Modelling an Agent's Mind and Matter

Catholijn M. Jonker, Jan Treur

Vrije Universiteit Amsterdam
Department of Mathematics and Computer Science
Artificial Intelligence Group
De Boelelaan 1081a, 1081 HV Amsterdam, The Netherlands
Email: {jonker,treur}@cs.vu.nl URL: http://www.cs.vu.nl

Abstract. In agent models often it is assumed that the agent maintains internal representations of the material world (e.g., its beliefs). An overall model of the agent and the material world necessarily incorporates sub-models for physical simulation and symbolic simulation, and a formalisation of the (static and dynamic) representation relation between the two types of sub-models. If it is also taken into account that the agent's mind has a materialisation in the form of a brain, the relations between mind and matter become more complex. The question of how the different types of interaction between mind and matter of an agent and the material world can be modelled in a semantically sound manner is the main topic of this paper. The model can be used to simulate a variety of phenomena in which (multiple) mind-matter interactions occur, such as sensing, acting, (planned) birth and death, causing brain damage, and psychosomatic diseases.

1 Introduction

To be able to maintain interaction with a dynamic (material) world is one of the crucial abilities of most agents. How to define the semantics of the relation of an agent with the material world in a formal manner is a nontrivial issue. For example, in [Pylyshyn, 1986] the relations are described by so-called transducers that connect aspects of the material world to symbolic representations and vice versa. Also in practical agent modelling projects the division and relation between agent and material world is not trivial. For example, a cup on a table can be considered part of the material world, but it is also convenient to consider material aspects of the agent as part of the world; for example a relation between the cup and a robot gripper that has picked up the cup then can be viewed as part of the structure of the material world. This perspective can be extended to a material world describing two agents shaking hands or even one agent, the left hand of which has gripped the right hand. These external material agent aspects (the agent's matter) can be modelled as different from the internal mental aspects of the agent such as its beliefs about the world, its goals and plans, and its reasoning (the agent's mind). If it is also taken into account that the agent's mind has a materialisation in the form of brain, the relations between mind and matter become more complex. The question of how the different manners in which the mind and matter aspects of an agent relate, and how their interaction can be modelled is the main topic of the current paper.

1.1 The Knowledge Representation Hypothesis

One of the starting points is the *knowledge representation hypothesis* formulated in [Smith, 1982, 1985]. The essence of this hypothesis is the strict division between (a) the meaning of a *representation*, that can be attributed from outside, and (b) the *manipulation* of representation structures independent of their meaning, that is, it proceeds on the basis of form only.

In logic the knowledge representation hypothesis is the basis for formal systems. These systems formally define a language (in which formulae stand for the representations of knowledge), e.g., the language of predicate logic. The attribution of semantics is formalised by formal structures (called models, standing for world states the knowledge refers to); e.g., [Tarski, 1956; Dalen, 1980; Chang and Keisler, 1973; Hodges, 1993]. For connections to reasoning systems, e.g., see [Weyhrauch, 1980; Treur 1988, 1991]. The manipulation of these syntactical structures is based on inference rules, such as modus ponens, conjunction introduction, and others. These inference rules are defined in a generic manner: they do not depend on the meaning of the formulae on which they are applied.

Formal systems as defined in logic can be used to formalise cognitive representation systems and their (reference) relation with the material world they represent. However, there is a second type of relation between a cognitive system and the material world: the cognitive representations themselves are embodied in a material form in the brain.

1.2 Two Types of Representations

Although it is not known in all details how the mental activities of a human agent take place, it is clear that the brain is an essential material aspect of it. Every thought is somehow physically embodied within the brain and every reasoning process is performed as physical brain activity. This is a completely different relation between a

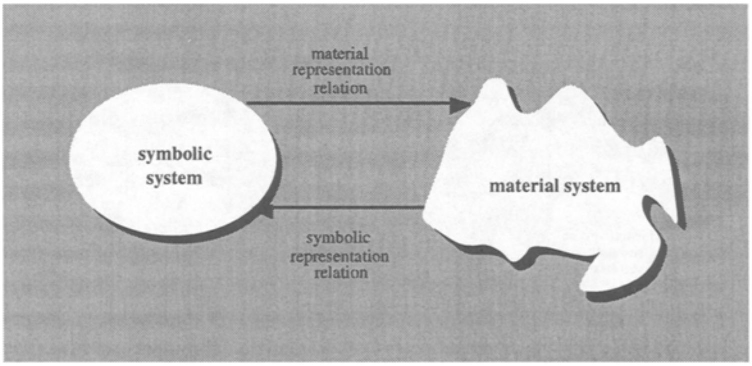

Fig. 1. Dual representation relations

symbolic system and a material system that has nothing to do with the content of the symbolic representation (i.e., the material world aspects the representations refer to), but only with the form in which the representation is materialised.

In this paper we interpret the materialisation of representations as a second process of representation, on which again the knowledge representation hypothesis can be applied. For example, consider the concept of time. The symbolic representation noon can be represented in a material manner by a clock. A clock, a material piece of machinery, represents the symbol noon by the material configuration in which both hands of the clock point upward. Manipulations with these material representations take place according to physical laws that indeed (as demanded by the knowledge representation hypothesis) are independent of the content the representations refer to; i.e., the movement of the hands of the clock just follow physical laws and are not affected in any manner by our attribution of semantics to the material configurations that occur.

Thus, following the knowledge representation hypothesis, it is not only possible to represent material aspects in a symbolic manner, it is also possible to represent symbolic or mental aspects in a material manner. We distinguish the two types of representation as *material representation* versus *symbolic representation*. Dual representation relations are obtained (see Figure 1): material aspects of the world have a symbolic representation, and symbolic aspects have a material representation. Note that these relations are not related in a direct manner; e.g., they are not each others' inverse. Specific and bi-directional types of mind-matter interaction do occur frequently: observations in the material world affecting the information in the brain (sensing), mental processes leading to material actions affecting the world (acting), material processes in the world affecting the brain itself (e.g., causing brain damage), or mental processes affecting the material state of the body (e.g., causing psychosomatic diseases).

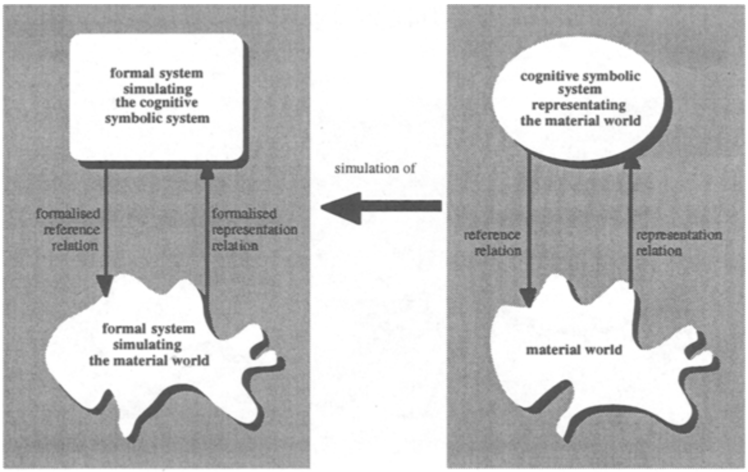

Fig. 2. Simulating the material world and the cognitive symbolic system representing it.

1.3 Simulation of Material and Symbolic Processes and their Interactions

The model developed in this paper simulates both types of mind-matter interaction. The material world in which agents live and think is depicted in Figure 2 at the right bottom. The cognitive symbolic system depicted at the right top represents the world and performs reasoning about the world (cf., [Lindsay and Norman, 1977, 1989; Newell, 1980; Laird, Newell and Rosenbloom, 1987; Simon and Kaplan, 1989]).

In order to make a model of the interacting material and symbolic processes that is executable on a computer system, a (formal) simulation model can be made. The simulation model is depicted on the left-hand side of Figure 2. It formalises the following processes:

- the material processes in the physical world
- the symbolic processes in the cognitive system
- the interaction between these two types of processes

Note that a simulation does not pretend to have the exact same behaviour as the original system: a rough approximation may be sufficient to obtain a specific insight in these processes.

2 An Example of Multiple Interaction Between Mind and Matter

Consider an agent walking down a street, see Figure 3 (position p1). The agent observes that he can buy an ice-cream in the supermarket across the street (the supermarket is at position p3 in Figure 3). As he has a desire for ice-cream, he sets himself the goal of visiting the super-market. To do this he has to cross the street. Although the shrub to his left limits his view of the road, he decides to cross the street as he does not see any cars.

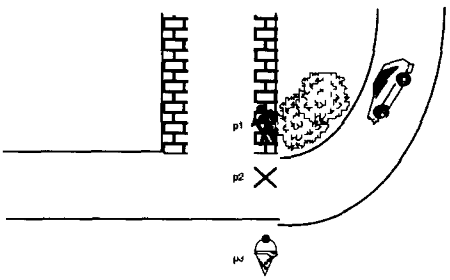

Fig. 3. Initial situation

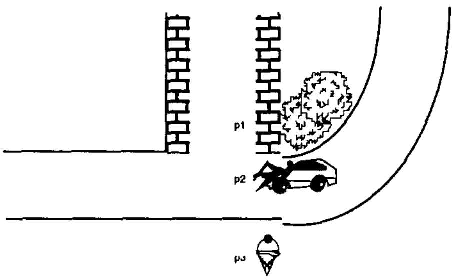

Fig. 4. Situation at the time of the accident

Unfortunately, there is a car coming down the street. The driver, being a bit in a hurry, comes around the curve with the shrub (position p2 in Figures 3 and 4) at the same moment that the agent arrives at position p2. As can be seen in Figure 4, the car hits the agent. Although the accident is minor one (the agent has no permanent injuries), the agent is momentarily stunned and suffers from temporary amnesia (his short-term memory is lost). One of the effects is that the agent cannot remember his goal to visit the super-market. Furthermore, he cannot remember any of the observations done prior to the crossing of the street. Realising that he lacks knowledge about his present predicament, the agent decides to observe his surroundings (again).

3 DESIRE: a Modelling Framework for Multi-Agent Systems

The model described in this paper is specified within the compositional modelling framework DESIRE for multi-agent systems (framework for DEsign and Specification of Interacting REasoning components; cf. [Brazier, Dunin-Keplicz, Jennings, Treur, 1995, 1997]). A number of generic models for agents and their tasks have been developed in DESIRE and have been used for a number of applications (e.g., [Brazier, Jonker and Treur, 1997]). The architectures upon which specifications for compositional multi-agent systems are based are the result of analysis of the tasks performed by individual agents and groups of agents. Task compositions include specifications of interaction between tasks at each level within a task composition, making it possible to explicitly model tasks which entail interaction between agents and interaction between agents and the world (which is modelled as a separate component). Models specified within DESIRE define the structure of *compositional architectures.*: Components in a compositional architecture are directly related to tasks in a task composition. The hierarchical structures of tasks, interaction and knowledge are fully preserved within compositional architectures. Below the formal compositional framework for modelling multi-agent tasks DESIRE is introduced, in which the following aspects are modelled and specified:

(1) a task composition,
(2) information exchange,
(3) sequencing of tasks,
(4) task delegation,
(5) knowledge structures.

3.1 Task Composition

To model and specify composition of tasks, knowledge of the following types is required:

- a *task hierarchy*,
- information a task requires as *input*,
- information a task produces as a *result* of task performance
- *meta-object* relations between tasks

Within a task hierarchy *composed* and *primitive* tasks are distinguished: in contrast to primitive tasks, composed tasks are composed of other tasks, which, in turn, can be either composed or primitive. Tasks are directly related to components: composed tasks are specified as composed components and primitive tasks as primitive components.

Information required/produced by a task is defined by *input* and *output signatures* of a component. The signatures used to name the information are defined in a predicate logic with a hierarchically ordered sort structure (*order-sorted predicate logic*). Units of information are represented by the ground *atoms* defined in the signature.

The role information plays within reasoning is indicated by the level of an atom within a signature: different (meta)levels may be distinguished. In a two-level situation the lowest level is termed *object-level information*, and the second level *meta-level information*. Meta-level information contains information about object-level information and reasoning processes; for example, for which atoms the values are still unknown (*epistemic information*). Similarly, tasks which include reasoning about other tasks are modelled as meta-level tasks with respect to object-level tasks. Often more than two levels of information and reasoning occur, resulting in meta-meta-... information and reasoning.

3.2 Information Exchange

Information exchange between tasks is specified as *information links* between components. Two types of information links are distinguished: *private* information links and *mediating* information links. For a given parent component, a private information link relates output of one of its components to input of another, by specifying which truth value of a specific output atom is linked with which truth value of a specific input atom. Atoms can be renamed: each component can be specified in its own language, independent of other components. In a similar manner mediating links transfer information from the input interface of the parent component to the input interface of one of its components, or from the output interface of one of its components to the output interface of the parent component itself. Mediating links specify the relation between the information at two adjacent abstraction levels in the component hierarchy. The conditions for activation of information links are explicitly specified as task control knowledge.

3.3 Sequencing of Tasks

Task sequencing is explicitly modelled within components as *task control knowledge*. Task control knowledge includes not only knowledge of which tasks should be activated when and how, but also knowledge of the goals associated with task activation and the extent to which goals should be derived. These aspects are specified as component and link activation together with task control foci and extent to define the component's goals. Components are, in principle, black boxes to the task control of an encompassing component: task control is based purely on information about the success and/or failure of component reasoning. Reasoning of a component is considered to have been successful with respect to its task control focus if it has reached the goals specified by this task control focus to the extent specified (e.g., any or every).

3.4 Delegation of Tasks

During knowledge acquisition a task as a whole is modelled. In the course of the modelling process decisions are made as to which tasks are (to be) performed by which agent. This process, which may also be performed at run-time, results in the delegation of tasks to the parties involved in task execution. In addition to these specific tasks, often generic agent tasks, such as interaction with the world (observation) and other agents (communication and cooperation) are assigned.

3.5 Knowledge Structures

During knowledge acquisition an appropriate structure for domain knowledge must be devised. The meaning of the concepts used to describe a domain and the relations between concepts and groups of concepts, are determined. Concepts are required to identify objects distinguished in a domain (domain-oriented ontology) , but also to express the methods and strategies employed to perform a task (task-oriented ontology). Concepts and relations between concepts are defined in hierarchies and rules based on order-sorted predicate logic. In a specification document references to appropriate knowledge structures (specified elsewhere) suffice; compositional knowledge structures are composed by reference to other knowledge structures.

The semantics of the modelling language are based on temporal logic (cf., [Engelfriet and Treur, 1994; Treur, 1994; Brazier, Treur, Wijngaards and Willems, 1996]. By explicitly modelling and specifying the semantics of static and dynamic aspects of a system, a well-defined conceptual description is acquired that can be used for verification and validation, but also is a basis for reuse. Conceptual design is supported by graphical tools within the DESIRE software environment. Translation to an operational system is straightforward; the software environment includes implementation generators with which formal specifications can be translated into executable code. DESIRE has been successfully applied to design both single agent and multi-agent systems.

4 The Material World and its Symbolic Representation

In this section the material world and its symbolic representation, as well as the concept of transducers are discussed. The approach discussed in the Introduction will be applied. In Figure 5 the component material_world simulates the actual material world. All changes with respect to physical aspects of objects take place within this component. The component symbolic_representation_of_material_world simulates the state of the symbolic representation of the material world over time. Both components and their interaction will be discussed in more detail in subsequent (sub)sections.

4.1 Material World

As discussed in the Introduction, the material world is simulated by a specification in terms of executable temporal rules. The vocabulary within the component material_world in which these temporal rules are expressed is defined by a signature that has a compositional structure:

```
signature material_world_sig
    signatures generic_material_world_sig, specific_material_world_sig, specific_material_brain_sig;
end signature
```

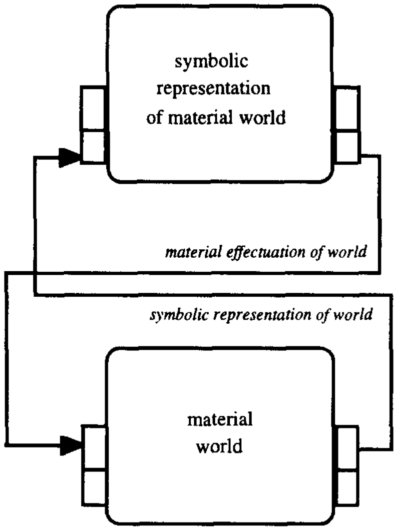

Fig. 5. Transduction links between the material world and its symbolic representation

The (composed) signature material_world_sig refers to three other signatures which are specified below. Referring to another signature means that all language elements of that other signature can be used to determine the vocabulary specified by the signature.

```
signature generic_material_world_sig
    sorts
        ACTION, AGENT, AGENT_PROPERTY, EVENT, OBJECT, POSITION, PROPERTY, SIGN, TIME ;
    sub-sorts
        ACTION :                  EVENT ;
        AGENT :                   OBJECT ;
        AGENT_PROPERTY :          PROPERTY ;
    objects
        agent :                   AGENT ;
        neg, pos :                SIGN ;
        t0, t1, t2, t3 :          TIME ;
    functions
        position:                 OBJECT * POSITION -> PROPERTY ;
    relations
        at_time:                  PROPERTY * SIGN * TIME ;
        current_time:             TIME ;
        currently :               PROPERTY * SIGN ;
        effect :                  EVENT * PROPERTY * SIGN ;
        event_after :             EVENT * TIME ;
        event_to_happen :         EVENT ;
        next :                    PROPERTY * SIGN ;
        next_time_point :         TIME ;
        precedes :                TIME * TIME ;
end signature

signature specific_material_world_sig
    objects
        car_to_appear:                  EVENT ;
        car, ice_cream, supermarket:    OBJECT ;
```

```
        p1, p2, p3:                      POSITION ;
        car_present:                     PROPERTY ;
    functions
        close_by_for:                    OBJECT * AGENT -> PROPERTY ;
        goto:                            POSITION -> ACTION ;
        has_hit:                         OBJECT * OBJECT -> PROPERTY ;
        next_on_path:                    POSITION * POSITION * POSITION -> PROPERTY ;
        sells:                           OBJECT * OBJECT -> PROPERTY ;
    relations
        current_action_by:               ACTION * AGENT ;
        current_observation_by:          PROPERTY * AGENT ;
        current_observation_result_of:   PROPERTY * SIGN * OBJECT ;
end signature

signature specific_material_brain_sig
    sorts
        AGENT_ATOM, BRAIN_LOCATION, INFORMATION_OBJECT, LTM_LOCATION, STM_LOCATION;
    sub-sorts
        AGENT_ATOM :                     AGENT_PROPERTY ;
        BRAIN_LOCATION :                 POSITION ;
        LTM_LOCATION :                   BRAIN_LOCATION ;
        STM_LOCATION :                   BRAIN_LOCATION ;
        INFORMATION_OBJECT:              OBJECT ;
    functions
        contents_of_stm_to_ltm:          INFORMATION_OBJECT * STM_LOCATION -> EVENT ;
        has_amnesia:                     AGENT -> AGENT_PROPERTY ;
        information_object:              AGENT_ATOM * SIGN -> INFORMATION_OBJECT ;
        ltm_location:                    AGENT_ATOM * TIME -> LTM_LOCATION ;
        recovered:                       AGENT -> AGENT_PROPERTY ;
        recovering:                      AGENT -> EVENT ;
        stm_location:                    AGENT_ATOM * TIME -> STM_LOCATION ;
        to_be_stored:                    AGENT_ATOM * SIGN ;
end signature
```

The (temporal) knowledge simulating the processes within the material world is specified as follows:

```
/* domain dependent knowledge */

at_time(position(supermarket, p3), pos, T: TIME) ;
at_time(sells(supermarket, ice_cream),pos,T: TIME) ;
at_time(position(agent, p1), pos, t1) ;
at_time(car_present, neg, t1) ;

effect(car_to_appear, position(car, p1), neg) ;
effect(car_to_appear, position(car, p2), pos) ;
effect(goto(P : POSITION), position(agent, P : POSITION), pos) ;
effect(recovering(X : OBJECT), has_amnesia(X : OBJECT), neg) ;

event_after(car_to_appear, t1) ;

precedes(t0, t1) ;
precedes(t1, t2) ;
precedes(t2, t3) ;

if   at_time(position(A : AGENT, p1), pos, T: TIME)
     and at_time(position(O:OBJECT,p3),pos,T:TIME)
then at_time(close_by_for(O : OBJECT, A : AGENT), pos ,T : TIME) ;

if   current_observation_by(P : PROPERTY, A : AGENT)
     and current_time(T : TIME)
     and at_time(P : PROPERTY, S : SIGN, T : TIME)
then current_observation_result_of(P : PROPERTY, S : SIGN, A : AGENT) ;
```

```
/* domain independent knowledge */
if   at_time(position(X : OBJECT, P : POSITION), pos, T1 : TIME)
     and at_time(position(Y : OBJECT, Q : POSITION), pos, T1 : TIME)
     and not equal(X : OBJECT, Y : OBJECT)
     and not equal(P : POSITION, Q : POSITION)
     and precedes(T1 : TIME, T2 : TIME)
     and at_time(position(X : OBJECT, R : POSITION), pos, T2 : TIME)
     and at_time(position(Y : OBJECT, R : POSITION), pos, T2 : TIME)
then at_time(has_hit(X : OBJECT, Y : OBJECT), pos, T2 : TIME) ;

if   at_time(has_hit(X : OBJECT, agent), pos, T : TIME)
then at_time(has_amnesia(agent), pos, T : TIME) ;

if   current_action_by(A : ACTION, X : AGENT )
     and effect(A : ACTION, P : PROPERTY, S : SIGN)
then next(P : PROPERTY, S : SIGN) ;

if   event_to_happen(E : EVENT )
     and effect(E : EVENT, P : PROPERTY, S : SIGN)
then next(P : PROPERTY, S : SIGN) ;

if   currently(has_amnesia(X : OBJECT), pos)
then event_to_happen(recovering(X : OBJECT)) ;

if   current_time(T2 : TIME)
     and precedes(T1 : TIME, T2 : TIME)
     and at_time(position(I : INFORMATION_OBJECT, B: STM_LOCATION), pos, T1 : TIME)
then event_to_happen(contents_of_stm_to_ltm(I : INFORMATION_OBJECT, B: STM_LOCATION)) ;

if   current_time(T1 : TIME)
     and precedes(T1 : TIME, T2 : TIME)
then effect(contents_of_stm_to_ltm(I : INFORMATION_OBJECT, B: STM_LOCATION),
            position(I : INFORMATION_OBJECT, ltm_location( : INFORMATION_OBJECT, T2 : TIME)), pos ) ;

if   currently(has_amnesia(X : AGENT), neg)
     and current_time(T2 : TIME)
     and precedes(T1 : TIME, T2 : TIME)
     and not event_to_happen(contents_of_stm_to_ltm(I : INFORMATION_OBJECT, B: STM_LOCATION))
     and at_time(position(I : INFORMATION_OBJECT, B: STM_LOCATION), pos, T1 : TIME)
then at_time(position(I : INFORMATION_OBJECT, B: STM_LOCATION), pos,  T2 : TIME) ;

if   current_time(T2 : TIME)
     and precedes(T1 : TIME, T2 : TIME)
     and at_time(I : INFORMATION_OBJECT, B: LTM_LOCATION), pos, T1 : TIME)
then at_time(I : INFORMATION_OBJECT, B: LTM_LOCATION), pos, T2 : TIME) ;

if   current_time(T : TIME)
     and at_time(P : PROPERTY, S : SIGN, T : TIME)
then currently(P : PROPERTY, S : SIGN) ;

if   event_after(E : EVENT, T : TIME )
     and current_time(T : TIME)
then event_to_happen(E : EVENT ) ;

if   not equal(P : POSITION, Q : POSITION)
then effect(goto(P : POSITION), position(agent, Q : POSITION), neg) ;

if   current_time(T1 : TIME)
     and precedes(T1 : TIME, T2 : TIME)
then next_time_point(T2 : TIME) ;

if   next_time_point(T2 : TIME)
     and next(X : PROPERTY, S : SIGN)
then at_time(X : PROPERTY, S : SIGN, T2 : TIME) ;
```

To execute the temporal rules specified above, updates are required from the current time point to the next time point. These updates are specified by an information link from the component material_world to itself.

4.2 Symbolic Representation of the Material World

In order to reason about the material world and its behaviour, a symbolic representation of the material world is called for. In Figure 5, the component symbolic_representation_of_material_world specifies a simulation of such a representation. The vocabulary used within the component symbolic_representation_of_material_world is specified by the following signature.

```
signature symbolic_representation_of_world_sig
    sorts
        WORLD_TERM ;
    meta-descriptions
        material_world_sig :        WORLD_TERM ;
    relations
        to_be_executed_by :         ACTION * AGENT ;
        to_be_observed_by :         PROPERTY * AGENT ;
        just_acquired :             WORLD_TERM ;
end signature
```

This signature introduces a new sort WORLD_TERM that is used in the construction of a meta-description of the signature material_world_sig. In the meta-description all n-ary relations of the signature are transformed into n-ary functions into the sort WORLD_TERM. This construction allows, for example, the following atom:

just_acquired(current_observation_result_of(car_present, neg, agent))

Within the component symbolic_representation_of_material_world no knowledge is specified. The component in principle only models the maintenance of representation states. Also within this component updates are maintained, i.e., whenever an observation has been performed. Updates are specified by an information link from the component symbolic_representation_of_material_world to itself.

4.3 Transduction Links

As discussed in the introduction, there are two issues in using a symbolic representation of the material world. The first is how changes in the material world become to be reflected in the symbolic representation (upward transduction). The second is how changes in the symbolic representation of the world affect the material world itself (downward transduction). In Figure 5, the simulations of transducers are modelled within the framework DESIRE as information links between the output and input interfaces of the components material_world and symbolic_representation_of_material_world. The information links that model transducers are called *transduction links* (and depicted in italics). The upward transducer is modelled by the transduction link *material_effectuation_of_world*, the downward by the transduction link *symbolic_representation_of_world*. The downward link transfers actions that are to be performed to the component material_world. Given that the computer systems uses of one or more sensors, observations can be made. The results of observations are transferred to the component symbolic_representation_of_material_world, by way of the transduction link *symbolic_representation_of_world*, during which a symbolic representation of the observation results is made that can be processed by the receiving component. In Figure 5 each component has a levelled interface (denoted by the rectangles on side of the components). The transduction link *symbolic_representation_of_world* transfers epistemic meta-level information on the material world (e.g., expressed by the truth of the atom true(current_observation_result_of(car_present, pos, agent))) to object level information that can be used by the component symbolic_representation_of_material_world (expressed by the truth of the atom just_acquired(current_observation_result_of(car_present, pos, agent))):

```
atom links
    (   true(current_observation_result_of(P : PROPERTY, S : SIGN, agent)),
        just_acquired(current_observation_result_of(P : PROPERTY, S : SIGN, agent))
    )   : <<true,true>> ;
    (   true(current_observation_result_of(P : PROPERTY, S : SIGN, agent)),
        just_acquired(current_observation_result_of(P : PROPERTY,  neg, agent))
    )   : <<false,true>> ;
```

The transduction link *material_effectuation_of_world* links information on actions to be executed of the component symbolic_representation_of_material_world, to meta-level information on the material world:

```
atom links
    (   to_be_executed_by(A : ACTION, agent),
        assumption(current_action_by(A : ACTION, agent), pos)
    )   : <<true,true>, <false,false>, <unknown,unknown>> ;
    (   to_be_observed_by(P : PROPERTY, agent),
        assumption(current_observation_by(P : PROPERTY, agent), pos)
    )   : <<true,true>, <false,false>, <unknown,unknown>> ;
```

In this example, the truth value combinations <false,false> and <unknown,unknown> ensure that previous actions are retracted, so that actions will not be performed ad infinitum.

5 An Agent's Rational Behaviour in Interaction with the Material World

As discussed in Section 4 the downward transduction link is needed for the actual execution of actions. However, the component symbolic_representation_of_material_world is not modelled as a component in which rational decisions are made on which observation or action is to be performed and when (pro-active behaviour). Such mental decision processes are modelled in the component agent, see Figure 6.

5.1 Agent

The component agent models the cognitive symbolic reasoning system of an agent as a logical system. The rational agent can determine to perform observations and actions. The vocabulary used within the component agent is specified by the following signature.

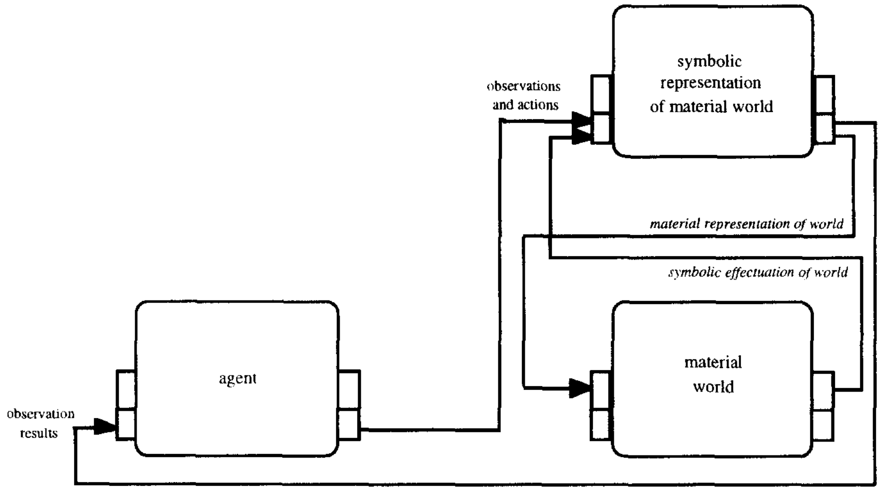

Fig. 6. Transduction and symbolic links connecting agent and material world

```
signature symbolic_agent_sig
    sorts
        WORLD_TERM ;
    meta-descriptions
        material_world_sig :            WORLD_TERM ;
    functions
        current_observation_result :    PROPERTY * SIGN -> WORLD_TERM ;
        close_by :                      OBJECT -> PROPERTY ;
        own_position :                  POSITION ;
        visiting :                      AGENT -> WORLD_TERM ;
    relations
        belief:                         WORLD_TERM ;
        current_belief:                 PROPERTY * SIGN ;
        desire:                         OBJECT ;
        goal:                           WORLD_TERM ;
        most_recent_observation:        PROPERTY * SIGN ;
        observed:                       PROPERTY ;
        observed_at:                    PROPERTY * SIGN * T : TIME ;
        possible_observation:           PROPERTY ;
        to_be_executed:                 ACTION ;
        to_be_observed:                 PROPERTY ;
end signature
```

Note again the meta-description construct within this signature.

5.2 The Agent Components

The agent is modelled as a composed component consisting of two sub-components, own_process_control and maintain_world_information, see Figure 7. The reasoning about its goals, desires, and plans is performed within the component own_process_control. Its knowledge about the world, obtained by observations, is maintained within the component maintain_world_information.

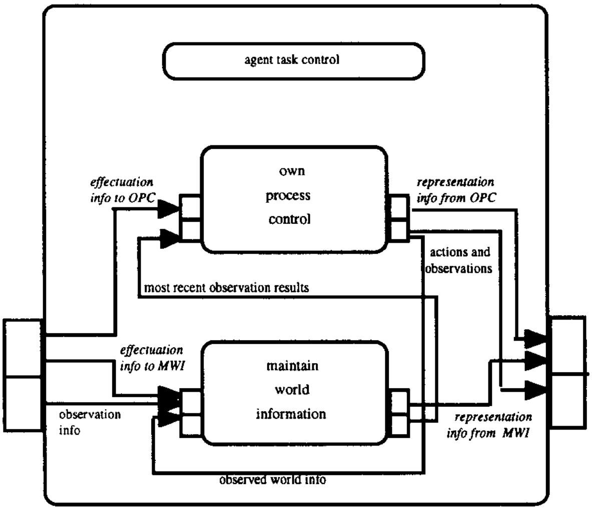

Fig. 7. Transduction links between the agent and its material representation

The component own_process_control contains the following knowledge:

desire(ice_cream) ;
to_be_observed(own_position(P2 : POSITION));
to_be_observed(car_present);

if desire(G : OBJECT)
then possible_observation(position(S : OBJECT, P1 : POSITION))
and possible_observation(close_by(S : OBJECT))
and possible_observation(sells(S : OBJECT, G : OBJECT));

if possible_observation(P : PROPERTY)
and not observed(P : PROPERTY)
then to_be_observed(P : PROPERTY);

if current_belief(sells(S : OBJECT, G : OBJECT), pos)
and desire(G : OBJECT)
and current_belief(close_by(S : OBJECT), pos)
then goal(visiting(S : OBJECT));

if goal(visiting(S : OBJECT))
and current_belief(position(S : OBJECT, P : POSITION), pos)
and current_belief(own_position(Q : POSITION), pos)
and current_belief(car_present, neg)
and current_belief(next_on_path(R : POSITION, Q : POSITION, P : POSITION), pos)
then to_be_executed(goto(R : POSITION));

if current_time(T : TIME)
and just_acquired(current_observation_result(P : PROPERTY, S : SIGN))
then observed_at(P : PROPERTY, S : SIGN, T : TIME);

The link observed_world_info transfers the just acquired knowledge about the world from own_process_control to maintain_world_information. The agent obtains this knowledge by observations. The link updates the truth values of the atom most_recent_observation ensuring that the atom indeed reflects the most recent information about the world.

The link most_recent_observation_results determines the beliefs that are to be held by the agent (within its component own_process_control).

5.3 Symbolic Links

The symbols representing the decisions to perform observations and actions are linked to the symbolic system modelled by the component symbolic_representation_of_material_world. All connections between symbolic systems are called *symbolic links*. Symbolic links are modelled as information links within the framework DESIRE. The symbolic link that transfers the symbolic representations of observations and actions that are to be performed is called observations_and_actions. This link connects the object level of the output interface of the component agent with the object level input interface of the component symbolic_representation_of_material_world:

```
term links
    (close_by(O : OBJECT),           close_by_for(O : OBJECT, agent)) ;
    (own_position(P : POSITION),     position(agent, P : POSITION)) ;
atom links
    (   to_be_observed(P : PROPERTY),
        to_be_observed_by(P : PROPERTY, agent)
    )   : <<true,true>, <false,false>, <unknown,unknown>> ;
    (   to_be_executed(A : ACTION),
        to_be_executed_by(A : ACTION, agent)
    )   : <<true,true>, <false,false>, <unknown,unknown>> ;
```

The results of observations performed within material_world are transferred to the component agent through the transduction link *symbolic_representation_world* (see previous section) and the symbolic link observation_results that connects the component symbolic_representation_of_material_world to the component agent:

```
term links
    (close_by_for(O : OBJECT, agent),   close_by(O : OBJECT)) ;
    (position(agent, P : POSITION),     own_position(P : POSITION)) ;
atom links
    (   just_acquired(current_observation_result_of(X : PROPERTY, S : SIGN, agent)),
        just_acquired(current_observation_result(X : PROPERTY, S : SIGN))
    )   : <<true,true>, <false,false>> ;
    (   just_acquired(current_observation_result_of(X : PROPERTY, S : SIGN, agent)),
        observed(X : PROPERTY)
    )   : <<true,true>> ;
    (   current_time(T : TIME),
        current_time(T : TIME)
    )   : <<true,true>, <false,false>, <unknown,unknown>> ;
```

6 An Agent and its Material Representation

In Figure 8 the cognitive symbolic system of the agent is modelled by the component agent described in the previous section. The component material_representation_of_agent models the material representation of (the symbolic system of) the agent. As discussed in the Introduction, the relation between the agent and its material representation is modelled in a manner similar to the manner in which the relation between the material world and its symbolic representation is modelled. An upward transducer defines how symbolic aspects of the agent are represented in a material form, a downward transducer defines how properties of the material world affect the processes of the symbolic system within the agent.

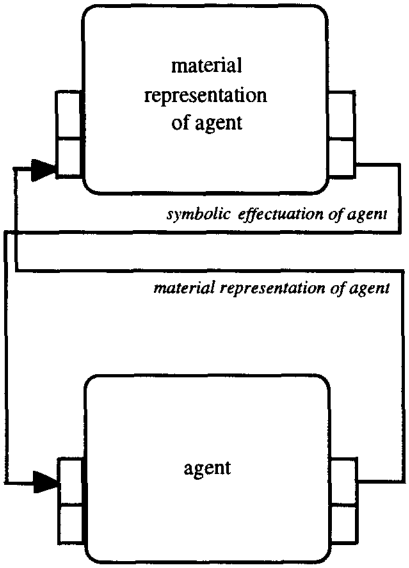

Fig. 8. Transduction links between the agent and its material representation

6.1 Material Representation of an Agent

The vocabulary used within the component material_representation_of_agent is specified by the following composed signature.

signature material_representation_of_agent_sig
  signatures
    generic_material_world_sig , specific_material_brain_sig ;
  meta-descriptions
    symbolic_agent_sig: AGENT_ATOM ;
end signature

The signature symbolic_agent_sig is used in a meta-description construct, such that the relations of that signature can be used as functions into the sort AGENT_ATOM. The sort AGENT_ATOM is a sub-sort of the sort AGENT_PROPERTY, which in turn is a sub-sort of the sort PROPERTY. Therefore, all relations that have PROPERTY as an argument and are specified in the signatures generic_material_world_sig and specific_material_brain_sig can be applied to the new terms.

Within the component material_world a simple model for memory is specified. The component material_representation_of_agent only maintains a state, from and to which information is transferred to and from the component material_world. The only exception is the following knowledge base rule that combines the information to be stored and the current time point and determines the actual storage of the information as a physical property:

```
if   current_time(T : TIME)
     and  to_be_stored(A : AGENT_ATOM, S : SIGN)
then at_time(position(information_object(A : AGENT_ATOM, S : SIGN),
          stm_location(A : AGENT_ATOM, T : TIME), pos, T : TIME) ;
```

6.2 Transduction Links

The information maintained by the agent is built of atoms with an explicit reference to their truth value in the form of a sign. The atom is transformed into a term by the transition from the agent to its material representation. For example, the atom observed_at(car_present, neg, t1) that could be used within the component agent, is represented by a *term* within the component material_representation_of_agent. If the atom is true within agent, the sign pos is to be added within material_representation_of_agent, if the atom was false, the sign neg is to be added. If the atom has the truth-value unknown it is not stored in material_representation_of_agent. If the agent, by reasoning, makes cognitive changes in its beliefs, desires, goals, or knowledge, the material representations of these changes are materialised in the brain. This process of upward transduction is modelled (see Figures 7 and 8) by the transduction links *material_representation_of_agent*, *representation_info_from_OPC* and *representation_info_from_MWI*. As an example, the atom links of the transduction link *representation_info_from_OPC* are specified as follows:

```
atom links
    (   true(A : IIOA),
        to_be_stored(A : AGENT_ATOM, pos)
    )   : <<true,true>> ;
    (   false(A : IIOA),
        to_be_stored(A : AGENT_ATOM, neg)
    )   : <<true,true>> ;
```

An example of an instantiated atom link of *representation_info_from_OPC* is:

```
    (   true(observed_at(car_present, neg, t1)),
        to_be_stored(observed_at(car_present, neg, t1), pos)
    )   : <<true,true>> ;
```

An example of an instantiated atom link of *material_representation_of_agent* is:

```
    (   to_be_stored(observed_at(car_present, neg, t1), pos),
        to_be_stored(observed_at(car_present, neg, t1), pos)
    )   : <<true,true>> ;
```

For simplicity in this paper it is assumed that there exist functions that relate information in memory to locations within the brain, i.e., positions:

```
position(I : INFORMATION_OBJECT, B : BRAIN_LOCATION)
```

The simple model for memory used in this paper has a short-term memory and a long term memory. To model this distinction, the sub-sort BRAIN_LOCATION of POSITION has two sub-sorts: STM_LOCATION and LTM_LOCATION. Given the atom of the agent (a term of the sort AGENT_ATOM) and a time point (a term of the sort TIME), the function stm_location relates information to a position within the short term memory, whereas ltm_location relates information to a position within the long term memory. The time point used by the function is the moment in time that the information is stored into the memory. An information object is specified as

information_object(A : AGENT_ATOM, S : SIGN),

where the sort AGENT_ATOM contains objects that refer to atoms of the agent, e.g., observed_at(car_present, neg, t1). The current status of the memory is modelled by atoms of the form:

currently(position(information_object(A : AGENT_ATOM, S : SIGN), B : BRAIN_LOCATION), pos)

If a physical change within the component material_representation_of_agent occurs, the symbolic interpretation of the changed information is linked to the component agent by the downward transduction process, modelled by the transduction links *symbolic_effectuation_of_agent*, *effectuation_info_to_OPC* and *effectuation_info_to_MWI*. The atom link of the transduction link *effectuation_info_to_OPC* are specified as follows:

atom links
(currently(position(information_object(A : AGENT_ATOM, S : SIGN), B : STM_LOCATION), pos),
assumption(A : AGENT_ATOM, S : SIGN)
) : <<true,true>, <false,false>, <unknown,unknown>> ;

By these transduction links object level information from the component material_representation_of_agent is transferred to meta-level information within the component agent, which defines the current information state of the agent.

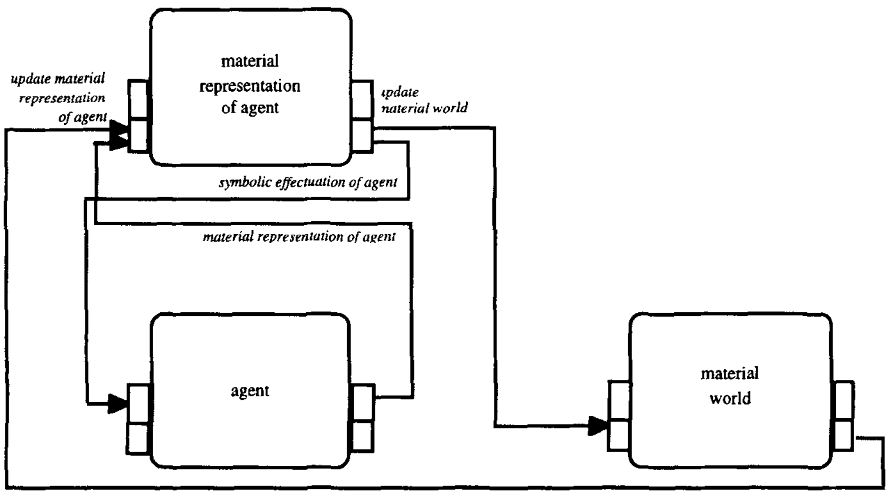

Fig. 9. Transduction and material links connecting material world and agent

7 The Material World's Physical Behaviour in Interaction with an Agent

The material representation of the agent is a part of the material world. Therefore, the component material_representation_of_agent is modelled as a simple component for passing on information. The material links connecting these two components (see Figure 9), update_material_world and update_material_representation_of_agent, are simple identity links, i.e., they only transfer information, they do not translate it. For example, the material link update_material_representation_of_agent links atoms to themselves:

```
(   at_time(position(I : INFORMATION_OBJECT, B : BRAIN_LOCATION), S : SIGN, T : TIME) ,
    at_time(position(I : INFORMATION_OBJECT, B : BRAIN_LOCATION), S : SIGN, T : TIME)
)   : <<true,true>, <unknown,unknown>, <false,false>> ;
```

8 The Complete Model

As can be seen from Figures 5, 6, 8, and 9 it is possible to create a symbolic representation of a material system and to create a material representation of a symbolic system. In Figure 10 all components and all information links (transduction, symbolic and material links) of the top level of the complete model are presented. Together, they sketch two connections between the agent and the material world. The connection between material representations and symbolic representations is made by transduction links, between symbolic representations by symbolic links and between material representations by material links.

9 Trace of the Example Interaction Patterns

In this section it is shown how the course of events in the example introduced in Section 2 is simulated as a reactive pattern using the model introduced in the previous sections. The trace is started at the moment that the agent is in position p1 and has observed that a supermarket where ice cream is sold is at position p3, and that a path from p1 to p3 is available with p2 as next position. Moreover, the agent has observed that no car was present. These observations where made using the transduction links symbolic_effectuation_of_world and material_representation_of_world between the material world and its symbolic representation, and the symbolic links observations_and_actions and observation_results. As a result the observation information is available within the agent (as current beliefs). The trace is started at time point t1. The situation at time t1 is represented in Figure 3.

- *reasoning within the component* agent;
  it derives conclusions goal(visiting(supermarket)), to_be_executed(goto(p2))
- *transfer the action to the material world*
  by the symbolic link observations_and_actions to the component symbolic_representation_of_material_world and by the downward transduction link material_effectuation_of_world to the component material_world

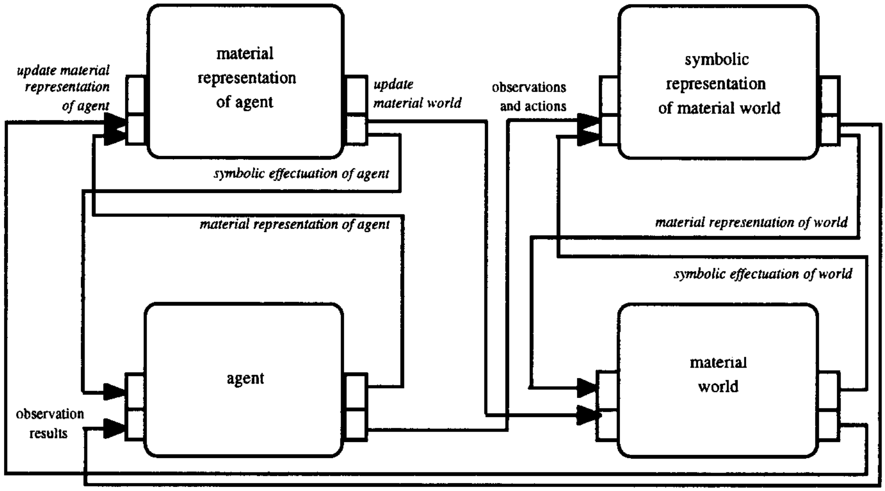

Fig. 10. Transduction, symbolic and material links connecting agent and material world

- *execution of the action and the event car appears within the material world; determination of consequences thereof, see Figure 4.*
  determination of the effect at_time(position(agent, p2), t2) of the action goto(p2);
  determination of the effect at_time(position(car, p2), t2) of the event car_to_appear;
  determination of at_time(has_hit(car, agent), t2);
  determination of at_time(has_amnesia(agent), t2);
  no determination of at_time(position(I:INFORMATION_OBJECT, B:STM_LOCATION), pos, t2)
  because the condition currently(has_amnesia(agent), neg) lacks;
- *transfer the effects to the agent*
  the material effects and their consequences are transferred by the material link update_material_representation_of_agent and from there by the downward transduction link symbolic_effectuation_of_agent to the component agent;
  because of this no facts are available anymore that were materially represented in the STM memory: the agent looses facts such as: goal(visiting(supermarket)), to_be_executed(goto(p2)), and all observations performed at the previous two time points
- *reasoning of the agent with lost STM.*
  the agent has a severe lack of knowledge about its current situation; e.g., what is its position, what was its goal; it decides to observe again. By new observations information on the current material situation can be obtained; however, information about its (previous) goal cannot be found that easily.

and so on.....

10 Other Types of Interaction Patterns Between Mind and Matter

In the previous sections the example course of events was simulated as a reactive pattern trough Figure 10 from the lower left hand side component (agent) to the upper right hand side component (symbolic_representation_of_material_world) to the lower right hand side component (material_world) to the higher left hand side component (material_representation_of_agent) to the lower left hand side component (agent). Also for various other types of interaction between symbolic systems and material systems such patterns can be identified. In this section a number of examples are discussed.

- *Drug use*

Using the model introduced in this paper the process of taking a (narcotic) drug can be simulated as follows (see Figure 10):

- *decision of the agent to take the drug*
  reasoning within the component agent; deriving conclusion to_be_executed(take_drug)
- *transfer the action to the material world*
  by the symbolic link observations_and_actions to the component symbolic_representation_of_material_world and by the downward transduction link *material_effectuation_of_world* to the component material_world
- *execution of the action take drug within the material world*
  determination of the effect active_brain of the action take_drug
- *transfer the effects of take drug to the agent*
  by the material link update_material_representation_of_agent and the downward transduction link *symbolic_effectuation_of_agent* to the component agent
- *execution of the agent with drug effect*

- *Agents planning and executing birth and death*

Using the model introduced in this paper the process of creating a new child agent by a rational agent can be simulated by a similar pattern in Figure 10:

- *decision of the agent to create a child agent*
  reasoning within the component agent; deriving conclusion to_be_executed(create_child)
- *transfer the action to the material world*
  by the symbolic link observations_and_actions to the component symbolic_representation_of_material_world and by the downward transduction link *material_effectuation_of_world* to the component material_world
- *execution of the action create child within the material world*
  determination of the effect of the action create_child
- *transfer the effects of to create child to the agent*
  by the material link update_material_representation_of_agent and the downward transduction link *symbolic_effectuation_of_agent* to the component agent; this link modifies the component agent by replacing it by two similar components
- *execution of the agent and its child agent*

In a similar manner a rational action to kill an agent can be modelled.

- *Psychosomatic diseases*

For psychosomatic diseases the pattern in Figure 10 proceeds in a different direction: from the lower left hand side component to the upper left hand side component to the lower right hand side component. For example, a heart attack induced by psychological factors can be modelled as follows.

- *the agent reasons about highly stress-provoking information*
  stressful reasoning within the component agent
- *transfer of the stress to the material representation of the agent*
  by the upward transduction link *material_representation_of_agent* to the component material_representation_of_agent (to the property over_active_brain) and by the material link update_material_world to the component material_world
- *execution of the material world*
  determination of the effect over_active_brain to heart functioning

11 Discussion

Internal representations of the material world as maintained by an agent, are related to the material world by a representation/reference relation. In this paper a simulation model is introduced covering both a sub-model for the agent (simulating its mental processes) and a sub-model for the material world (simulating its physical processes). The semantical relations between the two sub-models are formalised as dual repesentation relations. In the model it is taken into account that the agent's mind has a materialisation in the form of a brain.

Most parts of the specification of the model are generic; although the example instantiation that is used to illustrate the model is kept rather simple, the generic part of the model can be (re)used to simulate a variety of phenomena in which (multiple) mind-matter interactions occur. The compositional design method DESIRE supports that specific components in the model can be replaced by other components without affecting the rest of the model. For example, more sophisticated memory models can replace the rather simplistic model used as an illustration in this paper.

The work in this paper is of importance for

- foundational questions from a philosophical and logical perspective
- research in cognitive psychology, neuro-physiology, and their relation
- application to dynamic multi-agent domains in which agents can be created and killed

The relevance of the model for each of these three areas will be explained.

An interesting foundational philosophical and logical issue is the semantics of dual representation relations (see also [Hofstadter, 1979]). Both from a static and from a dynamic perspective further questions can be formulated and addressed. For example, the further development of a foundation of semantic attachments and reflection principles [Weyhrauch, 1980] in the context of dual representation relations, and dynamically changing mental and physical states. Another question is the semantically sound integration of (qualitative and quantitative) simulation techniques and (temporal) logical modelling.

Cognitive and neuro-physiological models can be semantically integrated using the model introduced in this paper. The presented generic model can be instantiated by existing models of both kinds. A useful test for existing philosophical approaches to the mind-body problem (e.g., such as described in [Harman, 1989]) is to investigate the possibility to operationalise them using the presented model.

Among the applications of the model are agents capable of planning and executing life affecting actions, such as giving birth and killing (other) agents. These capabilities are essential for Internet agents that can decide on the fly to create new agents to assist them in their tasks and removing these agents after completion of the task they were created for. This application area is one of the focusses of our current research.

References

Brazier, F.M.T. , Dunin-Keplicz, B., Jennings, N.R. and Treur, J. Formal specification of Multi-Agent Systems: a real-world case. In: V. Lesser (ed.), Proceedings of the First International Conference on Multi-Agent Systems, ICMAS'95, MIT Press, Cambridge, MA, 1995, pp. 25-32.

Brazier, F.M.T., Dunin-Keplicz, B., Jennings, N.R. and Treur, J. DESIRE: modelling multi-agent systems in a compositional formal framework, International Journal of Cooperative Information Systems, M. Huhns, M. Singh, (eds.), special issue on Formal Methods in Cooperative Information Systems: Multi-Agent Systems, vol. 6, 1997, pp. 67-94.

Brazier, F.M.T., Jonker, C.M., Treur, J., Formalisation of a cooperation model based on joint intentions. In: J.P. Müller, M.J. Wooldridge, N.R. Jennings (eds.), Intelligent Agents III (Proc. of the Third International Workshop on Agent Theories, Architectures and Languages, ATAL'96), Lecture Notes in AI, volume 1193, Springer Verlag, 1997, pp. 141-155.

Brazier, F.M.T., Treur, J., Wijngaards, N.J.E. and Willems, M. Temporal semantics of complex reasoning tasks. In: B.R. Gaines, M.A. Musen (eds.), Proceedings of the 10th Banff Knowledge Acquisition for Knowledge-based Systems workshop, KAW'96, Calgary: SRDG Publications, Department of Computer Science, University of Calgary, 1996, pp. 15/1-15/17.

Chang, C.C., Keisler, H.J., Model theory, North Holland, 1973.

Dalen, D. van, Logic and Structure, Springer Verlag, 1980.

Engelfriet, J., Treur, J. , Temporal Theories of Reasoning. In: C. MacNish, D. Pearce, L.M. Pereira (eds.), Logics in Artifical Intelligence, Proceedings of the 4th European Workshop on Logics in Artificial Intelligence, JELIA '94, Springer Verlag, 1994, pp. 279-299. Also in: Journal of Applied Non-Classical Logics 5 (1995), pp. 239-261.

Harman, G. Some Philosophical Issues in Cognitive Science: Qualia, Intentionality, and the Mind-Body Problem. In: (Posner, 1989), pp. 831-848.

Hofstadter, D., Gödel, Escher, Bach: an Eternal Golden Braid. Basic Books, New York, 1979.

Hodges, W., Model theory, Cambridge University Press, 1993.

Laird, J.E., Newell, A., Rosenbloom, P.S., Soar: An architecture for general intelligence. Artificial Intelligence 33 (1987), pp. 1-64.

Lindsay, P.H. Norman, D.A., Human Information Processing. Academic Press, 1977.

Newell, A., Physical Symbol Systems, Cognitive Science 2 (1980), pp. 135-184.

Newell, A., Rosenbloom, P.S., Laird, J.E., Symbolic Architectures for Cognition. In: [Posner, 1989], pp. 93-132.

Posner, M.I. (ed.), Foundations of Cognitive Science. MIT Press, 1989.

Pylyshyn, Z.W., Do mental events have durations? Behavioral and Brain Sciences, vol. 2 (1979), pp. 277-278.

Pylyshyn, Z.W., Computation and Cognition: Towards a Foundation for Cognitive Science. MIT Press, 1984.

Pylyshyn, Z.W., Computing in Cognitive Science. In: [Posner, 1989], pp. 49-92.

Simon, H.A., Kaplan, C.A., Foundations of Cognitive Science. In: [Posner, 1989], pp. 1-48.

Smith, B.C., Reflection and semantics in a procedural language. MIT Comp. Science Tech. Report 272, Cambridge, Massachusetts, 1982.

Smith, B.C., Prologue to Reflection and Semantics in a Procedural Language. In: Brachman, R.J., Levesque, H.J. (eds.), Readings in Knowledge Representation. Morgan Kaufman, 1985, pp. 31-40.

Tarski, A., Der Wahrheitsbegriff in den formalisierten Sprachen. *Studia Philosophica,* 1:261-405, 1936. English translation: A. Tarski, *Logic, Semantics, Metamathematics.* Oxford University Press, 1956.

Treur, J., Completeness and definability in diagnostic expert systems, In *Proceedings European Conference on Artificial Intelligence, ECAI-88*, 1988, pp. 619-624.

Treur, J., Declarative functionality descriptions of interactive reasoning modules, In H. Boley, M.M. Richter (eds.) Proceedings International Workshop on Processing of Declarative Knowledge, PDK'91, *Lecture Notes in Artificial Intelligence*, vol. 567, Springer-Verlag, 1991, pp. 221-236.

Treur, J., Temporal Semantics of Meta-Level Architectures for Dynamic Control of Reasoning. In: L. Fribourg, F. Turini (ed.), Logic Program Synthesis and Transformation - Meta-Programming in Logic (Proc. of the Fourth International Workshop on Meta-Programming in Logic, META'94), Lecture Notes in Computer Science, vol. 883, Springer Verlag, 1994, pp. 353-376.

Weyhrauch, R.W., Prolegomena to a theory of mechanized formal reasoning, Artificial Intelligence 13 (1980), p 133-170.