Modelling the Dynamics of Intracellular Processes as an Organisation of Multiple Agents

Tibor Bosse[1], Catholijn M. Jonker[2] and Jan Treur[1]

[1]Vrije Universiteit Amsterdam, Department of Artificial Intelligence
De Boelelaan 1081a, NL-1081 HV Amsterdam, The Netherlands
[2]Radboud Universiteit Nijmegen, Nijmegen Institute for Cognition and Information,
Montessorilaan 3, 6525 HR Nijmegen, The Netherlands

Abstract. This paper explores how the dynamics of complex biological processes can be modeled as an organisation of multiple agents. This modelling perspective identifies organisational structure occurring in complex decentralised processes and handles complexity of the analysis of the dynamics by structuring these dynamics according to an organisational structure. More specifically, dynamic properties at different levels of aggregation in the organisational structure are identified, and related to each other according to the organisational structure. The applicability of this organisational modelling approach to address complexity in biological context is illustrated by a case study: the organisation of intracellular processes.

1 Introduction

To handle complex decentralised dynamics, often some type of organisational structure is exploited. The dynamics that emerge from multiple interacting agents within human society have been studied within Social Sciences in the area of Organisation Theory and within Artificial Intelligence in the area of Agent Systems; e.g., [8], [9], [10], [13], [15]. To manage complex, decentralised dynamics in human society, organisational structure is a crucial element: organisation provides a structuring and co-ordination of the processes in such a manner that a process or agent involved can function in a more adequate manner. The dynamics shown by a given organisational structure are much more dependable than in an entirely unstructured situation. It is assumed that the organisational structure itself is relatively stable, i.e., the structure may change, but the frequency and scale of change are assumed low compared to the more standard dynamics through the structure. Also in Nature several forms of organisational structure have been developed; typical examples are a beehive, the coordinated processes of organs in mammals, and the well-organised biochemistry of a living cell.

By using multi-agent organisation modelling techniques for analysis and simulation, the inherent complexity of the dynamics of multiple interacting processes within a society can be made manageable by choosing the right level of abstraction in describing them. In Nature, many phenomena have the same characteristic: they also involve complex dynamics of multiple distributed processes and their interaction.

Therefore, a natural question is whether a multi-agent-organisation modelling perspective is promising for this domain of biological complexity.

Organisations can be viewed in two ways: (1) as adaptive complex information processing systems of (boundedly) rational agents, and (2) as tools for control; central issues are [9]:

- How to identify properties of the whole, given properties of parts; from the first view: 'given a set of assumptions about (different forms of) individual behaviour, how can the aggregate properties of a system be determined (or predicted) that are generated by the repeated interaction among those individual units?'
- How to identify properties of parts, given desired or required properties of the whole; from the second view: 'given observable regularities in the behaviour of a composite system, which rules and procedures - if adopted by the individual units - induce and sustain these regularities?'

Recently a number of formal and computational modelling techniques have been developed that can be used for simulation or for formal analysis of the dynamics within a multi-agent organisation. Examples of this formalisation trend can be found in books such as [9], [13], and in a recently created journal: Computational and Mathematical Organisation Theory; e.g., [11]. For an organisation, different levels of aggregation can be identified, from single agent behaviour to the dynamics of the overall organisation. Dynamics can be described in an abstract manner by focusing on one of these levels and specifying dynamic properties for this level. Moreover, interlevel relationships between dynamic properties at different levels can be identified.

One of the organisation modelling approaches that have been developed within the agent systems area is the Agent-Group-Role (AGR) approach, introduced in [2], extended with operational semantics in [3], and with a modelling approach for dynamic properties in [4]. A related dynamic modelling framework for specification, analysis and simulation of AGR-organisation models, and supported by a software environment is described in [7]. This dynamic modelling environment allows to:

- *specify dynamic properties* for the different elements and levels of aggregation within an AGR organisation model
- *relate these dynamic properties* to each other according to the organisational structure
- use dynamic properties in *executable* form as a declarative *specification of a simulation model*
- perform *simulation experiments*
- automatically *check dynamic properties* for simulated or empirical traces

In this paper, first in Section 2, Ferber and Gutknecht's Agent-Group-Role (AGR) organisation modelling approach [2] is introduced, with an emphasis on organisational structure. It is illustrated by a model of the organisational structure of intracellular processes within *E.coli*. Section 3 addresses the dynamics of the organisation, described in terms of dynamics properties expressed in a Temporal Trace Language. In Section 4 relations between different levels of aggregation are discussed. Next, Section 5 provides some simulation results, and Section 6 concludes the paper by a discussion.

2 Organisational Structure

One of the organisation modelling approaches that have been developed within the Agent Systems area is the Agent-Group-Role (AGR) approach [2]. In this section, first a brief introduction of the AGR organisation modelling approach can be found (Section 2.1). In Section 2.2 the use of the approach is illustrated by describing the internal organisational structure of the unicellular organism *Escherichia coli* [12]. In this example, which for reasons of presentation is kept limited, the main property to focus on is growth under different environmental circumstances.

2.1 The AGR Organisation Modelling Approach

An AGR organisational structure for an overall process (or organisation) is a specification based on a definition of groups, roles and their relationships. An organisation as a whole is composed of a number of *groups*. A group structure identifies the *roles* and the *intragroup transfers* between roles. In addition, *intergroup role interactions* between roles of different groups specify the connectivity of groups within an organisation. *Agents* are allocated to roles; they realise the organisation. However, the aim of an organisation model is to abstract from any specific agent allocated. Therefore instead of particular agents, roles are used as abstract entities, defining properties agents should have when they are to function in a given role within an organisation. In Section 2.2 the AGR organisation modelling approach is illustrated for the unicellular organism *E. coli*.

2.2 Organisational structure of the living cell

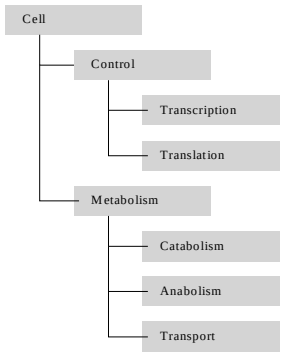

Fig. 1. Overview of the aggregation levels of the organisation model of *E.coli*.

In Figure 1 the aggregation levels of the AGR-organisation model of *E. coli* are depicted. In this picture the right hand side nodes connected to a node are called the children of the latter node, which itself is called a parent node for those children.

For example, the node Cell is the parent node of the nodes Control and Metabolism. The latter nodes are children of Cell. This means that they are the main categories or functional units that are distinguished for the processes in the cell. To be more

specific, Metabolism and Control are the main parts of the regulation and control cycle of a cell. At one aggregation level lower, the Metabolism expands to Catabolism, Anabolism and Transport. The Catabolism is the category of processes that decompose substances and extract free energy from them. In the Anabolism the processes that utilise this free energy to create more and more complex substances reside. The Transport processes move substances across the cell membrane. The Control is decomposed into Transcription and Translation. These processes generate mRNA and enzymes, respectively.

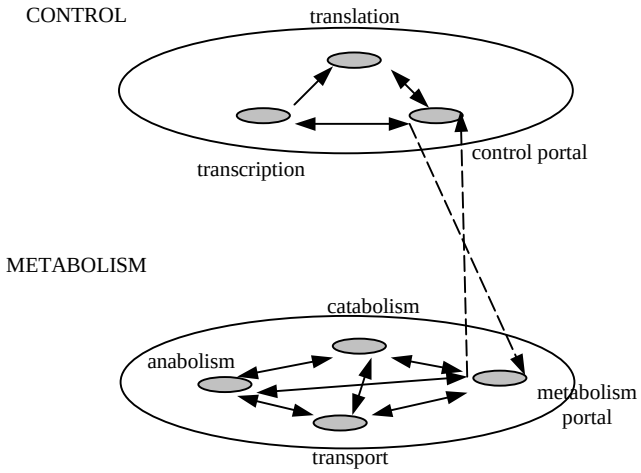

Fig. 2. *E.coli* : groups and interactions.

An AGR-model of *E.coli*'s organisational structure is shown in Figure 2. The functional units Control and Metabolism are depicted as different *groups* here (depicted by the larger ovals). Their children (according to Figure 1) are depicted in Figure 2 as *roles* (depicted by smaller ovals) within the groups. The behaviour of these roles, in the next section described by role behaviour properties, is as follows: they receive as input the presence of some substances generated by another role, in order to generate the presence of some new substances as output. The solid arrows represent *intragroup role transfers*, the transfer of substances between roles: they express that a substance produced by one role is used by another role. Notice that each group contains an additional Portal role. The idea is that these roles collect the output substances produced by all other roles within their group, to be able to interact with the other group. The dashed arrows between both portal roles represent *intergroup role interactions*, relating the input of one portal role to the output of the other. Note that the model depicted in Figure 2 is a simplification of the true living cell. For example, only control at the transcriptional and translational level is included, and 'post translational modifications' (such as phosphorylation) are left out. Nevertheless, it reflects the main aspects of its organisational structure in a way that is understandable.

3 Organisation Dynamics

The AGR organisation modelling approach was extended with a dynamic modelling approach in [4]. To characterise the dynamics within an organisation, dynamic properties of various types can be formulated. For example, a dynamic property of the organisation as a whole, such as

If oxygen, resources and some nutrients are externally available, then the cell will produce CO_2.

Other examples are dynamic properties of one specific role within an organisation, or dynamic properties that characterise how two roles cooperate.

An organisational structure provides a basis to distinguish in a systematic manner dynamic properties for different elements and aggregation levels within the organisation. In particular, as an extension of the AGR organisation model dynamic properties can be specified for each of the following aggregation levels within the model:

I. At the (highest) aggregation level of the organisation as a whole

- dynamic properties for the *organisation* as a whole; the highest aggregation level, relating any roles within the organisation over time;
- dynamic properties for *intergroup role interaction*, relating the input of one role to the output of a role in another group;

II. At the aggregation level of a group within the organisation

- dynamic properties at the level of a *group*, relating states of roles within a given group over time;
- dynamic properties for *transfer* between roles within a group (from output state of the source role to input state of the destination role);

III. At the (lowest) aggregation level of a role within a group

- dynamic properties at the level of a *role* within a group, relating input and output state (and possibly internal state) of the role;

3.1 Dynamic Properties of the Organisation as a Whole

The example model for *E. coli*'s dynamics was inspired by the model described in [5], which is based on a different modelling approach: the compositional organisation modelling approach. For the example of the living cell, global properties of the organisation as a whole can be expressed in terms of interaction with an Environment. Note that this environment is not shown in Figure 1 and 2, since we consider it not being part of the organisation itself. The cell can use as input from the environment the (external) presence of glucose, gluconate, lactose, O_2, N, P and S. It may export CO_2, ethanol and acetate to the environment. For example, CP1 in Box 1 specifies the property that if O_2 is externally available, as well as resources and at least one of the nutrients glucose, lactose, gluconate, then the cell produces CO_2. Moreover, CP2 specifies an analogue property for the anaerobic case. Note that in addition to d1, w1, also α is a variable, which makes it possible to have different instantiations of one property. For instance, property CP1(d1, w1, α) may be instantiated to CP1(0.3, 0.5, glucose). For all properties, notice that it is explicitly mentioned when interaction with the environment is involved. More specifically, if by transport a substance is emitted to the environment, this is phrased as 'exports to the Environment', and if a

substance is available for transport (i.e., import) within the environment, this is phrased as 'is present within the Environment'. In contrast, the internal exchange of the presence of substances within the organisation model is indicated by the words *generates* and *receives*. For α ranging over {glucose, lactose, gluconate}, the properties shown in Box 1 characterise the cell-environment dynamics.

CP1(d1, w1, α) CO_2 production
if within the Environment the substances α, O_2, N, P and S are present
then there exists a time point t' with $t+d1 \leq t' \leq t+w1$ such that at t'
the cell exports CO_2 to the Environment
CP2(d2, w2, α) Acetate and ethanol production
if within the Environment the substances α, N, P and S are present
and within the Environment the substance O_2 is not present
then there exists a time point t' with $t+d2 \leq t' \leq t+w2$ such that at t'
the cell exports acetate and ethanol to the Environment

Box. 1. Dynamic properties of the cell as a whole.

Within Computer Science and Artificial Intelligence a number of high-level specification languages have been developed to specify dynamic properties with mathematical precision, thereby allowing qualitative and (sometimes) quantitative aspects. To formally express the properties presented in this paper, the high-level Temporal Trace Language (TTL) has been chosen, introduced in [6], to model and analyse the internal and external dynamics of agents, and of multi-agent organisations.

A *trace* or *trajectory* in the state space is a sequence of states indexed over time. *States* are characterised by *state properties* indicating, for example, value assignments to certain variables. *Dynamic properties* are properties of traces, i.e., properties that relate states over time. To express dynamic properties the sorted predicate logic temporal trace language TTL is used. This language is built on atoms referring to, e.g., a *trace* γ, a *time point* t and a *state property* p, such as

in trace γ *at time point* t *state property* p *holds* formalised by $state(\gamma, t) |== p$.

As an example, formalising dynamic property CP1 from Box 1 in TTL yields the following:

$$\forall t\ [\ state(\gamma, t) |== in\_environment(\alpha) \wedge state(\gamma, t) |== in\_environment(O_2) \wedge state(\gamma, t) |== in\_environment(N) \wedge state(\gamma, t) |== in\_environment(P) \wedge state(\gamma, t) |== in\_environment(S) \Rightarrow \exists t'\ \ t + d1 \leq t' \leq t + w1\ \ \&\ \ state(\gamma, t') |== cell\_exports(CO_2)\ ]$$

The Temporal Trace Language TTL can play a useful role in modelling complex phenomena from an agent-oriented perspective in the following manners:

- it provides a way to obtain well-defined and mathematically formalisable *specifications* of *dynamic properties* of externally observable agent behaviour, their internal processes, and their organisation; such dynamic properties can be specified at any level of precision as desired.
- for further *analysis* it supports the identification of formalised relationships between different dynamic properties, for example between properties of an agent's externally observable behaviour and its internal processes, or between

properties of externally observable agent behaviour and properties of an organisation in which they function.

- it offers possibilities to specify and execute *simulation models* in a high level language, for example simulation of an agent's externally observable behaviour on the basis of its internal processes, or simulation of an organisation on the basis of given or assumed properties of externally observable behaviour of the agents involved.

Throughout the remainder of this paper, dynamic properties will not be formally expressed, but in the semi-formal format presented earlier, to enhance readability. Within this format, each property always holds for all traces γ over the ontology, but γ is not mentioned explicitly to keep the notation simple.

3.2 Intergroup Role Interaction Properties

Within the AGR organisation modelling approach intergroup role interaction properties model connections between groups by specifying how input state of a role in one group can be (temporally) related to output state of another role in a different group. Within the current example, the intergroup role interaction properties take care of the exchange of substances between both groups. This is done by relating the input of the portal role of one group to the output of the portal role of the other group. The properties expressing this are shown in Box 2. The delay parameters in these intergroup role interaction properties can be used to model some form of mobility of molecules produced by one process before they are used in another process. However, for simplicity we assume the exchange to be instantaneous, all delays (ci's and ri's) are 0 in this example, i.e. t' = t in the dynamic properties.

IGIP1(c1, r1) Control Portal-Metabolism Portal Intergroup Role Interaction
For all substances β,
if Control Portal receives a substance β
then there exists a time point t' with $t+c1 \leq t' \leq t+r1$ such that at t'
Metabolism Portal generates the substance β

IGIP2(c2, r2) Metabolism Portal-Control Portal Intergroup Role Interaction
For all substances β
if Metabolism Portal receives a substance β
then there exists a time point t' with $t+c2 \leq t' \leq t+r2$ such that at t'
Control Portal generates the substance β

Box. 2. Dynamic properties for Intergroup Role Interaction.

3.3 Dynamic Properties of the Metabolism and Control Group

For each of the groups, dynamic properties are considered that contribute to the properties of the organisation as a whole. A group property is specified in terms of temporal relationships between input and output states of roles within this group.

Within the group Metabolism, which includes transportation through the cell's membrane (import and export), substances present outside the cell, but also substances produced by Control can be used. Likewise, it can produce substances that are exported to the environment, as well as substances used by Control. The exchange of substances to and from Control goes via the Metabolism Portal role. Metabolism

property MP4 is an example of a complex property that has input from and output to both the environment and Control. For α ranging over {glucose, lactose, gluconate}, the dynamic properties in Box 3 characterise the Metabolism dynamics.

MP0(m_{init}) Metabolism Initialisation
there exists a time point t with $0 \leq t \leq m_{init}$ such that at t
Metabolism Portal generates the substances ATP, nucleotides and aminoacids
MP1(p1, q1) Metabolism CRPcCAMP production
if within the Environment the substance glucose is not present
then there exists a time point t' with $t+p1 \leq t' \leq t+q1$ such that at t'
Metabolism Portal receives the substance CRPcAMP
MP2(p2, q2) Metabolism allolactose production
if within the Environment the substance lactose is present
then there exists a time point t' with $t+p2 \leq t' \leq t+q2$ such that at t'
Metabolism Portal receives the substance allolactose
MP3(p3, q3) Metabolism gluconate_6P production
if within the Environment the substance gluconate is present
then there exists a time point t' with $t+p3 \leq t' \leq t+q3$ such that at t'
Metabolism Portal receives the substance gluconate_6P_observation_amount
MP4(p4, q4, α) Metabolism ATP-nucleotides-aminoacids production and CO_2 export
if within the Environment the substances α, N, P, S and O_2 are present
and Metabolism Portal generates the substances ADP, P, respiration_enzymes and import_enzymes for α
then there exists a time point t' with $t+p4 \leq t' \leq t+q4$ such that at t'
Metabolism Portal receives the substances ATP, nucleotides and aminoacids
and the cell exports CO_2 to the Environment
MP5(p5, q5, α) Metabolism ATP-nucleotides-aminoacids production/ acetate-ethanol export
if within the Environment the substances α, N, P and S are present
and within the Environment the substance O_2 is not present
and Metabolism Portal generates the substances ADP, P, fermentation_enzymes and import_enzymes for α
then there exists a time point t' with $t+p5 \leq t' \leq t+q5$ such that at t'
Metabolism Portal receives the substances ATP, nucleotides and aminoacids
and the cell exports acetate and ethanol to the Environment
MP6(p6, q6) Metabolism ArcB_P production
if within the Environment the substance O_2 is present
then there exists a time point t' with $t+p6 \leq t' \leq t+q6$ such that at t'
Metabolism Portal receives the substance ArcB_P

Box. 3. Dynamic properties for the Metabolism group.

As opposed to Metabolism, the group Control does not interact with the environment. Via its role Control Portal certain substances produced by Metabolism are available, and (abstracting from intermediate steps) it can itself produce particular enzymes, ADP, and P. In Box 4 the dynamic properties for the Control group are shown.

3.4 Transfer Properties

Transfer properties are assumed to have a generic pattern: that every transfer generated (in its output state) by any role r1 for any role r2 is received (in its input state) by role r2. In the example, for transfer properties similar assumptions are used as for intergroup role interaction properties, namely instantaneous transfer of all substances (i.e., no time durations taken into account for molecule mobility between

chemical processes; all gi's and hi's are 0). All solid arrows in Figure 2 stand for transfer properties. Because of space limitations, no transfer properties are shown in this paper.

CoP1(u1, v1) Glucose_import_enzymes production
if Control Portal generates the substances nucleotides, ATP and aminoacids
then there exists a time point t' with t+u1 ≤ t' ≤ t+v1 such that at t'
Control Portal receives the substances ADP, P and glucose_import_enzymes
CoP2(u2, v2) Respiration_enzymes production
if Control Portal generates the substances ArcB_P, nucleotides, ATP and aminoacids
then there exists a time point t' with t+u2 ≤ t' ≤ t+v2 such that at t'
Control Portal receives the substances ADP, P and respiration_enzymes
CoP3(u3, v3) Fermentation_enzymes production
if Control Portal generates the substances nucleotides, ATP and aminoacids
and Control Portal does not generate the substance ArcB_P
then there exists a time point t' with t+u3 ≤ t' ≤ t+v3 such that at t'
Control Portal receives the substances ADP, P and fermentation_enzymes
CoP4(u4, v4) Lactose_import_enzymes production
if Control Portal generates the substances allolactose, CRPcAMP, nucleotides, ATP and aminoacids
then there exists a time point t' with t+u4 ≤ t' ≤ t+v4 such that at t'
Control Portal receives the substances ADP, P and lactose_import_enzymes
CoP5(u5, v5) Gluconate_import_enzymes production
if Control Portal generates the substances gluconate_6P_observation_amount, CRPcAMP, nucleotides, ATP and aminoacids
then there exists a time point t' with t+u5 ≤ t' ≤ t+v5 such that at t'
Control Portal receives the substances ADP, P and gluconate_import_enzymes

Box. 4. Dynamic properties for the Control group.

3.5 Role Behaviour Properties

Dynamic properties for a role characterise how the role behaves, given its input. Such a dynamic property typically is expressed in terms of a temporal relationship between input state and output state of the role. For the case of *E. coli*, role behaviour properties have been specified for the roles Translation, Transcription, Anabolism, Catabolism, and Transport. Because of space limitations, these properties are not shown in this paper. However, the complete set of dynamic properties can be found at the following URL: http://www.cs.vu.nl/~tbosse/cell/.

4. Interlevel Relations

The idea of expressing dynamic properties at different levels of aggregation is that certain logical interlevel relationships can be identified between properties at the different levels. Typically, dynamics of the whole organised (multi-agent) system can be related to dynamic group properties and intergroup interaction properties via the following pattern:

dynamic properties for the groups & dynamic properties for intergroup role interaction
⇒ dynamic properties for the organisation

This implication should be understood as follows: ‘for any organisation, if for any trace the group properties and intergroup role interaction properties hold, then the general properties for the organisation also hold’. Likewise, dynamic properties of groups can be related to dynamic properties of roles in the following way:

dynamic properties for roles & dynamic properties for transfer between roles
⇒ dynamic properties for a group

The next sections will describe some interlevel relationships between dynamic properties within the example of the living cell.

4.1 Interlevel Relations for Overall Properties of the Cell Dynamics

Global property CP1(glucose) states that the cell will produce CO_2 if the substances O_2, glucose, N, P and S are available within the environment. Careful investigation of the group properties and intergroup role interaction properties yield the interlevel relationship depicted in Figure 3. The interlevel relationship between Global Property CP1(lactose) and the properties it depends on is depicted in Figure 4. This property states that the cell will produce CO_2 if the substances O_2, lactose, N, P and S are available within the environment. However, nothing is said about the availability of glucose. An argumentation of the dependencies shown could therefore be obtained by reasoning by cases: suppose all lower level properties of Figure 4 hold. Then, if glucose is present within the environment, this will be used in order to export CO_2, according to properties MP0, MP6, IGIP2, CoP1, CoP2, IGIP1, and MP4(glucose). But if glucose is not present and lactose is present within the environment, then lactose will be used, according to properties MP0, MP1, MP2, MP6, IGIP2, CoP2, CoP4, IGIP1, and MP4(lactose). Hence, if all lower level properties hold, then CO_2 will always be exported, making use of either glucose or lactose from the environment. It may thus be concluded that CP1(lactose) holds.

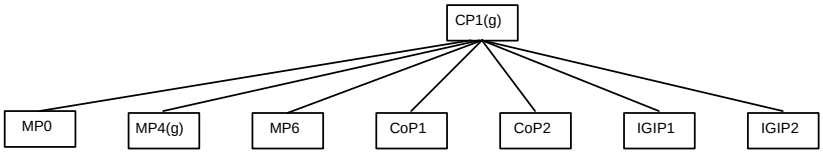

Fig. 3. Property CP1(g) related to group properties and intergroup role interaction properties.

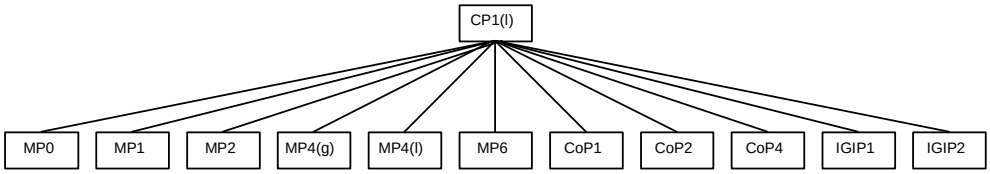

Fig. 4. Property CP1(l) related to group properties and intergroup role interaction properties.

4.2 Diagnosis based on Interlevel Relationships

The dynamic properties as presented above can be formalised in a mathematical-logical manner. Based on such a formalisation, a software environment can and actually has been developed to *automatically check* whether such properties hold for a given (empirical or simulated) trace over time for the dynamics of an organisation.

If the interlevel relationships between the dynamic properties are known, for example as depicted in Figure 4, they can be used for *diagnosis of dysfunctioning* within an organisation. For example, suppose for a given trace at some point in time it has been detected that the dynamic property CP1(glucose) at the highest aggregation level of the organisation does not hold, i.e., the cell does not produce CO_2 although the substances O_2, glucose, N, P and S are available within the environment. Given the AND-tree structure in Figure 4, at least one of the children will not hold (if they all would hold for the given trace, also CP1(glucose) would hold for this trace), which means that either MP0, MP6, IGIP2, CoP1, CoP2, IGIP1, or MP4(glucose) will not hold. Suppose by further checking it is found that MP6 does not hold. Then the diagnostic process can be continued by focusing on this property. Checking the children of property MP6 will pinpoint the cause of the organisation's dysfunctioning. Notice that this diagnostic process is economic in the sense that the whole subtree under e.g. CoP1 is not examined since there is no reason for that, as CoP1 holds.

5 Simulation and Checking

A software environment has been created to enable the simulation of executable organisation models specified at a high conceptual level. The input of this simulation environment is a set of dynamic properties. In Section 3.1 the language TTL was introduced as an expressive language for the purpose of specification and checking of dynamic properties. For the purpose of simulation, to obtain computational efficiency the format used for dynamic properties is more restricted than the TTL format used to specify various types of dynamic properties: they are in so-called 'leads to' format. This is a real time-valued variant of Executable Temporal Logic [1]. Roughly spoken, in *leads to* format the following can be expressed:

if a certain state property α holds for a certain time interval with duration g, then after some delay (between e and f) another state property β will hold for a certain time interval h

Making use of these *leads to* properties, the software environment generates simulation traces. A trace is developed by starting at time t = 0 and for each time point up to which the trace already has been constructed, checking which antecedents of executable properties hold in the already constructed trace. For these executable properties, add the consequent to the trace, i.e., extend the trace in time in such a manner that the consequent holds.

The relation between the specification and the constructed trace is that the trace is a model (in the logical sense) of the theory defined by the specification, i.e., all executable dynamic *leads to* properties of the specification hold in the trace.

5.1 Simulation

The software environment described above has been used to simulate the internal dynamics of the organisation of the cell. In order to do this, all lowest level properties have been expressed in *leads to* format. For this example, these were all intergroup role interaction properties, role behaviour properties and transfer properties.

In order to initialise the simulation, the truth values of all state properties have been set to *true* from time point 0 to 60. Furthermore, for each simulation run particular settings had to be assigned to the environment. An example situation, where lactose and resources are always present, the presence of glucose and O_2 is fluctuating, and gluconate is always absent, can be seen in Figure 5. In this trace, time is on the horizontal axis, the properties are on the vertical axis. A dark box on top of the line indicates that the property is true during that time period, and a lighter box below the line indicates that the property is false during that time period. Another part of this trace, depicting the reaction of the cell to this environment, is shown in Figure 6. Notice that the cell exports acetate, ethanol and CO_2 at the very beginning, because of the initialisation conditions. However, as it adapts to the environment only CO_2 is exported. As the environmental oxygen disappears, the cell's CO_2 emissions stop very soon, and acetate and ethanol are produced instead. After the oxygen re-appears in the environment, the cell adapts by stopping the acetate and ethanol emissions after a while and returning to CO_2 production. Note that the acetate and ethanol emissions are not stopped immediately. This is because the internal substances needed for these emissions (including fermentation enzymes) persist for some time.

An interesting observation is the fact that the fluctuating presence of glucose in the environment does not seem to have any influence on the production of CO_2, acetate and ethanol. According to the highest level properties CP1 and CP2, this is indeed the correct behaviour, since for the behaviour at this level it does not matter whether it is glucose, lactose, or gluconate, as long as one of the nutrients is available. And in this particular case, lactose is always present in the environment. Nevertheless, the fluctuating presence of glucose does influence the behaviour of the cell at a lower level. For instance, consider the next part of the same trace, depicting the output of the roles Anabolism, Catabolism, Transport, Transcription and Translation, see Figure 7.

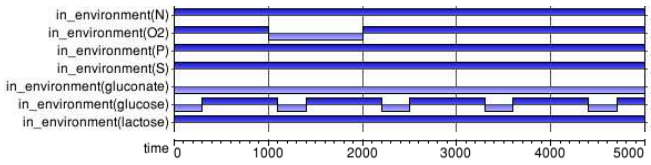

Fig. 5. Environmental Dynamics

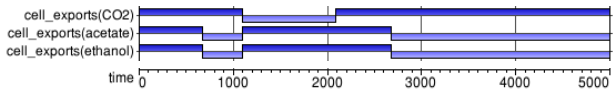

Fig. 6. Simulated overall behaviour

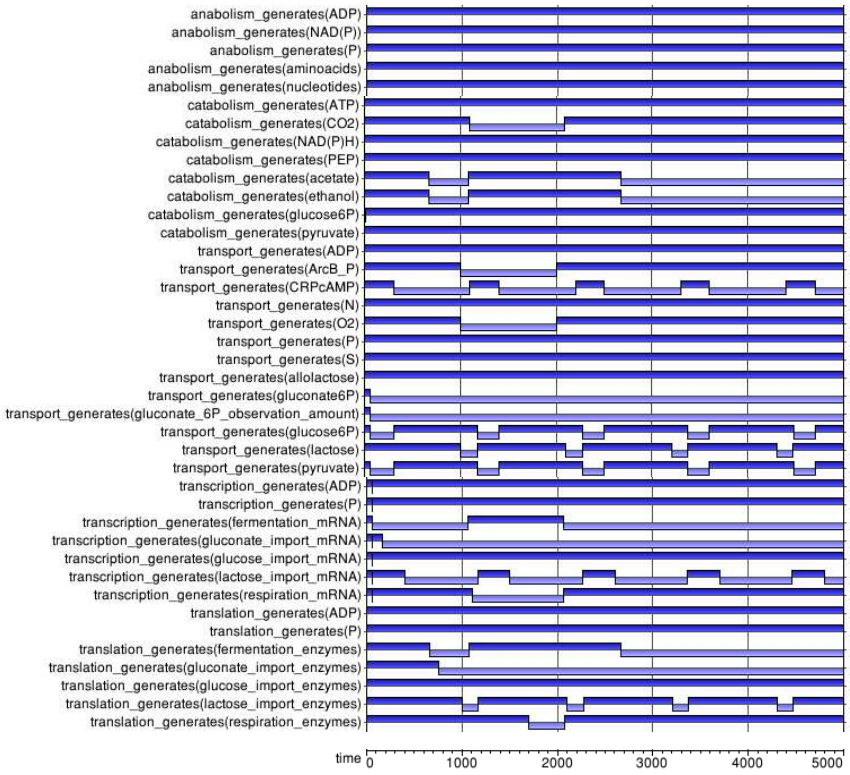

Fig. 7. Simulated internal dynamics

The specific timing parameters used for this simulation were inspired by [5]. Figure 7 shows that the presence of glucose in the environment influences, for instance, the internal production of the substance CRPcAMP by the Transport role. As a consequence, the presence of (among others) this CRPcAMP leads to the creation of lactose_import_mRNA by the Transcription role, whilst glucose_import_mRNA is created continuously. To go one step further, lactose_import_mRNA and glucose_import_mRNA are used by the Translation role to create, with a certain delay, lactose_import_enzymes and glucose_import_enzymes. It can thus be concluded that from an external perspective there is no visible difference in behaviour of the cell, whether there is only lactose outside or both lactose and glucose. Nevertheless, from an internal perspective many differences can be seen. The entire trace resulting from this simulation covers 245 state properties, representing not only the output but also the input state properties of the roles shown above. However, since the transfer of substances is instantaneous and without delay in our model, each output state property for one role results in several identical input state properties for the other roles. Likewise, the input and output state properties of the Metabolism Portal and Control Portal group are identical to state properties already shown above. Hence, for reasons of presentation, the rest of the trace is not shown in this paper.

5.2 Checking Properties

As mentioned in Section 4.2, interlevel relationships between properties, as depicted in the tree of Figure 4, can be very useful in the analysis of dynamic properties of an organisation. In order to perform such an analysis, some mechanism is needed to check if a certain property holds for a given trace. To this end, the simulation software described above automatically produces log files containing formal representations of the traces. In addition, software has been developed that is able to read in these formally represented traces together with a set of dynamic properties and to perform the checking process. As a result, the software determines not only whether the properties hold for the trace or not, but in case of failure, it also pinpoints which parts of the trace violate the properties. For our simulation, checks of this kind have actually been performed for all Global Properties and Group Properties, i.e. all properties of Section 3.1 and 3.3. They all turned out to hold for the generated traces. This validates the interlevel relationships.

6. Discussion

Analysis and simulation of biological (and in particular, cellular) processes is a huge research area in which many groups are working, e.g. [14]. As a novel contribution to this area, the current paper shows how an organisation modelling approach can be used to analyse and simulate the dynamics of biological organisation, illustrated for the functioning of intracellular processes. This biological system can be modeled as consisting of a number of active components or agents that are connected and grouped together in such a manner that everything functions well. Dynamic properties at different levels of aggregation of the organisation model have been identified, and relationships between these dynamic properties at different aggregation levels were made explicit. Based on the executable properties, simulation has been performed and (higher-level) properties have been checked for the produced simulation traces. Thus the interlevel relationships between properties at different aggregation levels have been verified. This case study shows that organisation modelling techniques can play a useful role in biological application areas.

The analysis method for the dynamics from an organisation modelling perspective involves the following ingredients:

- Specify state properties and dynamic properties of the overall process
- Identify the agents and their roles within the overall process
- Specify state properties and dynamic properties for the behaviour of these roles
- Identify groups of roles
- Specify dynamic properties for groups
- Specify dynamic intergroup role interaction and transfer properties
- Identify interlevel relations between dynamic properties at different levels of aggregation: relating role, group and organisation dynamics
- Specify executable dynamic properties
- Simulate dynamics based on executable dynamic properties
- Check given traces of dynamics against dynamic properties

Software support for some of these items within analysis has been developed or is under development. For example, an editor to specify dynamic properties according to

a specific format, and a (model) checker that checks whether dynamic properties hold in a given trace; e.g., [7].

Acknowledgements

The authors have learned a lot of this area from discussions and cooperation with Jaap Heringa, Jacky Snoep, Hans Westerhoff, Wouter Wijngaards.

References

1. Barringer, H., M. Fisher, D. Gabbay, R. Owens, and M. Reynolds (1996). The Imperative Future: Principles of Executable Temporal Logic, Research Studies Press Ltd. and John Wiley & Sons.
2. Ferber, J. and Gutknecht, O. (1998). A meta-model for the analysis and design of organisations in multi-agent systems. In: *Proceedings of the Third International Conference on Multi-Agent Systems* (ICMAS'98), IEEE Computer Society Press, pp. 128-135.
3. Ferber, J., and Gutknecht, O. (2000). Operational Semantics of a role-based agent architecture. In: Jennings, N.R. & Lesperance, Y. (eds.) *Intelligent Agents VI*, Lecture Notes in AI, vol. 1757, Springer Verlag, 2000, pp. 205-217.
4. Ferber, J., Gutknecht, O., Jonker, C.M., Müller, J.P., and Treur, J., (2001). Organization Models and Behavioural Requirements Specification for Multi-Agent Systems. In: Y. Demazeau, F. Garijo (eds.), *Multi-Agent System Organisations. Proceedings of the 10th European Workshop on Modelling Autonomous Agents in a Multi-Agent World, MAAMAW'01*, 2001. Lecture Notes in AI, Springer Verlag. To appear, 2002.
5. Jonker, C.M., Snoep, J.L., Treur, J., Westerhoff, H.V., and Wijngaards, W.C.A. (2002). The Living Cell as an Organisation: A Compositional Organisation Model of Intracellular Dynamics. Technical Report. Vrije Universiteit Amsterdam, Department of AI, 2002.
6. Jonker, C.M. and Treur, J. (1998). Compositional Verification of Multi-Agent Systems: a Formal Analysis of Pro-activeness and Reactiveness. In: W.P. de Roever, H. Langmaack, A. Pnueli (eds.), Proceedings of the International Workshop on Compositionality, COMPOS'97. LNCS, vol. 1536, Springer Verlag, 1998, pp. 350-380. Extended version in: *International Journal of Cooperative Information Systems*, vol. 11, 2002, pp. 51-92.
7. Jonker, C.M., Treur, J., and Wijngaards, W.C.A. (2002). Temporal Languages for Simulation and Analysis of the Dynamics Within an Organisation. In: B. Dunin-Keplicz and E. Nawarecki (eds.), *From Theory to Practice in Multi-Agent Systems, Proc. of the Second International Workshop of Central and Eastern Europe on Multi-Agent Systems, CEEMAS'01*, 2001. Lecture Notes in AI, vol. 2296, Springer Verlag, 2002, pp. 151-160.
8. Kreitner, R., and Kunicki, A. (2001). Organisational Behavior, McGraw – Hill.
9. Lomi, A., and Larsen, E.R. (2001). Dynamics of Organizations: Computational Modeling and Organization Theories, AAAI Press, Menlo Park. Manna, Z., and Pnueli, A. (1995). *Temporal Verification of Reactive Systems: Safety*. Springer Verlag.
10. Mintzberg, H. (1979). The Structuring of Organisations, Prentice Hall, Englewd Cliffs, N.J.
11. Moss, S., Gaylard, H., Wallis, S., and Edmonds, B. (1998). SDML: A Multi-Agent Language for Organizational Modelling, *Comput. and Mathem. Org. Theory 4*, (1), 43-70.
12. Neidhardt, F.C., Curtiss III, R., Ingraham, J.L., Lin, E.C.C., Brooks Low, K., Magasanik, B., Reznikoff, W.S., Riley, M., Schaechter, M., and Umbarger, H.E., eds. (1996). *Escherichia coli* and *Salmonella typhimurium*. ASM Press, Washington, D.C.
13. Prietula, M., Gasser, L., Carley, K. (1997). Simulating Organizations. MIT Press.
14. Takahashi, K. (2004). E-Cell: A Multi-Algorithm, Multi-Timescale Simulation Software Environment. URL: http://ecell.sourceforge.net/.
15. Weiss, G. (ed.) (1999). *Multiagent Systems*. MIT Press