Modelling the Dynamics of Organisational Change

Catholijn M. Jonker
Delft University of Technology

Jan Treur
Vrije Universiteit Amsterdam

Modelling the Dynamics of Organisational Change

Catholijn M. Jonker[1] , Martijn C. Schut[1], and Jan Treur[1,2]

[1]Vrije Universiteit Amsterdam, Department of Artificial Intelligence
De Boelelaan 1081a, NL-1081 HV Amsterdam, The Netherlands
Email: {jonker, schut, treur}@cs.vu.nl
URL: http://www.cs.vu.nl/~{jonker, schut, treur}
[2]Utrecht University, Department of Philosophy
Heidelberglaan 8, 3584 CS Utrecht

Abstract For an information-agent-based system to support virtual (Internet-supported) organisations, changes in environmental conditions often demand changes in organisational behaviour, i.e., organisational changes. As organisational behaviour relates to organisational structure, rethinking the structure of an organisation as environmental conditions demand so, renders such an organisation imperatively flexible and adaptive. This paper presents a formal modelling approach for such organisational dynamics. The contributions of the reported research are (1) a formal model as a basis for simulation (2) formal specifications of dynamic properties of organisational behaviour at different aggregation levels, (3) logical interlevel relationships between these properties, (4) a prototype implementation, and (5) simulation and checking results.

1 Introduction

Within the field of Organisation Theory organisational structures regulating societal dynamics, and thus entailing organisational behaviour are studied; e.g., [12], [19], [22]. A particular area for which this field has become quite relevant is the area of virtual (Internet-supported) organisations. Supporting the design of virtual organisations based on information agents on the Internet asks for a dedicated organisation modelling approach. Within the area of Computational Organisation Theory and Artificial Intelligence, a number of organisation modelling approaches have been developed to simulate and analyse dynamics within organisations; e.g., [24], [20], [23], [7]. Some of these approaches explicitly focus on modelling organisational structure, abstracting from the detailed dynamics. Other approaches put less emphasis on organisational structure but focus on the dynamics in the sense of implementing and experimenting with simulation models. The Agent/Group/Role (AGR) approach as described in [7] is an example of an approach focussing on organisational structure, abstracting from the details of the dynamics. However, [8], [9] are some first steps to specify the organisational behaviour by adding specifications of dynamic properties to the organisational structure provided by AGR.

In [15] it is shown how the relation between an organisational structure and organisational behaviour can be founded formally. In [13] practical application of the connection between structure and behaviour is presented in the form of an approach that enables to derive organisation properties from role properties (and interaction properties), according to the organisational structure. However, in these papers it is not addressed how to model *organisational change*, i.e., an organisation that is changing its behaviour over time, a phenomenon that receives much attention in recent literature on Organisation Theory; e.g., [19], [20]. Organisational change is a process that allows an organisation to adapt its behaviour to changing environmental conditions. In virtual organisations such changes occur on a regular basis, and in fact may be part of the normal functioning of such an (evolving) organisation.

The initiative for changes of organisational structure usually lies within the organisation (in interaction with the environment). In organisations in human society, often the underlying decision process is embedded within the organisation in the form, e.g., of a director or management board supported by a strategic management department. In this sense, the process to obtain a changed organisation is itself part of the organised dynamics. This makes the organisational dynamics a reflective process.

To model the dynamics of this reflective process in order to support the evolution of virtual, information agent-based organisations is the challenge addressed in this paper. In Section 2 an overview of the modelling approach is presented. Section 3 describes an example scenario and the AGR organisation modelling approach. In Section 4 the representations used in the model are presented. Section 5 provides more detail about the manner in which the strategic management for organisational change is modelled. In Section 6 the prototype implementation is described, and in Section 7 some of the simulation results are shown. Finally, in Section 8 it is discussed how the dynamics of organisational change can be characterised and checked by formally specified dynamic properties.

2 The Modelling Approach

Processes of organisational change (in succession) realise an evolving organisation that is able to adapt to its (changing) environment; e.g., [19], [20]. From a strategic management perspective, this evolution is to be guided by deliberation. Based on such deliberation the organisational changes are initiated and accomplished. The deliberation may involve (the demands from) the environment, organisational goals, organisational structure, allocation of agents, and actions to achieve goals.

In the approach presented in this paper, to obtain a changed organisational structure is a means to achieve a goal. Such a goal concerns (dynamic) properties of the organisational behaviour. So, the deliberation starts by identifying the goals in terms of the required changes in organisational behaviour, given changed environmental conditions; see the right hand side in Figure 1.

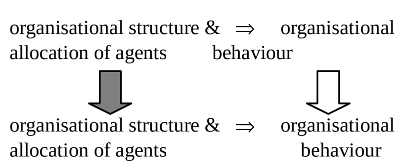

Figure 1 Organisational change as second-order behaviour

Next, using relations (the horizontal ⇒ arrows) between organisational behaviour and organisational structure and allocation of agents, it is determined which changes in organisational structure and allocations of agents (the left hand side of Figure 1) are needed to achieve the required changes in organisational behaviour. Finally, the changes in organisational structure and allocation of agents are actually made. Thus the dynamics of organisational change as a form of adaptive behaviour can be viewed as second-order dynamics, i.e., dynamics of dynamics: what is achieved is a change in organisational behaviour; see the right downward arrow in Figure 1. This behavioural change is realised by means of a change in organisation structure and allocation of agents: the left downward arrow.

To model such a process, models are needed for deliberation and modification. Moreover, a number of explicit representations play a role: for organisational behaviour, organisational structure and relations between behaviour and structure. In the adopted modelling perspective two models play an important role. In the first place a goal-directed agent model is used to model the deliberative aspects of the organisational change, and in the second place a model for modification of the organisation structure and agent allocation. The deliberation model of the organisation-as-deliberative-agent is embedded within the organisation structure. The following representations are used:

- representation of monitored and desired *dynamic properties* (requirements) of the *organisational behaviour*.
- representation of *organisation structures*
- *relationships* between specific requirements on organisational behaviour and specific organisational structures

These representations are discussed in more detail in Section 4.

3 AGR and Example Scenario

The Agent-Group-Role (AGR) organisation modelling approach [7] abstracts from the behaviour of individual agents by considering organisational concepts as roles, groups and structures as first class citizens. As such, an agent is considered an active communicating entity which plays roles within groups. Groups are aggregated sets of agents and a role is an abstract representation of an agent function within a group. Organisational structure is considered at group and organisation level. On group level, a structure is an abstract description of a group, identifying all roles and interactions within the group. An organisational structure is a set of group structures together with intergroup interactions. The actual realisation of the organisation then constitutes the allocation of a set of agents to roles. The original aim of the AGR organisation modelling approach concentrates on organisation structure, while our concern is more with the actual dynamics. To incorporate the structure within the logical description of the internal dynamics of the organisation: firstly, for each sort in the organisational structure, a specific set of dynamic properties is introduced; secondly, relationships are identified between these sets, based on the structural relations between sorts; for further details on this formalisation, see [14].

The approach is illustrated by a toy example: a banking case study. Clients put in requests, e.g., opening an account or withdrawing funds from an account, to the bank through a central call center. The call center forwards such a request to a bank in the region where the client resides. If a local bank then has fulfilled the request, the client is informed by the call center that the request is finished. The organisational structure is shown in Figure 2, where groups are indicated by bigger boxes and roles by small boxes; arrows denote interactions. The first part is the actual organisation on the left hand side. The clients requests come in via the Client Service group, from where they are forwarded to representatives of local banks. Client requests are allocated to local banks on the basis of the regions in which clients live. In local banks employees fulfill these requests. Interaction is bidirectional: requests go upstream from the clients to the employees and the different possible stati of these requests (finished, accepted, rejected) go downstream.

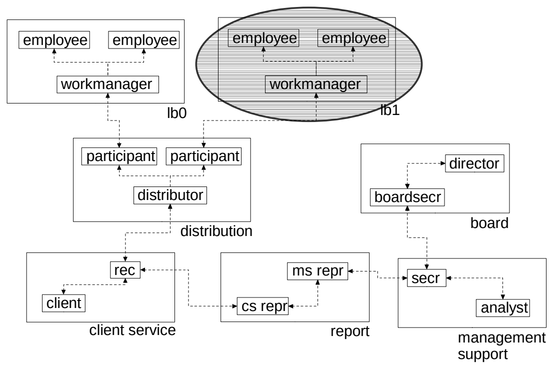

Figure 2 Organisational AGR structure of the case study

By a simple example the organisational change considered can be illustrated as follows. Assume that at some moment, the Rabobank has a single local bank called lb0 in region reg0, but dealing with client requests from reg0 and reg1. As the number of client requests increases over time, management may decide to open up another local bank, called lb1. After this reorganisation, lb0 deals with requests from region reg0 and lb1 deals with requests from region reg1. The right hand side is the place where analysis and deliberation on organisational change, as discussed in this paper, takes place. The Report group communicates statistics on the incoming requests, to the Management Support group. Based on this information within this group new organisational goals and actions are generated, upon which final decisions are made by the Director.

4 Representations Used

Explicit *representations of organisational structure* for the AGR-modelling approach are based on the many-sorted logic formalisation presented in 14. In Table 1 part of an explicit representation of the example described in Section 3 is depicted. Here, e.g., t1 denotes the transfer within the Client Service group from Client to Receptionist, and, e.g., i1 denotes intergroup interaction from Receptionist to Client Service representative.

Client Service group	**Report group**
role_in(client, clientservice)	role_in(cs_repr, report)
role_in(rec, clientservice)	role_in(ms_repr, report)
source_of_transfer(client, t1)	source_of_transfer(cs_repr, t3)
destination_of_transfer(rec, t1)	destination_of_transfer(ms_repr, t3)
source_of_transfer(rec, t2)	source_of_transfer(ms_repr, t4)
destination_of_transfer(client, t2)	destination_of_transfer(cs_repr, t4)

Management Support group	Intergroup interactions
role_in(secr, managementsupport) role_in(analyst, managementsupport) source_of_transfer(secr, t5) destination_of_transfer(analyst, t5) source_of_transfer(analyst, t6) destination_of_transfer(secr, t6)	source_of_interaction(rec, i1) destination_of_interaction(cs_repr, i1) source_of_interaction(cs_repr, i2) destination_of_interaction(rec, i2) source_of_interaction(ms_repr, i3) destination_of_interaction(secr, i3) source_of_interaction(secr, i4) destination_of_interaction(ms_repr, i4)

Table 1 Organisational structure representation

Representations of organisational behaviour are used both to describe existing (possibly undesired) behaviour as has been monitored, and to describe desired or required behaviour that can be taken as goal. A formalisation is needed of the dynamic properties that characterise organisational behaviour. To this end the formal specification language for dynamic properties of organisational behaviour as presented in [14] is adopted. An example of such a representation is:

∀ tid : TaskId, ∀ t1, tf : T
∀ C: CLIENT:client_service∀ R: RECEPTIONIST: client_service
[[t ≤ t1 ≤ t' &
state(T , t1, output(C)) |= comm_from_to(requested(tid, tf), C, R) ⇒
∃ t2 : T [t1 ≤ t2 ≤ t1 + d &
[state(T , t2, input(C)) |= comm_from_to(rejected(tid), R, C) ∨
state(T , t2, input(C)) |= comm_from_to(accepted(tid), R, C)]]

This organisation behaviour property specifies that for trace T, at any point in time between t and t', if a client communicates a request to the receptionist, then within time duration d the receptionist will communicate either an acceptance or a rejection of the request to this client. Using the abstraction mechanism introduced in [17] such properties can be expressed in a conceptual high-level (but yet formal) language, without having to involve all (temporal) details. For example the property above can be represented in an abstract manner by

has_maximal_request_response_time(T , t, t', d)

This abstraction mechanism allows the deliberation model to work with relatively simple expressions and relations between them.

5 Strategic Management

Strategic management involves the deliberation process that takes into account (changed) environmental conditions, goals on the organisational behaviour, and organisational structure which is the vehicle to realise behavioural goals. In this section it is discussed how such a deliberation process takes place (Section 5.1) and how such a process is embedded within the organisational structure (Section 5.2).

5.1 Organisational Behaviour and Structure

The deliberation process is explained by means of an example scenario. Suppose in the period between t and t' a substantial increase in number of clients in a certain region occurs (for example due to an extension of a city). Based on monitor information this environmental change is detected. Moreover, as part of the analysis process it is found out that for the given organisation structure, the maximal response time has increased, i.e. the dynamic property has_maximal_request_response_time(T , t, t', d') holds, where d' is larger than before: d'>d. As part of the deliberation process on goals on the future organisational behaviour, this d' is considered much too high. Moreover, if an organisational change is performed, then this is considered as an opportunity to get the maximal response time even less than before: a number d"<d is determined such that a goal is that has_maximal_request_response_time(T , t', t", d") will hold. The new goal is this; without repeating this all the time, also the goal that the permanent costs per client are limited to some bound b will be imposed. As part of the deliberation process about organisational structure modification actions it is found out that this new combined goal will be achieved if the organisational structure is modified in the sense that a new location of the bank is set up for the region. As a consequence of this deliberation it is decided to actually make the modification to the organisation structure. Within the deliberation process as sketched, supporting techniques and tools can be used. First, within the monitor and analysis process, on the basis of a trace, dynamic properties of the current organisational behaviour and environment can be checked in a formal and automated manner by the approach described in [13]. Moreover, in deliberation about an organisation structure modification action it can be determined what the behavioural properties for a possible organisational structure are, also in a formal and automatically supported manner as in [13].

5.2 Embedding of the Deliberation Model within the Organisation Model

The deliberation process consists of an analysis of the recent past (say of the last 3 months) of organisational behaviour, and a planning process to determine a goal and a modification action that will influence the organisation behaviour in the future. Both determination of a goal and determination of an action are performed by first generating one or more options and then select one (possibly with a slight change).

For the example organisation in the example domain the Management Support group is responsible for analysing the monitoring results, for suggesting goals for the future behaviour of the organisation, and for proposing modification actions to the organisation structure to satisfy certain goals. The Board is responsible for deciding on the goals and modification actions. Below, properties of groups will be indicated by GP and properties describing intergroup interaction by GI.

Proper transfer of information is important throughout the deliberation process and throughout the organisation. For all roles R1 and R2 that have to communicate with each other, a transfer property has been formulated:

TP(R1, R2, m) ≡ If R1 communicates m to R2 then R2 receives m.

If this holds for any m, this is denoted by TP(R1, R2), and two-way transfer by TTP(R1, R2).

Monitoring

The monitoring requires that response time report is generated on a regular basis. The following properties show where in the organisation this is performed and how it can be specified that the monitoring results arrive timely at the appropriate places within the organisation.

Client Service group

GP1 ≡ If in a certain week the Clients generate a number of requests, then the Receptionist will receive these requests in this week.

Client Service – Report group interaction

GI1 ≡ If in a certain week the Receptionist in the Client Service group receives a number of requests, then the Client Service representative in the Report group will communicate the weekly response time report to the Management Support group representative within the Report group.

Report group property
GP2 is the transfer property TP(CSrepr, MSrepr, weekreport)
Report - Management Support group interaction
GI2 ≡ If the Management Support representative in the Report group receives the weekly response time report, then the Secretary of the Management Support group communicates this report to the Analyst of the Management Support group.

Goal Determination
The goal determination process in our example is a joint responsibility of the Management Support group (generating options for goals) and the Board (selecting a goal). The following properties show a specification of the shared responsibilities, from receipt of monitoring information to formulation of new goals.

Management Support group
GP3 ≡ If the Secretary within the Management Support group communicates a weekly response time report to the Analyst in which the maximal response time is unacceptable given the organisation's directions, and in the previous week report this was not the case, then the Secretary will receive one or more goal proposals from the Analyst.

Management Support – Board group interaction
GI3 ≡ If the Secretary of the Management Support group receives goal proposals from the Analyst, then the Secretary of the Board communicates these goal proposals to the Director.

Board Group
GP5 ≡ If the Secretary of the Board communicates goals proposed by the analyst to the Director, then the Secretary of the Board will receive a (possibly slightly altered) selected goal from the Director.

Board – Management Support group interaction
GI4 ≡ If the Secretary of the Board receives a goal from the Director, then the Secretary of the Management Support group communicates this goal to the Analyst.

Modification Action Determination
After a decision has been made as to what the goal for the organisational behaviour is, it is determined which modification action is used to satisfy that goal. In the example organisation that responsibility is shared by the Management Support group (generating options) and the Board (selecting an action).

Management Support group
GP4 ≡ If the Secretary of the Management Support group communicates a director's goal to the Analyst, then the Secretary will receive from the Analyst one or more modification action proposals to satisfy this goal.

Management Support – Board group interaction
GI5 ≡ If the Secretary of the Management Support group receives a modification action proposal from the Analyst, then the Secretary of the Board communicates the proposal to the Director.

Board group
GP6 ≡ If the Secretary of the Board communicates a modification action proposal (from the analyst), then the Director performs a (possibly slightly altered) modification action.

Relationships for Group Properties
Within each group, the group properties can be related to transfer properties and role behaviour properties (RP) as follows:

TTP(Client, Receptionist)	⇒ GP1
TP(CSrepr, MSrepr, weekreport)	⇒ GP2
TTP(Secr, Analyst) & RP(Analyst, goal)	⇒ GP3
TTP(Secr, Analyst) & RP(Analyst, action)	⇒ GP4
TTP(Boardsecr, Director) & RP(Director, goal)	⇒ GP5
TTP(Boardsecr, Director) & RP(Director, action)	⇒ GP6

6 Prototype and Simulation

A prototype has been developed for validating the ideas introduced in this paper. This prototype involves temporal formalisation of the dynamics, part of which was described in Section 3, specified as an executable organisation model in 'leads to' format; cf. [18], and executed using the available software environment. The prototype includes some simplifications, without compromising the main ideas underlying the model presented in the paper. For example, the prototype does not include weekly response time reports. Instead, requests are communicated individually by the Receptionist to the Secretary of the Management Support group; subsequently, the Analyst accumulates the number of incoming requests and, if applicable, generates a corresponding goal proposal, e.g., to decrease the response time.

Some simulation results are shown below. For simplicity's sake the trace has been kept limited in size. Although the prototype includes a goal to decrease the response time, the effect of undertaking the action (adding a bank) to fulfill this goal is not fully demonstrated in the trace, due to the fact that only a few requests are shown. However, the trace does demonstrate the effect of the reorganisation as such that after the reorganisation client requests are being dealt with the local bank in the region where the client lives. This indicates that when more requests are received from the new region, the actual response time will indeed decrease.

Tables 2 and 3 show summaries of results of experimenting with the prototype executable organisation model, as described in the previous Section. The starting point of the depicted trace is at the point where the organisation receives its last request before the criterion for organisational change is reached. The Tables are derived from one and the same organisational trace1, albeit each of them demonstrates a different perspective on the trace.

Table 2 considers the trace from the strategic management point of view. The events in this Table concern the events and communications that take place in the Report, Management Support, and Board groups. The Table demonstrates that after the second request has been received, the Analyst generates a goal proposal to decrease the response time. This goal proposal is communicated through the organisation to the Director, who approves this proposal and adopts the actual goal. This goal is communicated through the organisation to the Analyst. The Analyst generates an action proposal to add a bank which will satisfy the goal as set by the Director. The action proposal is communicated through to the Director, who approves the action proposal and performs the actual action to add a bank. This part of the trace follows the deliberation process as described in Section 5.2, taking into consideration the simplifications of the prototype mentioned in Section 6.

Time req0	Time req1	Event
60	80	req is communicated from csrepr to msrepr
61	81	req is communicated from secr to analyst
		number of reqs is being computed by analyst
		analyst generates goal proposal g to decrease response time
	117	goal proposal g is communicated from analyst to secr
	119	goal proposal g is communicated from boardsecr to dir
		dir approves this proposal g
	125	goal g is communicated from dir to boardsecr
	135	goal g is communicated from secr to analyst
		analyst generates action proposal a to add a bank
	140	action proposal a is communicated from analyst to secr
	150	action proposal a is communicated from boardsecr to dir
		dir approves this proposal a
	155	dir performs action a to add a bank

Table 2 Summary of the trace for Report, Management Support and Board groups until reorganisation[1]

Table 3 considers the trace from the client point of view, and involves the Client Service, Distribution and local bank (lb0, lb1) groups. Here, the requests of two clients from different regions are being handled by the same bank before the reorganisation, whilst their subsequent requests are handled by different banks after the reorganisation.

Time	Event
15	cl0 (lives in reg0) puts in request req0
35	cl1 (lives in reg1) puts in request req1
92	cl0's request is handled by employee plb01 (bank lb0 serves reg0)
102	cl1's request is handled by employee plb01 (bank lb0 serves reg1)
145	cl0 receives acknowledgement that his request has been executed
160	cl1 receives acknowledgement that her request has been executed
215	cl0 (lives in reg0) puts in request req2
235	cl1 (lives in reg1) puts in request req3
292	cl0's request is handled by employee plb01 (bank lb0 serves reg0)
302	cl1's request is handled by employee plb11 (bank lb1 serves reg1)
345	cl0 receives acknowledgement that his request has been executed
360	cl1 receives acknowledgement that her request has been executed

Table 3 Summary of the trace for Client Service, Distribution, lb0 and lb1 groups[1]

1 For the complete trace as generated by the developed software, see http://www.cs.vu.nl/~schut/tracecia.png.

7 Analysing the Organisation's Adaptive Behaviour

An organisation that is able to change its behaviour based on policy and environmental changes shows some form of adaptive behaviour. Dynamic properties can be used to characterise not only current organisational behaviour (the organisation's first-order dynamics), but also adaptive organisational behaviour over time (the organisation's second-order dynamics).

7.1 Properties for the Adaptive Organisation

To illustrate how this can be achieved, for the example scenario, its second-order dynamics will be characterised. The idea is that every time that an increase in number of clients occurs, and as a consequence an increase in maximal response time, the organisation responds with a change in its behaviour to get the maximal response time back under a certain bound, whereas permanent costs per client are also kept below a given bound. This global pattern can be specified in semiformal form as follows.

OP0

for all t
if the number of clients from t to t + e is increasing
and the maximal response time between t - e and t is at most d
and the maximal response time between t and t + e is higher than d
then the maximal response time between t + e + f and t + 2e + f is at most d.

Here e and f are duration parameters: e the interval on which behavioural performance is measured; f is the delay parameter: the organisational change has a measurable effect with delay at most f after the effects of environmental change were measurable. This dynamic property can be formalised at an abstract level by the abstract dynamic property adaptation($\mathcal{T}$, d, d', e, f) which is definable as

for all time points t
increasing_number_of_clients($\mathcal{T}$, t, t+e) &
has_maximal_request_response_time($\mathcal{T}$, t-e, t, d) &
has_maximal_request_response_time($\mathcal{T}$, t, t+e, d')
$\Rightarrow$ has_maximal_request_response_time($\mathcal{T}$, t+e+f, t+2e+f, d)

Such adaptive dynamic properties can be checked automatically (and actually have been for the trace described above) for a given trace or set of traces, using the approach described in [13].

7.2 Relating Organisation Properties

This section shows how dynamic properties at different levels of aggregation of the organisation can be related to each other.

Additional Global Organisation Properties

OP1 ≡ If the Clients in the Client Service group generate an increasing number of requests then after some time the Director in the Board performs an appropriate modification action.

OP2 ≡ If the Director in the board performs an appropriate modification action then after some time the average response time drops below the desired response time limit.

Properties for Parts of the Organisation

For the three main phases of the deliberation process a number of milestone properties have been identified: OP3 for monitoring, OP4 for goal determination, and OP5 for modification action determination.

OP3 ≡ If in a certain week the Clients generate a number of requests, then the Management Support representative in the Report group will receive the weekly response time report from the Client Service representative within the Report group.

OP4 ≡ If the Secretary within the Management Support group communi-cates a weekly response time report to the Analyst in which the maximal response time is unacceptable given the organisation's directions, and in the previous week report this was not the case, then the Secretary of the Board will receive a selected goal from the Director.

OP5 ≡ If the Secretary of the Management Support group communicates a director's goal to the Analyst, then the Director within the Board will select and perform a modification action.

The relationships between these dynamic properties at different levels are as follows:

OP1 & OP2	⇒ OP0
OP3 & OP4 & OP5 & GI2 & GI 4	⇒ OP1
GP1 & GP2 & GI1	⇒ OP3
GP3 & GP5 & GI3	⇒ OP4
GP4 & GP6 & GI5	⇒ OP5

These relationships between dynamics at different levels of aggregation can be depicted in an AND-tree (Figure 3); they can be combined with the relationships for group properties specified at the end of Section 5.

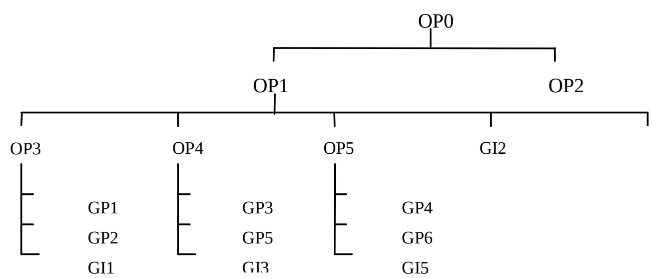

Figure 3 Interlevel Relationships for the Adaptive Dynamics

8 Discussion

The contribution of this paper is a rather complex model, designed as a specific type of composition of diverse ingredients. Due to this complexity, a rather simple (toy) example case study was used to illustrate the model. Moreover, given the model as a composition of various ingredients, one or more of the ingredients can be replaced by others without changing the overall model in an essential manner. In this discussion, first the diverse ingredients as adopted are reviewed. Next some further work for the future is discussed.

8.1 Ingredients Used

The choices made for the various ingredients used in the model are discussed. Within the overall model, for most of the ingredients alternative choices could have been made. Some of these alternatives are briefly discussed.

- *modelling approach for organisational structure*

First of all a modelling approach for organisational structure is needed that is used at two levels: the level of the overall (simulation) model, and within this model as an explicit (reflective) representation of the model itself. The choice made for the organisational structure is the AGR-modelling approach [7]. Alternative choices would well be possible, as long as they are defined in a precise manner; e.g., [11].

- *representation format for organisational structure*

To be able to use representations of an organisation structure in a reflective manner within the model, the chosen organisation structure modelling approach needs to be formalised and reified. Usually such modelling approaches are based on pictoral, graph-like forms, i.e.,consisting of different types of relations between nodes. It is well possible to formalise such forms in a (many-sorted) predicate logic format, and using the atoms of such a description as terms (meta-description), where the predicate symbols become function symbols; in [15] this has been shown. This approach was adopted to obtain explicit representations for organisational structures. Alternative choices may be obtained by using description languages; e.g., [2]. Also formalisations based on different organisational modelling approaches such as [1], [11] would be alternative options.

- *modelling approach for organisational behaviour*

As for organisational structure, a modelling approach is needed to express dynamic properties on organisation behaviour that is used at two levels: the level of (adaptive) properties of the overall (simulation) model, and within this model as an explicit (reflective) representation to be able to formulate and deliberate about goals on organisation behaviour. As organisation behaviour depends on organisational structure, the choice for this approach needs to be coherent with the choice for the modelling approach for organisational structure. The choice was made to adopt the approach described in [9], related to the AGR organisational modelling approach. An alternative choice could have been the one put forward in [8], but this is technically more complicated. Other choices for behavioural modelling could have been made (e.g., if another choice for the organisational structure modelling approach was made); e.g., [1], [5], see also next paragraph. But these would still need to be related to the chosen specific organisational structure.

- *representation format for organisational behaviour*

As for organisational structure, to be able to use representations of dynamic properties of organisational behaviour in a reflective manner within the model, the chosen organisational behaviour modelling approach needs to be formalised and reified. Given a modelling approach for organisation behaviour, a formal representation language has to be chosen. The organisational modelling variant of the Temporal Trace Language (TTL, cf. [14]) was chosen together with the abstraction mechanism described in [17]. Other choices could be [1] or [4]. Yet other alternative choices could be based on standard forms of temporal logic, such as in [3], but serious drawbacks of these languages are that no variables and number can be used. Some other temporal logical languages are able to incorporate numbers and real-time; for example, [25]. However, these languages are not suitable for comparative properties, i.e., properties that compare histories.

- *relations between organisational structure and behaviour*

For a formalisation of the relationships between organisational structure and behaviour, the foundational account presented in [14] was used as a basis. Formalised foundations of the relationship between organisational structure and behaviour seems not a well-developed area; it is not clear yet what alternative choices could possibly be.

- *checking techniques and automated support*

Checking dynamic properties against traces is exploited both internally in the model to check current behaviour, and externally to check the adaptive behaviour of the model as a whole. The choice was made to use the - automated - approach described in [13]; this fits well to the chosen format of the dynamic properties. If this format is chosen differently, for example, based on some temporal logic, then also theorem provers and/or model checkers for temporal logic are a possibility here.

- *agent model for strategic management*

The agent model for strategic management (a quite simple goal-directed agent model, based on a form of means-end analysis) was chosen. As an alternative, a much more sophisticated agent model can be chosen, for example taking into account cooperation with other organisations and commitments.

- *simulation environment for organisation models*

For simulation the decision was made to use the 'leads to' format described in [18] as a conceptual-level specification format, and the related software environment to execute the simulations. An alternative choice would be to use Executable Temporal Logic [3] as a specification format for executable models, and the Concurrent MetateM [10] software environment to execute these specifications. Other alternative choices could be based on variants of transition systems, for example, [1].

- *transitions between deliberation and actual modification*

The model to describe the transitions between deliberation and actual execution of modification actions has been based on the conceptual-level mind-matter interaction model described in [16]. An alternative for this choice would be to leave these transitions out of the conceptual description, and just add them as 'implementation details' in the implementation code. This would not be a satisfactory choice. Given the reflective aspects, other possible choices could be destilled from the literature on reflective and meta-level architectures (e.g., [21]), although for each of the choices work needs to be done to (1) adapt it to the context of organisation models, and (2) to obtain a neat conceptual-level description.

8.2 Further Work

In [4] and [6] modification of a multi-agent system was addressed, both from the angle of simulation (the former paper) and analysis (the latter paper). Differences between these papers and the current paper are as follows. First of all, in these two papers no organisational structure is used within the multi-agent system, what is the focus of the current paper. As a consequence, what is modified in [4] and [6] is the multi-agent system, extending it by one new agent, not an organisational structure as in the current paper. Second, in these papers the deliberation about modification of the multi-agent system is performed within one of the agents, by a specific design agent including a task model for redesign, it is not distributed over the organisation and integrated within the organisation model. Furthermore, in [4] no explicit formalised representations and formal analysis of behaviour are used, and in [6] no simulation is performed.

In the current paper, in the context of organisational structure both are addressed in an integrated manner.
More work will be done to identify different types of adaptive properties. For example, in the current context the following property comparing the histories in two different traces can be formulated semi-formally:

if before time t in trace $\mathcal{T}2$ more often organisational change has occurred than in trace $\mathcal{T}1$,
then at t an organisational change will take less time in trace $\mathcal{T}2$ than in trace $\mathcal{T}1$.

This is an example of a relative adaptive property of a learning organisation. This type of property can be formalised in TTL as well. A variant of this comparative dynamic property is that changes are not (only) made in a faster manner, but (also) of higher quality, i.e., changes are initiated in time, but not too early. Also such a property can be expressed in TTL.

Acknowledgements

Lourens van der Mey and Wouter Wijngaards have contributed to the development of the software environment.

References

1. Artikis, A., Sergot, M., and Pitt, J., Specifying Electronic Societies with the Causal Calculator. Proc. of the 3d Int. W. on Agent-Oriented Software Engineering, AOSE'02. Lecture Notes in Computer Science, Springer Verlag, to appear.
2. Baader, F., D. Calvanese, D.L. McGuinness, D. Nardi, P.F. Patel-Schneider (eds.), Description Logic Handbook, Cambridge University Press, 2002
3. Barringer, H., Fisher, M., Gabbay, D., Owens, R., and Reynolds, M., (1996). The Imperative Future: Principles of Executable Temporal Logic, Research Studies Press Ltd. and John Wiley & Sons.
4. Brazier, F.M.T., Jonker, C.M., Treur, J., and Wijngaards, N.J.E., Deliberative Evolution in Multi-Agent Systems. International Journal of Software Engineering and Knowledge Engineering, vol. 11, 2001, pp. 559-581.
5. Carron, T., Boissier, O. (2001). Towards a Temporal Organizational Structure Language for Dynamic Multi-Agent Systems. In: Y. Demazeau, F. Garijo (eds.), Multi-Agent System Organisations. Proceedings of MAAMAW'01, 2001.
6. Dastani, M., Jonker, C.M., and Treur, J., A Requirement Specification Language for Configuration Dynamics of Multi-Agent Systems. In: Wooldridge, M., Weiss, G., and Ciancarini, P. (eds.), Proc. of the 2nd International Workshop on Agent-Oriented Software Engineering, AOSE'01. Lecture Notes in Computer Science, vol. 2222. Springer Verlag, 2002, pp. 169-187.
7. 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.
8. Ferber, J., and Gutknecht, O. (1999). Operational Semantics of a role-based agent architecture. Proceedings of the 6th Int. Workshop on Agent Theories, Architectures and Languages (ATAL'1999). In: Jennings, N.R. & Lesperance, Y. (eds.) Intelligent Agents VI, Lecture Notes in AI, vol. 1757, Springer Verlag, 2000, pp. 205-217.
9. 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 MAAMAW'01, 2001.

10. Fisher, M. (1994). A survey of Concurrent MetateM — the language and its applications. In: D.M. Gabbay & H.J. Ohlbach (eds.), Temporal Logic - Proceedings of the First International Conference, Lecture Notes in AI, vol. 827, pp. 480–505.
11. Hannoun, M., Boissier, O., Sichman, J.S., and Sayettat, C. (2000). Moise: An organizational model for multi-agent systems, In: M.C. Monard and J.S. Sichman, (eds.), Advances in Artificial Intelligence (International Joint Conference IBERAMIA-SBIA), Lecture Notes in AI, vol. 1952, Springer Verlag, 2000, pp. 156-165.
12. Huczynski, A. and Buchanan, D. (1985). Organizational Behaviour, Prentice Hall
13. Jonker, C.M., Letia, I.A., and Treur, J., (2002). Diagnosis of the Dynamics within an Organisation by Trace Checking of Behavioural Requirements. In: Wooldridge, M., Weiss, G., and Ciancarini, P. (eds.), Agent-Oriented Software Engineering II, Proc. AOSE'01. Lecture Notes in Computer Science, vol. 2222. Springer Verlag, 2002, pp. 17-32.
14. Jonker, C.M., and Treur, J., Compositional Verification of Multi-Agent Systems: a Formal Analysis of Pro-activeness and Reactiveness. *International Journal of Cooperative Information Systems*, vol. 11, 2002, pp. 51-92.
15. Jonker, C.M., and Treur, J. (2002). Relating Structure and Dynamics in an Organisation Model. In: J.S. Sichman, F. Bousquet, and P. Davidson (eds.), Proc. of the Third International Workshop on Multi-Agent Based Simulation, MABS'02, 2002, pp. 71-80. To be published by Springer Verlag.
16. Jonker, C.M., and Treur, J., Modelling Multiple Mind-Matter Interaction. International Journal of Human-Computer Studies, vol. 57, 2002, pp. 165-214. Preliminary and shorter version in: In: M. Boman, W. van de Velde (eds.), Multi-Agent Rationality, Proc. MAAMAW'97, Lecture Notes in AI, vol. 1237, Springer Verlag, Berlin, 1997, pp. 210-233.
17. Jonker, C.M., Treur, J., and Vries, W. de, Reuse and Abstraction in Verification: Agents Acting in a Dynamic Environment. In: P. Ciancarini, M.J. Wooldridge (eds.), Agent-Oriented Software Engineering, Proceedings of the First International Workshop (AOSE-2000). Lecture Notes in Computer Science, vol. 1957, Springer Verlag, 2001, pp. 253-268.
18. Jonker, C.M., Treur, J., and Wijngaards, W.C.A., (2001). 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, Proceedings 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.
19. Kreitner, R., and Kunicki, A. (2001). Organisational Behavior, McGraw – Hill.
20. Lomi, A., and Larsen, E.R. (2001). Dynamics of Organizations: Computational Modeling and Organization Theories, AAAI Press, Menlo Park.
21. Maes, P., D. Nardi (eds), Meta-level architectures and reflection, Elsevier Science Publishers, 1988.
22. Mintzberg, H. (1979). The Structuring of Organisations, Prentice Hall, Englewood Cliffs, N.J.
23. Moss, S., Gaylard, H., Wallis, S., and Edmonds, B. (1998). SDML: A Multi-Agent Language for Organizational Modelling, Computational and Mathematical Organization Theory 4, (1), 43-70.
24. Prietula, M., Gasser, L., Carley, K. (1997). Simulating Organizations. MIT Press.
25. Pugh, D.S. (ed.), 1971, Organization Theory: Selected Readings, Penguin Books, Harmondsworth
26. Yovine, S. (1997). Kronos: A verification tool for real-time systems. International Journal of Software Tools for Technology Transfer, Vol. 1, Issue 1/2, pages 123-133, October 1997.