RAPID BEHAVIOUR MODELLING FOR AN AGENT-BASED SIMULATION

Sascha A. Goldner
EADS Deutschland GmbH, Defence and Security, Landshuter Straße 26, Unterschleißheim, Germany

Keywords: Agent- based simulation, Behaviour, Business process, Case-based reasoning, Critical infrastructure, Expert system, Human behaviour, PECS, Process modelling, Security, Simulation.

Abstract: Agent-based modelling and simulation has been applied to many different domains for studying highly complex systems. Usually these contain many different entities with their own specific behaviour patterns. The primary strength of agent-based simulation is to model and analyse human behaviour. In this context, one of the most complex and time-consuming tasks is the implementation of behavioural models for the human-like agents. In order to reduce this effort two additional methodologies are taken into consideration and applied to the agent-based model. Business process modelling and case-based reasoning is used for a rapid development of the behavioural part of an agent. This paper describes the scientific goals, ongoing work and interim results of the approach using the security system of an airport as an example.

1 INTRODUCTION

One of the critical infrastructures of modern society is air transport, with airports being both its operational bases and potential targets of terrorist attacks. Past and recent security incidents at international airports show that new and innovative methodologies are needed in order to improve airport security. This task is often just tackled by the implementation of new technology without assessing the effectiveness of the security measures as a whole. Besides security technologies (e.g. scanners, CCTV, etc.), business rules and regulations also have to be considered. Moreover, the many different involved authorities work by experience and implicit knowledge with regards to their organisation specific guidelines for decision making. Especially the process of decision making is highly influenced by human factors and therefore the need arises to consider these factors in detail. Our research focuses on the modelling and simulation of an airport with its entire infrastructure, users and business processes with a special focus on the security relevant procedures.

The first part of the paper will shortly introduce the used methodologies which are agent-based models, business process modelling and case-based reasoning. The second part will describe how these methodologies are used in the context of behaviour creation for agent-based models.

2 AGENT-BASED MODELS

An agent can be seen as an entity that can perceive its environment through sensors and (inter-) act within the environment in a goal-oriented way by effectors (Russell and Norvig, 2003). The most important characteristics about agents are that they consist of complex behavioural properties, such as: (i) they are autonomous and not passive, and (ii) able to interact through exchange of messages and not by explicit task invocation (Wagner, 2003). Because of these characteristics agents are widely used for modelling real world behaviour and especially human behaviour as they act as virtual representatives of the real world entities. Agent-based models are capable to represent the non-linear effects triggered by the behaviour of individuals and their influence on their environment, respectively on other individuals.

A basic principle for the application of agent technology and for valid agent models is to have a structural and behavioural similarity with the original system. The effect on the design of agents is that they have to be constructed with respect to their structure and behaviour in a way, which makes them

Goldner S..
RAPID BEHAVIOUR MODELLING FOR AN AGENT-BASED SIMULATION .
DOI: 10.5220/0003184402570262
In *Proceedings of the 3rd International Conference on Agents and Artificial Intelligence* (ICAART-2011), pages 257-262
ISBN: 978-989-8425-41-6

similar to their empirical counterparts. In case an agent is used for modelling a human being, the agent has to feature all the properties and behavioural patterns of the real human which are relevant in the given scenario. Human behaviour in agent-based systems of social systems is often reduced to cognitive abilities and cognitively controlled actions. Human beings are often modelled as purely rational decision makers. One of the most known and commonly used approaches is the BDI methodology. By using BDI modelling, the agent is provided three mental states: belief, desire and intention. Rao and Georgeff (1995) provide a very comprising and descriptive introduction to the BDI methodology. But one of the biggest weaknesses of the BDI methodology is that it uses rational decision-making in agents as an assumption. The view of human beings as rational decision makers who are perfectly informed and maximise an exogenously given utility function turns out to be too restrictive.

With the increasing complexity of models for human beings the demands made on the design methodology for agent-based simulation models also rise. There is a need for agents which are able to consist of complex internal conditions as well as provide interactions between physical and psychical processes. The PECS (Physis, Emotion, Cognition, Social Status) reference model (see Figure 1) meets this requirements and extends the concepts for the construction of agents featuring a complex and human-like behaviour.

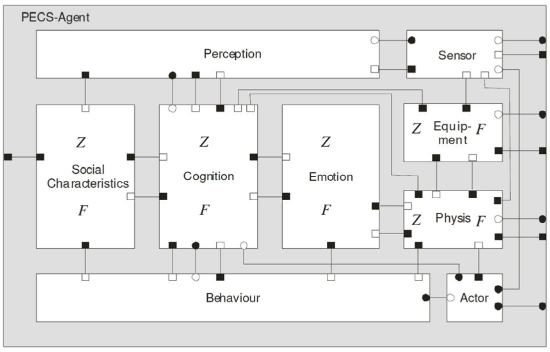

Figure 1: The PECS – agent reference model.

2.1 The PECS Reference Model

PECS is classified as an hybrid architecture (Urban, 2000). The meaning of an hybrid architecture is that the architecture supports modelling of reactive and deliberate agent modes as well. Thus, compared to the aforementioned BDI method that only supports static pre-programmed beliefs and desires, an agent modelled with PECS is able to expand its knowledge base while acting in the environment and pursue agent created plans.

The PECS reference model complies with two major design principles. The first one relates to the structuring of models and is called component-oriented, hierarchical modelling (Urban, 2000). According to this principle it is possible to functionally decompose complex models into a set of smaller model components. Each model component is responsible for modelling a special part of the required functionality and may be connected to other model components. Doing so allows generating more complex components on a higher level of abstraction. This principle leads to modular, clearly structured and well understandable models.

The second principle applies to the description of attributes and model behaviour. PECS follows a system-theoretic approach (Urban, 2000). Every component is characterised by an internal state which is defined by the current values for the given set of model quantities at each calculated point in time. This internal state may be influenced by a time-dependent input and also an output may be produced according to the given dynamic behaviour. For the dynamic behaviour of a model component time-continuous as well as time-discrete state transitions may be specified. This system-theoretic approach leads to a comfortable handling of complex internal states and state transitions and is therefore especially useful for the description of agents which are strongly influenced by complex internal processes. A complete description of the PECS architecture can be found in Urban and Schmidt (2001).

The next chapter focuses on a graphical notation to describe processes. As mentioned before, the behaviour component contains pre-defined rules to trigger certain actions of the agent. Theses rules represent mainly universal or nominal behaviour patterns of an agent which indicate what "essential tasks have to be executed in order to reach a specific goal". Therefore a business process notation is used in our approach to specify this pre-defined and universal behaviour patterns.

3 BUSINESS PROCESS MODELLING

Business process modelling can be considered as a subset of the business process management discipline. Gadasch (2003) defines a business process as "goal-oriented, chronological sequence of tasks which can be executed by different organizations or organizational units using information and communications technologies". A business process is used to produce certain results or services in order to reach the process goals which comply to the company's overall strategy. A very important aspect in the context of business processes is the consideration of legal framework requirements because most of the processes have to comply with regulatory constraints and are thereby significantly determined by these statutory regulations.

In general, business processes are described hierarchical in different levels of detail and from different perspectives - depending on the use case for example. A very abstract description of business process usually gives a good overview and impression what core processes exist at all, what actors are involved, what strategic goals should be reached and what different processes interact with each other (Enterprise and strategic level). A very detailed description instead shows a single process in its elementary steps which can be executed by a single actor to reach the elementary sub-goals and thereby allows a deep analysis (operational process model). Within our approach we decided upon four levels of detail for the business process models (see Figure 2). Each level is characterized by certain syntactical requirements regarding for example the number of tasks in a single process, the number of actors or process lanes, communications and data flows and so on.

In the context of airport security one of the first tasks is to capture the core business processes of the airport system with its actors, process goals, interfaces and process interactions in order to create an overall understanding of the system. For an airport operator these are for example passenger, baggage and cargo handling. One process step for the airport operator within the passenger handling process on strategic level is to ensure the "Departure" of passengers.

During a next step these process descriptions are detailed with focus on the security relevant aspects. The activity "Departure" for example can be broken down into security relevant activities like "Check-In Counter", "Security Check Point" and "Boarding". The operational level then describes the process tasks with all the elementary steps and sub-goals that have to be executed by a single user.

At this point it has to be decided which business process modelling notation should be used in order to operate most suitably along the given use case.

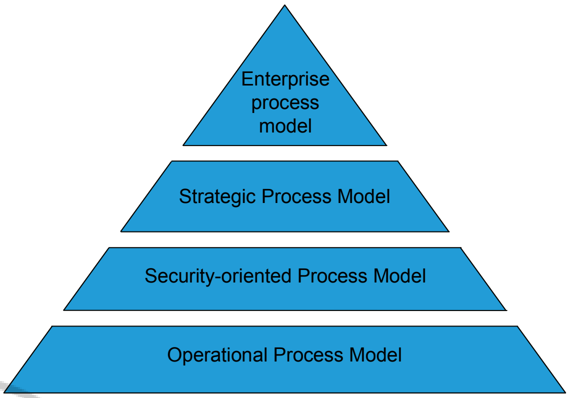

Figure 2: Business process hierarchy.

3.1 Visual Representation and Modelling Language

One of the major requirements for a process modelling notation in our relevant use case is the capability of describing not only the activities of the entities of the airport system but also their interactions and communications. Hence the modelling notation has to meet the following requirements:

- Process execution dependency: processes can be mutually dependent on each other. That is that the execution of one process has to stop until the input of the other process arrives and follow-up actions can be triggered.
- Time- and event-triggered actions: actions should be able to be triggered by events (process results) or time.
- Actor-oriented perspective: the structure of a process model should be defined by the number of participating actors. If for example a process consists of two actors then there should be two process lanes each describing the internal process for each actor. Between the two lanes the information flow among the actors can be displayed. This requirement is particularly important for the combination of agent-based models with business process models because of the identical point of view.
- Modelling of tasks, communication and data flow

The graphical representation of business process information has proven effective for presenting it to different types of users. After comparing different graphical notations for business process modelling (workflow nets, event-driven process chains, process algebras, unified modelling language, petri nets) we decided to use the Business Process Modelling Notation (BPMN). The BPMN 2.0 standard provides a graphical notation which is more expressive than the other mentioned notations. Furthermore it meets our specific requirements defined above. The processes in the BPMN are created actor-oriented – the process of every actor is described in a single pool or lane. Further the perspective is nearly identical with the perspective of creating agent-based models. Each agent can be represented by a single process lane. The process steps within a single lane describe the internal behaviour of an agent. The communication of the agent with other agents or the environment is represented by message and data flows to other lanes in the process model.

Also the business process hierarchy (see Figure 2) can be implemented due to the support of nested processes in the BPMN. Processes on the strategic level simply can be expanded until the level of detail reaches the elementary level and vice versa.

The requirement for time- and event-triggered actions is also covered by the support of different types of events such as exceptions, error or time events. Using these possibilities a business process model that usually describes the "happy path" can easily be enriched by special cases, for example a breakdown of the IT system that causes all Check-In processes to abort instantly.

BPMN offers an extensive syntax for modelling communication and data flows between different actors. Therefore it is sufficient to describe the communication of the agents in the agent-based model as well.

The two described methodologies, agent-based models and business process modelling provide two things so far. First, the business processes define all available procedures that are defined for a system to reach certain business goals. Second, the PECS-architecture defines how the actors responsible for executing the business processes can be simulated as agents in a Multi-Agent-System. Yet, no mechanism has been implemented that triggers certain behaviour for a given specific situation. The individual and autonomous behaviour of the agents is still missing and will be realized by using the approach of case-based reasoning.

4 INTERACTIVE CASE-BASED REASONING

Agnar and Plaza define case-based reasoning (CBR) as a problem solving paradigm that in many respects is fundamentally different from other major artificial intelligence approaches. Instead of relying solely on general knowledge of a problem domain, or making associations along generalized relationships between problem descriptors and conclusions, CBR is able to utilize the specific knowledge of previously experienced, concrete problem situations (cases). That means that a decision that is made in a concrete situation is directly correlated with a huge number of factors which define this particular moment. If this situation with exactly the same influencing factors appears again, the same decision will be made again based on the previous retrieved knowledge

As stated in the previous chapter business process models define the structure of a process and thereby contain all possible paths a process can be executed. However they do not contain the behaviour of the process. No information for decision making is given, meaning there are no rules in the process models which indicate what path has to be taken given a particular situation. That information purely has to be defined in the behavioural model.

One of the biggest problems in the modelling and simulation domain is the gap between the model developer and the domain expert. The developer mostly only implements the model and ensures that the model can be executed in the simulation framework. The domain expert on the other hand has a deep knowledge of the model's behaviour, e.g. given a particular situation the domain expert exactly can tell what decision at what point in the model have to be made in order to create a well and sound behaviour of the model. Finally, to implement an effective simulation system, the knowledge of the domain expert has to be integrated into the simulation model. The reason that behaviour capture is difficult is that the more complex the project, the more tacit knowledge the users possess, and the harder it is to make this knowledge explicit. Tacit knowledge can be defined as knowledge that is not made explicit because it is highly personal, not easily visible or expressible, and usually requires joint or shared activities to transmit it. Users cannot express their tacit knowledge: they do not consciously know all the behaviour that they apply to a situation. What is written into the process models are the commonly occurring rules, logical processes and inevitabilities. Hidden in the user’s tacit knowledge are the multitudes of exceptions that need to be included for the process model to be effective.

To overcome this problem we use an interactive CBR-system. The system does not attempt to make a user's knowledge explicit; instead it captures tacit knowledge in the same way that people learn. It is a widely accepted observation in the area of knowledge acquisition that while a user cannot explain the rules that they use in advance, they can always justify their conclusions when presented with a situation. By asking the domain expert to tell the system what should happen, then asking the domain expert why this should occur, the system builds up the behaviour needed. The domain expert uses the interactive CBR-system to create behaviour, entering conclusions and justifications for that behaviour. When a new situation arises, new behaviour can be added - the user simply tells the CBR-system how to deal with it and justifies their position. The mechanism when creating a rule is called "conclusion and justification". The user not only has to define what decision based on the input data has to be made, but more he has to define why it has to be made.. To justify a conclusion the specific training situation has to be used and therefore it has to be sufficient. The CBR-system determines that the justification is acceptable by checking that the new behaviour is not inconsistent with the previous defined behaviour. The beauty of this process is that the user simply continues to work within their knowledge arena, on their usual tasks, responding to the queries made by the CBR-system.

A complex system contains of many different domains. To capture the overall system behaviour the domain experts of each domain are used to train the CBR-system with their specific knowledge. Thereby the whole system's behaviour is gathered eventually.

5 INTEGRATION

The first part of the paper introduced three separated methodologies. This chapter will describe how business process modelling and case-based reasoning can be used in order to define the behavioural model of the agent-based simulation model. The explanation will use the example of a boarding pass control which takes place at the entrance of a passenger security check point at an airport. To simplify matters we use the following course of events:

- A passenger without carry-on baggage arrives at the security check point.
- The passenger is requested to show the boarding pass to the security employee.
- The security employee decides on the validity of the boarding pass and denies or grants access to the subsequent security procedures.

5.1 The Approach

The definition of the agents for the simulation basically consists of three steps.

First of all the attributes of the PECS agents for the different users at an airport have to be defined. In the example two agents have to be specified. One for the passenger and one for the security employee at the entrance of the security check point performing the boarding pass control. A required attribute in the PECS component 'equipment' of the passenger agent is the boarding pass containing corresponding properties like the date of issue and flight number.

As a second step the business process models for each of the agents have to be provided. The process models describe the universal behaviour for an agent that is the sequence of tasks to reach a specific goal. In the context of the example, a simple process of the security employee would contain the steps "Request boarding pass", "Check boarding pass", "Deny access" and "Grant access". As can be seen in Figure 3, the process of the security employee is triggered by an arriving passenger. Also a data flow can be observer: the boarding pass is handled from the passenger to the security employee and back in case the access is granted.

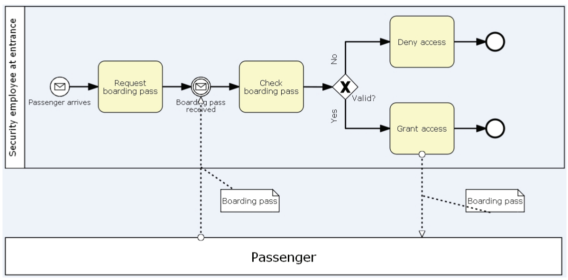

Figure 3: Business process model.

Once the agents and their business process models have been created, the goal of the third step is the definition of situation dependent behaviour for each agent. The basis for this task is the business process models. The process shown in Figure 3 describes the universal process of the security employee. However, it does not contain any information which process path has to be executed given a certain situation or in other word, it does not answer the question "Is the boarding pass of a passenger valid or not?". In order to automatically make a decision in this case, case-based reasoning comes into play. The agents including their process models are imported into the CBR-system. Sticking to the example of the boarding pass control, the behaviour of the security employee has to be specified in more detail. The CBR-system has to be taught by an expert in what situations the boarding pass is considered to be valid or invalid. A required attribute is for example the date on the boarding pass. If this date is identically to today's date, the boarding pass can be considered to be valid. A corresponding rule is then created. If for example the departure time is in four hours and passengers are allow entering the security check point only two hours before, the access has to be denied. Thus, a rule has to be created stating that even the date on the boarding pass is identical to today's date, the access is denied because the departure is greater than two hours.

Once all the necessary rule sets are defined to specify the universal and reactive behaviour of the agents, they can be incorporated into the agents reference model (see Figure 4).

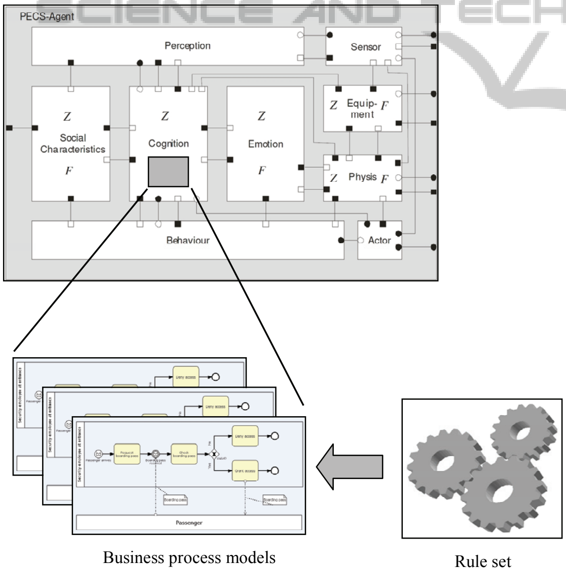

Figure 4: Extension of the PECS agent by business process models and situational behaviour rules.

6 CONCLUSIONS

The combination of agent-based models, business process modelling and case-based reasoning aims to leverage the advantages of each methodology.

The PECS agent architecture provides a structured framework for modelling human-behaviour and contains several advantages compared to the BDI architecture. The downside of current implementations of this architecture is the time and effort that has to be spent in order to implement the behaviour models for the agents. Therefore we apply business process modelling and case-based reasoning that have proven to be very successful in gathering and representing behavioural information.

Business process modelling and especially the combination with agent-based models poses a very interesting field of research because the creation of even very general and universal behaviour patterns in agent-based simulations can be quite complex and time-consuming. Business process modelling instead allows a rapid behaviour modelling due to its very intuitive nature. The recently published BPMN 2.0 standard provides the user with an extensive syntax for creating process models including in addition to the classical task-oriented elements also communication and data flows.

Case-based reasoning and in particular the CBR-system we use has proven to be a very solid approach in gathering expert knowledge – especially the tacit part. This knowledge is interactively gathered from the experts and encapsulated in a rule set that can be accessed during the runtime of a simulation.

These three methodologies are integrated into a single approach in order to create agent-based models with focus on human-behaviour. Our in-house simulation framework shortly will be used to execute these models.

REFERENCES

Agnar, A., Plaza, E., 1994. Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches. In *Artificial Intelligence Communications 7, no. 1 (1994): 39-52.*

Rao, A. S., Georgeff, M. P., 1995. BDI Agents: From Theory to Practice. In *Proceedings of the First International Conference on Multi-Agent Systems (ICMAS)*. AAAI.

Russell, S. and Norvig, P., 2003. *Artificial Intelligence: A Modern Approach*, Prentice Hall, Upper Saddle River, 3rd edition.

Urban, C., 2000. PECS – A Reference Model for the Simulation of Multi–Agent Systems In. *Suleiman R., Troitzsch K. G., G.N. (ed.): Tools and Techniques for Social Science Simulation*, Physica-Verlag.

Urban, C., Schmidt, B., 2001. PECS – Agent-based models of Human Behaviour. In *AAAI Technical Report FS-01-02,* AAAI.

Wagner, G., 2003. The Agent-Object-Relationship Meta-Model: Towards a Unified View of State and Behavior. In *Information Systems, v. 28:5, pp. 475–504.*