π . A synchronous activation of more than 2 reactants of a chemical reaction can be described easily, similarly as it can be done in the process algebra PEPA (Calder, Gilmore, and Hillston 2004). So by realizing the micro-macro link in *ml*-DEVS certain desirable features for modeling biochemical systems came for free.

An approach that addresses explicitly the problem of capturing the micro-macro link within a formalism is described in (Schillo, Fischer, and Klein 2001). The micro-macro link has been modeled based on Reference Petri Nets (Valk 2001). Reference Petri Nets are Petri Nets within Petri Nets. A token can represent a “micro” net, which is consumed and produced by the transitions at macro level. Upward and downward activation are not distinguished, and information at a different level is only implicitly accessed by means of transitions being activated. The application of a Reference Petri Net for micro-macro modeling has been discussed in the context of qualitative modeling, thus, for quantitative discrete event simulating the approach has to be extended.

7 CONCLUSION

To support micro-macro modeling in cell biology we added yet another modeling formalism to the already large family of DEVS variants (Zeigler, Praehofer, and Kim 2000): *ml*-DEVS. The modeling formalism is aimed at explicitly describing the micro and macro level of a system and the linkage between both levels in terms of upward and downward causation. Therefore, we equipped the DEVS coupled model with a state and a behavior of its own. The downward causation is realized by value couplings with which micro level models can access information at macro level and by the ability of the macro model to synchronously activate several micro models via events. The latter facilitates the modeling and simulation of chemical reactions. The upward causation from micro to macro is based on the ability of micro models to change their ports. The macro model can access the information by defining an activation function. As soon as the activation constrain, i.e. a certain situation of the components signaled by their ports, is fulfilled the transition function at macro level is triggered.

Although Multi-Level-DEVS supports an explicit modeling of different abstraction layers, the formal semantics

cannot ensure that these layers do not overlap. It is still the task of the modeler to choose the abstraction level for each entity. As do other DEVS variants, *ml*-DEVS supports an arbitrary nesting of models, and the combination with other DEVS variants. Also in the tradition of DEVS we defined an abstract simulator to make the execution semantics of the new formalism explicit. Unfortunately, the new formalism shares also problems with other DEVS variants: the act of modeling is not very intuitive, e.g. for Biologists. Therefore, we are currently looking into possibilities for a graphical more intuitive notation for describing cellular systems at micro and macro level and the link between both.

ACKNOWLEDGMENTS

This research is supported by the DFG (German Research Foundation) as part of the research projects DIERMOSIS and DIEM OSIRIS.

REFERENCES

- Alexander, J., B. Giessen, R. Munch, and N. Smelser. (Eds.) 1987. *The Micro Macro Link*. Berkeley University of California Press.
- Alur, R., R. Grosu, Y. Hur, V. Kumar, and I. Lee. 2000. Modular Specification of Hybrid Systems in CHARON. In *HSCC '00: Proceedings of the Third International Workshop on Hybrid Systems: Computation and Control*, 6–19: Springer-Verlag.
- Barros, F. J. 1997. Modeling formalisms for dynamic structure systems. *ACM Trans. Model. Comput. Simul.* 7 (4): 501–515.
- Berleant, D., and B. Kuipers. 1997. Qualitative and Quantitative Simulation: Bridging the Gap. *Artificial Intelligence* 95 (2): 215–255.
- Calder, M., S. Gilmore, and J. Hillston. 2004, August. Modelling the influence of RKIP on the ERK signalling pathway using the stochastic process algebra PEPA. In *Proceedings of the BioConcur Workshop on Concurrent Models in Molecular Biology*, ed. A. Ingolfsdottir and H. R. Nielson. London, England.
- Campbell, D. T. 1974. Downward causation' in hierarchically organised biological systems. In *Studies in the philosophy of biology: Reduction and related problems*, ed. F. J. Ayala and T. Dobzhansky, 179–186. London/Basingstoke: Macmillan.
- Daniels, D. L., and W. I. Weis. 2005, March. Beta-catenin directly displaces Groucho/TLE repressors from Tcf/Lef in Wnt-mediated transcription activation. *Nature Structural & Molecular Biology* 12 (4): 364–371.
- de Jong, H. 2002. Modeling and Simulation of Genetic Regulatory Systems: A Literature Review. *Journal of Computational Biology* 9 (1): 67–103.

- Degenring, D., M. Röhl, and A. Uhrmacher. 2004. Discrete Event, Multi-Level Simulation of Metabolite Channeling. *BioSystems* 75 (1-3): 29–41.
- Elmqvist, H., S. E. Mattsson, and M. Otter. 2001. Object-Oriented and Hybrid Modeling in Modelica. *Journal Européen des systèmes automatisés* 35 (1): 1–10.
- Ewald, R., C. Maus, A. Rofls, and A. M. Uhrmacher. 2007. Discrete event modeling and simulation in systems biology. *Journal of Simulation* 1 (2): 81–96.
- Fishwick, P. A., and B. P. Zeigler. 1992. A Multimodel Methodology for Qualitative Model Engineering. *Modeling and Computer Simulation* 2 (1): 52–81.
- Gillespie, D. 1977. Exact stochastic simulation of coupled chemical reactions. *The Journal of Physical Chemistry* 81 (25): 2340–2361.
- Harel, D. 1987. Statecharts: A visual formalism for complex systems. *Science of Computer Programming* 8 (3): 231–274.
- Henson, M. A., D. Müller, and M. Reuss. 2002, December. Cell population modelling of yeast glycolytic oscillations. *Biochem J* 368 (Pt 2): 433–446.
- Himmelspach, J., and A. M. Uhrmacher. 2004. A component-based simulation layer for JAMES. In *Parallel and Distributed Simulation, 2004. PADS 2004. 18th Workshop on*, 115–122.
- Hoops, Stefan, Sahle, Sven, Lee, Christine, Pahle, Jurgen, Simus, Natalia, Singhal, Mudita, Xu, Liang, Mendes, Pedro, Kummer, and Ursula. 2006, December. COPASIA COMplex PATHway SIMulator. *Bioinformatics* 22 (24): 3067–3074.
- Hu, X., B. Zeigler, and S. Mittal. 2005. Variable Structure in DEVS Component-Based Modeling and Simulation. *Simulation* 81 (2).
- Kitano, H. 2002, March. Systems biology: a brief overview. *Science* 295 (5560): 1662–1664.
- Klamt, S., J. Stelling, M. Ginkel, and E. D. Gilles. 2003, January. Fluxanalyzer: exploring structure, pathways, and flux distributions in metabolic networks on interactive flux maps. *Bioinformatics* 19 (2): 261–269.
- Knorr-Cetina, K., and A. Cicourel. (Eds.) 1981. *Advances in Social Theory and Methodology - Towards an Integration of Micro and Macro Sociologies*. Boston: Routledge and Kegan Paul.
- Köhler, M., M. Martens, and H. Rölke. 2003. Modelling social behaviour with Petri net based multi-agent systems. In *Proceedings of the Workshop MASHO'03 at the KI 2003*.
- Le Novère, N., and T. S. Shimizu. 2001, June. STOCHSIM: modelling of stochastic biomolecular processes. *Bioinformatics* 17 (6): 575–576.
- Mittal, S., E. Mak, and J. J. Nutaro. 2006, October. Devs-based dynamic model reconfiguration and simulation control in the enhanced dodaf design process. *Journal of Defense Modeling and Simulation* 3 (4): 239–267.
- Pidd, M., and R. Castro. 1998. Hierarchical modular modelling in discrete simulation. *Proceedings of the 30th conference on Winter simulation*:383–390.
- Priami, C. 1995. Stochastic π -calculus. *Computer Journal* 6:578–589.
- Priami, C., and P. Quaglia. 2005. Beta binders for biological interactions. *Transactions on Computational Systems Biology*.
- Salthe, S. 1985. *Evolving Hierarchical Systems*. Columbia University Press.
- Schillo, M., K. Fischer, and C. T. Klein. 2001. The Micro-Macro Link in DAI and Sociology. In *MABS '00: Proceedings of the Second International Workshop on Multi-Agent-Based Simulation-Revised and Additional Papers*, 133–148. London, UK: Springer-Verlag.
- Uhrmacher, A., D. Degenring, and B. Zeigler. 2004. Multi-Level Discrete-Event Modeling in Systems Biology. *Computational Methods in Systems Biology*.
- Uhrmacher, A. M. 1993, September. Variable Structure Models: Autonomy and Control - Answers from Two Different Modeling Approaches. In *Proc. of the 4th Annual Conference on Artificial Intelligence, Simulation and Planning in High Autonomy Systems*, 133–142. San Diego: IEEE-Press. Arizona, Tucson.
- Uhrmacher, A. M. 1995. Reasoning about Changing Structure: A Modeling Concept for Ecological Systems. *International Journal on Applied Artificial Intelligence* 9 (2): 157–180.
- Uhrmacher, A. M., J. Himmelspach, M. Röhl, and R. Ewald. 2006. Introducing Variable Ports and Multi-Couplings for Cell Biological Modeling in DEVS. In *Proc. of the 2006 Winter Simulation Conference*.
- Uhrmacher, A. M., and C. Kuttler. 2006. Multi-level modeling in systems biology by discrete event approaches. *IT Themenheft Systems Biology*.
- Valk, R. 2001. Concurrency in communicating object petri nets. 164–195. Secaucus, NJ, USA: Springer-Verlag New York, Inc.
- Zeigler, B. P., H. Praehofer, and T. G. Kim. 2000. *Theory of Modeling and Simulation, Second Edition*. Academic Press.

AUTHOR BIOGRAPHIES

ADELINDE M. UHRMACHER is an Associate Professor at the Department of Computer Science at the University of Rostock and head of the Modeling and Simulation Group. Her research interests are in modeling and simulation methodologies, particularly multi-level, parallel, distributed modeling and simulation and their applications. Her e-mail address is lin@informatik.uni-rostock.de and the Web page of the group can be found at www.mosi.informatik.uni-rostock.de.

ROLAND EWALD holds a MSc in Computer Science from the University of Rostock and is a PhD student in the Modeling and Simulation group. His research interests are in distributed, parallel simulation.

MATHIAS JOHN holds a MSc in Computer Science from the University of Rostock and is a PhD student in the Modeling and Simulation group. His research interests are in visualization and modeling formalisms and their applications to cell biological systems.

CARSTEN MAUS holds a MSc in Biology from the University of Düsseldorf and is a PhD student in the Modeling and Simulation group. His research interests are in multi-level modeling and simulation in Systems Biology.

MATTHIAS JESCHKE holds a MSc in Computer Science from the Technical University of Dresden and is a PhD student in the Modeling and Simulation group. His research interests are in parallel, distributed simulation and the design of efficient simulation engines for cell biological models.

SUSANNE BIERMANN holds a MSc in Computer Science from the University of Rostock and is a PhD student in the Modeling and Simulation group. Her research interests are in visualization, and modeling in Systems Biology.