

MODELING AND SIMULATION FOR USER ASSISTANCE IN SMART ENVIRONMENTS

Alexander Steiniger
Adeline M. Uhrmacher

University of Rostock
Albert-Einstein-Str. 21
Rostock, 18059, GERMANY

ABSTRACT

Smart environments are slowly but surely entering our everyday life. Their design provides many challenges. Not only heterogeneous devices acting and interacting in a dynamic environment but also intentions and activities of humans have to be taken into account. Diverse processes are responsible for achieving unobtrusive and pro-active user assistance. Those can be structured into a pipeline of perception, sensor interpretation, intention analysis, strategy synthesis, and actuation. Along this pipeline we analyze specific possibilities and requirements for modeling and simulation.

1 INTRODUCTION

Profiting from technological advances in different areas, such as energy supply, communication technology, and sensor technology, smart environments have become of major interest in research and industry over the last years. Smart environments can be considered as an application of the ubiquitous or pervasive computing paradigm, which are inspired by Weiser's vision of ubiquitous computers supporting people in their everyday lives in an unobtrusive manner (Weiser 1999).

Cook and Das define a smart environment as *one that is able to acquire and apply knowledge about an environment and also adapt to its inhabitants in order to improve their experience in that environment* in (Cook and Das 2004). This definition rather reflects the fundamental capabilities of a smart environment than describes the environment itself. A smart environment can be considered as a physical environment where actors perform tasks cooperatively while being assisted by mobile and stationary devices (Wurdell et al. 2009). The devices thereby operate as a smart ensemble in order to accomplish their tasks and support the users. According to Cook and Das the basic features and characteristics of smart environments are: remote control of devices, device inter-communication, information acquisition by sensors, enhanced service by intelligent devices, e.g., by pro-active service provision, predictive and decision-making capabilities (Cook and Das 2004).

Thus, smart environments are concurrent, heterogeneous, and dynamic systems that interact with their physical environment and users continuously. As they shall act autonomously and pro-actively, the agent metaphor fits nicely (Das and Cook 2006, Uhrmacher and Weyns 2009). Correspondingly, the relations between smart environments and simulation reflect those between agents and simulation. As simulation is used to support the design of agent and multi-agent systems, simulation can be used to help designing and developing smart environments. As simulation is a means of deliberating about the future state of the world, agents can also exploit simulation for deliberation (Lee and Fishwick 1997). Similarly, smart environments try to predict future behavior of users by exploiting and analyzing behavior models (Burghardt and Kirste 2009). The latter implies that the simulation is used on-line during the operation of the smart environment. We will focus in the following on using simulation for supporting the design and development of smart environments.

2 ENTITIES OF INTEREST IN SMART ENVIRONMENTS

Smart environments are supposed to provide unobtrusive support to its users. Therefore, devices are required that offer this support. The problem of being open and dynamic leads to programming metaphors that realize such a flexible support based on a set of changing devices, i.e., services. The functioning of smart environments depends on perceiving, thus sensors are key players to correctly interpret the situation. The latter of which comprise the physical environment

and the users. A correct interpretation of users and their possible intentions is of utmost importance, thus considering users' intentions plays a central role in designing and developing smart environments, and correspondingly also in modeling and simulation. In the following, we discuss shortly the entities that should be considered in modeling and simulation.

- **Users:** According to the application context of smart environments, they are intended for various numbers of users. Das and Cook distinguish between a single inhabitant and a more challenging multiple inhabitant case in (Das and Cook 2006). However, there is usually more than one user acting in a smart environment and interacting with it.
- **Devices:** The devices composing a smart environment are interconnected and vary in size, purpose, and computational capabilities. (Weiser 1999) distinguishes between three types of devices: tabs, pads, and walls. In modern smart environments, devices can be more versatile depending on the application context. For instance, Notebooks, PDAs, Smart Phones, Beamers, and Webcams are typical devices of a smart meeting room. Increasingly, devices are integrated that no longer depend on explicit interaction with the users.
- **Sensors:** Although sensors are devices too, they are usually distinguished from other devices embedded in a smart environment in order to emphasize their essential role. Tilak et al. define a sensor as a *device that implements physical sensing of environmental phenomena and reporting of measurements (through wireless communication)* in (Tilak et al. 2002). He underlines sensors as a salient feature of smart environments.
- **Services:** The devices of smart environments provide diverse services. A service can access and/or change the state of the environment (Marquardt and Uhrmacher 2009a). The number and types of services offered by a device depend on the type of the device and its purpose.
- **Physical environment:** A smart environment itself is embedded into a physical environment consisting of walls, doors, stairways, or windows, e.g., a room or building. In addition to the devices embedded in a smart environment, a physical environment usually contains further physical objects without computing and communicational capabilities. Those objects can be obstacles to the users or they can be part of the users' interactions or activities, e.g., a chair, and they can exhibit some behavior of their own, e.g., having a dog in a meeting.

The entire smart environment can be considered as a single deliberative and intelligent agent (Das and Cook 2006). However, also the entities composing the smart environment, e.g., the users (Urban and Schmidt 2001), the environment itself (Bouron et al. 1990), and the devices (Lesser et al. 1999) can be modeled as agents. Therefore, the smart environment can be viewed as a multi-agent system (Cook 2009), i.e., as a community of multiple interacting agents.

Smart environments are characterized by diverse interactions. Interactions exist between users and devices, users and the physical environment, devices and the physical environment, between users, between devices, and in the physical environment itself. In the context of interaction, sensors and services considered as devices too. Interaction between users and between devices can be complex including negotiation, competition, and collaboration. All entities and all types of interactions have to be taken into account when modeling and simulating a smart environment. However, at which level of detail these entities and interactions are represented varies depending on the objective of the simulation study. This objective depends on the phase of the user assistance pipeline that the software should support. We will explain the role of the user assistance pipeline below.

3 USER ASSISTANCE IN SMART ENVIRONMENTS

Unobtrusive and pro-active user assistance is a central goal of smart environments, e.g., smart meeting rooms or ambient home environments. Automated execution of services is one possibility to assist a user unobtrusively, because the number of explicit interactions required is reduced by automation.

Figure 1 depicts a possibility to assist users in smart environments based on the goal-based user assistance introduced and described in (Heider and Kirste 2002, Hellenschmidt and Wichert 2005, Reisse et al. 2008). Processes that are important for this kind of user assistance are: perception (by sensors), sensor interpretation, intention analysis, strategy synthesis, and execution of services (actuation). In the original model of Heider et al., the interpretation of sensor data is part of the intention analysis. In contrast, we consider the sensor interpretation as a separate process in order to emphasize the role of interpreting sensor data correctly. This also corresponds to the traditional, abstract BDI (Belief-Desire-Intention) agent architecture (Rao and Georgeff 1992), where updating the beliefs by interpreting the sensors as a prerequisite to develop new intentions are separately modeled. However, the different processes compose some kind of pipeline, which is executed repeatedly in order to provide continuous assistance to the users. We call this pipeline the user assistance pipeline in the following.

The data generated due to perceiving the physical environment and its inhabitants via sensors deployed in the environment represents the basis for any reaction of the smart environment. These data is interpreted and used for updating the current beliefs. In this context, users' intentions are considered as the users' future actions or activities. Given the

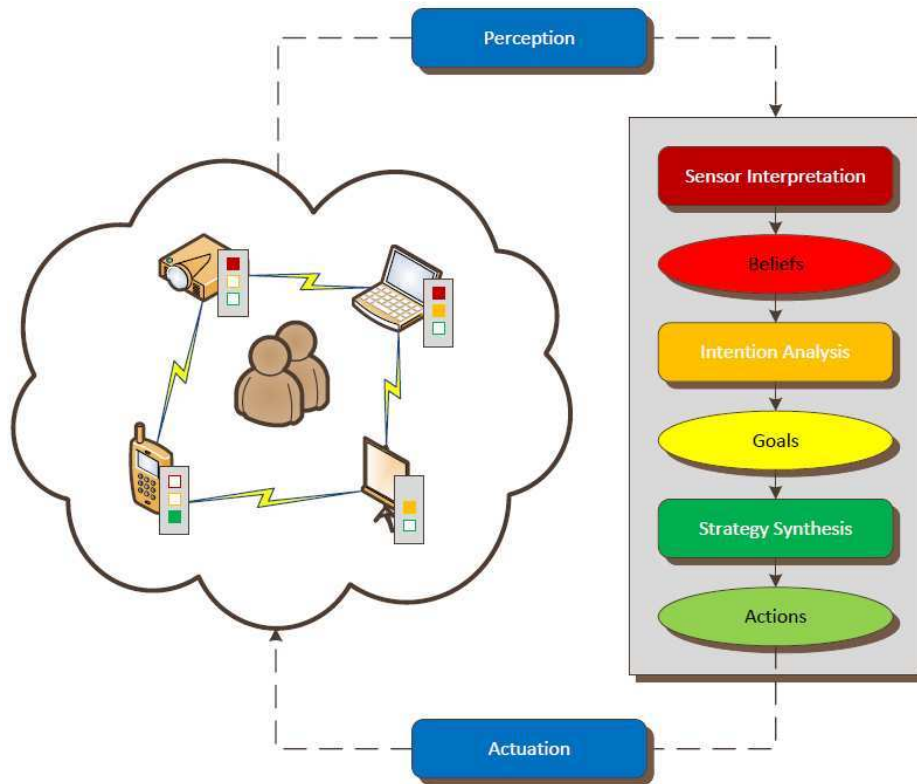


Figure 1: User assistance in smart environments.

beliefs about the current state of the world and the possible intentions of the users, the task of the strategy synthesis is to come up with a service that supports users in their intentions. Often this requires the composition of numerous services, this can be done, e.g., by exploiting AI (Artificial Intelligence) planning algorithms (Marquardt and Uhrmacher 2008). As a result a service composition is created defining a (partial) order for executing services and can be specified as a plan. Finally, the services are executed on the corresponding devices.

Each of the processes shown in Figure 1 is usually executed on a single device, each of which can execute more than one of these processes. Although, we typically have a sequential flow of control between these processes, it might be that processes are started concurrently on different devices and the first that completes the tasks wins, e.g., for strategy synthesis different planners are started to compose services, and the first completed plan is taken for being executed (Marquardt and Uhrmacher 2009a).

Which process is executed on which device depends on various factors, such as computing power or energy supply. Thus, the devices responsible for executing a process might change during the operation of a smart environment. Some devices, e.g., simple sensor nodes, are not capable of or intended for executing these processes at all apart from the perception in case of sensors.

Each of these processes can be related to one or more of the entities mentioned above. Table 1 shows the relations between processes and the entities that are mainly in the focus of the corresponding processes.

Table 1: Relation between processes and the entities the processes focus on.

Process	Entities
Perception	Sensors and users
Sensor Interpretation	Sensor, users, and devices
Intention Analysis	Users and devices
Strategy Synthesis	Services and devices
Actuation	Services and devices

This relation is reflected in the simulation approaches that are exploited for supporting the different processes in the user assistance pipeline. In the following, we will discuss this for the different processes.

3.1 COLLECTION AND PROPAGATION OF SENSOR DATA

Sensors are responsible for collecting the data that form the basis for reactions of smart environments in general (Tilak et al. 2002). Thus, their functioning plays a central role in realizing the depicted user assistance pipeline.

In smart environments, usually different types of sensors are deployed and used, such as positioning sensors, tactile sensors, temperature sensor, or video sensors. Sensors can be autonomous or integrated in other devices, such as a microphone integrated in a notebook. The sensors deployed in a smart environment are interconnected in order to transmit and propagate their generated data through (mainly) wireless communication to devices that are interested in this data for further processing.

The field of sensor network and wireless sensor network (WSN) simulation is vast. Due to the connection to computer networks as networks of interconnected and communicating entities simulation of WSNs can be supported by many established network simulators, e.g., ns-2 (<http://www.isi.edu/nsnam/ns>) and its successor ns-3 (<http://www.nsnam.org>). However, specialized simulation environments, e.g., SensorSim (Park et al. 2000), SENSE (Chen et al. 2004), and J-Sim (Sobeih et al. 2005), have been developed in order to address the specific characteristics of WSNs that distinguish them from usual computer networks. The first is an extension of ns-2 that provides additional features for modeling sensor networks. The motivation behind the component-based SENSE is to allow a large range of different models in terms of components in a simulation study (Chen et al. 2004). This is also the motivation for exploiting general modeling formalism for this purpose. For instance, discrete formalisms are extended to take the location and mobility of sensor nodes and their spatial relations, which constraints the interaction of sensors, into account. To those belong Space Time Petri Nets (Luo and Tsai 2005) and SpacePi (John et al. 2008) which are extensions of Petri nets and the pi calculus, respectively. However, SpacePi was introduced in a biological motivated context, but can be used in other domains. Other approaches abstract from the continuous space and focus on the design and operation of the individual sensor nodes such as (Antoine-Santoni et al. 2008b, Antoine-Santoni et al. 2008a). For instance, Antoine-Santoni et al. utilize the Discrete Event System Specification (DEVS) formalism introduced by Zeigler (Zeigler et al. 2000) to model the communication between the different components of a sensor node.

The diversity of nodes and their data play a central role for analyzing different strategies to collect information and propagate data in smart environments. In contrast to sensor networks used in other domains, comparatively few highly specialized sensors are interacting with each other in a smart environment. Usually, little redundancy in the sensor data exists. In typical WSN simulations, users, services, or other devices are described in a rather abstract manner, e.g., in terms of an abstract signal or data model, e.g., a Gaussian probability distribution (Giersich and Kirste 2007) or an abstract language model (Aipperspach et al. 2006). The type of data communicated in the network is of little importance. Simply, the quantity of data or the absence of data is relevant for the simulation. However, all of these aspects might influence the unobtrusive operation of the smart environment. Therefore, flexible simulation approaches that allow taking the diversity of nodes and the type of propagated data into account are particularly suitable.

Communication between different devices and propagation of data is not limited to sensors and sensor data. Therefore, network simulators and network simulation in general can be exploited for evaluating and testing different communication strategies and protocols in smart environments. Thereby, the level of abstraction of the network simulation will reflect the concrete requirements and the diversity of systems to be evaluated in the smart environment, e.g., communication of user goals, distributed planning and execution strategies, or information retrieval strategies supported by distributed data bases or the middleware.

3.2 FROM SENSOR DATA TO INTENTIONS

The next step in the user assistance pipeline is the interpretation of sensor data provided by the sensors and updating the beliefs of the smart environment. Depending on the variety of sensors and their observations, e.g., video streams or location data, sensor data have to be transformed into symbolic knowledge that forms the basis of further reasoning. These are problems smart environments share with robotics and robotic systems (Pollack et al. 2002, Thrun 2003).

Due to the fact that users are in focus of smart environments, perceptions and beliefs refer mainly to users and their activities. In this context, the term activity recognition is often used for the interpretation of sensor data (Aipperspach et al. 2006), whereat the current users' activities are inferred based on the sensor data. For instance, the state change of a ubiquitous sensor attached to the door of the refrigerator in a smart kitchen at a certain time is interpreted as the activity "user opens the refrigerator at time t " (Tapia et al. 2004) encoded in an suitable symbolic notation. Accordingly, simulation systems aim at testing the correct interpretation of data and turning them into "symbolic" beliefs that can be found in this context. Here, valid models of the sensors are crucial to produce plausible data to be interpreted. This supports testing whether incoming sensor data are correctly transformed into beliefs.

In current smart environments the beliefs are focusing on the users. Further information might refer to the state of the devices and some basic background information of the physical system. Referring to information about the users, current approaches typically consider the location of the users (Patterson et al. 2003, Burghardt and Kirste 2009) or the interaction between users and devices (Rao and Cook 2003, Tapia et al. 2004). Sensor data are usually noisy and intermittent (Burghardt and Kirste 2009), particularly location data. If those noisy sensor data is used as data basis,

uncertainties in the interpretation can be a result. Simulation can be used to test to what degree the system can deal correctly with noise in the data.

Once the smart environment knows what its users currently do, the smart environment tries to predict the users' intentions, which are the users' future actions and activities, on basis of his knowledge and beliefs. Therefore, the term action prediction is often used in this context. Predicting users' intentions the smart environment requires an appropriate model of the users' behavior. In this context, a human behavior model represents typical user activities and user intentions in a certain application context. Different application contexts result in different behavior models, e.g., in a smart meeting room typically other activities are possible than in a smart kitchen.

Based on symbolic beliefs and user models a support can be derived in smart environments. For instance, Autominder, an automated agent designed for assisting elderly with cognitive impairments in carrying out required activities of daily life (ADLs), uses those symbolic beliefs to track the execution of the elderly' ADLs and remind them of the next activities according to their ADLs (Pollack et al. 2002). A symbolic belief can be, for instance, the activity "user went to kitchen". The interpretation of sensor data in terms of a transformation to symbolic beliefs is done outside of Autominder.

However, numerous approaches circumvent the interpretation of sensor data as an independent process by integrating the interpretation directly in the intention analysis. Quantitative sensor data are directly used in combination with a human behavior model to infer users' intentions.

Hidden Markov models (Rabiner 1989), shortly HMMs, are a common and well understood approach for modeling human behavior in order to predict future actions of users on basis of noisy and intermittent sensor data and a-priori knowledge of user activities (Burghardt and Kirste 2009). Various extensions to the basic HMM formalism exist that improve the prediction accuracy and expressiveness of the formalism, such as task-based Markov models (Rao and Cook 2003) and hierarchical hidden Markov models (Lühr et al. 2003).

In (Giersich and Kirste 2007) and (Burghardt and Kirste 2009), dynamic Bayesian networks are used, to model the behavior of a team of users reflecting the fact that there are usually multiple users collaborating with each other in a smart environment. A team agenda is used to improve the prediction accuracy and speed. The agenda specifies the activities that have to be performed in order to reach a certain team goal.

For these approaches, simulation can be used in a similar way as for testing the interpretation of sensor data. Data is generated to test the intention analysis strategy. However, it should be noted at this point that the simulation model has to comprise not only a valid sensor data model but also plausible models that relate activities of users with their intentions.

The interesting aspect in testing the strategy of intention analysis by simulation is that user models need to be realized within the simulation, to test the intention analysis that typically includes user models itself (see above). The user model exploited in the simulation should be different from the user model used by the smart environment to infer intentions. At this point it becomes clear that testing via simulation should be done by a different group which develops the simulation model for testing independently of the group developing the smart environment. In (Gierke and Uhrmacher 2005), a virtual testing environment is used to evaluate the reminding strategies of Autominder (Pollack et al. 2002) in order to support elderly in their home. Please note, that this reminding cannot be done unobtrusively, but is another form of user assistance. In the virtual environment used for testing a detailed model of an elderly in tradition of human behavior models (Silverman et al. 2001) is realized. Figure 2 shows the virtual environment including the model of elderly (Gierke and Uhrmacher 2005).

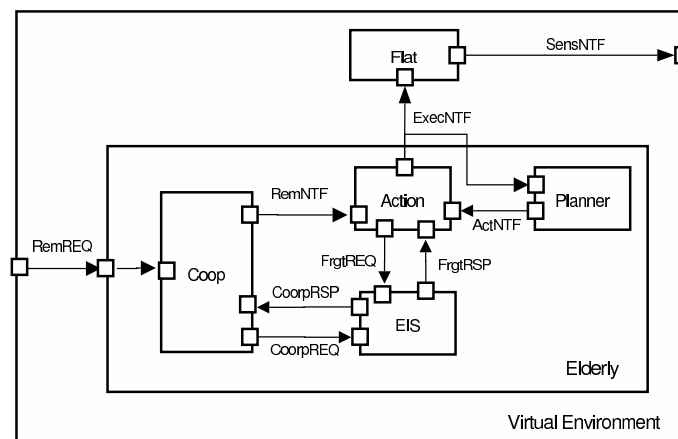


Figure 2: Virtual environment for testing Autominder.

The elderly model depicted in Figure 2 comprises different components each realizing a special functionality. The Cooperation component deals with the reminder acceptance of the elderly. The Action component is responsible for the execution of actions by the virtual elderly. The Planner manages the routine plan the behavior of the virtual elderly is based on. Some actions may be forgotten by the model in order to model cognitive impairments. Furthermore, the model does not have to cooperate with Autominder. The virtual elderly is not capable to interact with other humans in a natural manner, because it is no objective of the model. A central processing unit is taking the history of the elderly' activities into account.

However, the creation of valid and plausible user models that can be used for reliable testing is crucial and challenging. Smart environments are usually intended for multiple users, so interactions between these users have to be addressed by a suitable user model.

Once the smart environment has a notion of the users' intention, it can use this knowledge to decide what next in order to assist the users in their intentions.

3.3 FROM INTENTIONS TO SERVICES

Given the intention of a user, the smart environment has to decide which of the available services have to be executed in which order. These services are provided by and related to the devices embedded in the smart environment. A device usually provides a limited number of services. The kind of services provided depends on the device itself. Furthermore, the smart environment has to decide on which device a service has to be executed. Thus, services with the same effect can be provided by more than one device, e.g., there can be more than one beamer in a smart meeting room.

In (Marquardt and Uhrmacher 2009b), AI planning is used to create service compositions for automated execution of a service sequences. The problem is thereby described as a planning problem, where the intentions of users, usually provided by the intention analysis, are specified as goal state and the services are represented by planning operators with preconditions and effects on the state of the environment. On basis of the available planning operators (services) a planner tries to find a sequence of operators, i.e., plan, which transfers the initial state to the goal state. However, it is not sure if such a plan exists and if it is possible to reach the goal state with the available services. For instance, a device that provides a service required for a plan can fail or be removed from the smart environment.

Furthermore, classical planning cannot react to changes in the environment during planning time, thus changes in the environment typically require a replanning. As a consequence planning should be sufficient fast to support unobtrusive user assistance (Marquardt and Uhrmacher 2009a). Therefore, runtime performance is a crucial aspect of using AI planning for composing services and should be tested. In (Marquardt and Uhrmacher 2008), Simulation is used for evaluating different planners referring to their execution time based on test data generated by a simulation model implemented in the PDEVs formalism. Semantic details of the devices are excluded.

In this context it would be interesting to include a change of devices and services and a change of intentions of the users over time in the simulation environment. The latter can be done by detailed models of user behavior, e.g., (Gierke and Uhrmacher 2005), but often more abstract models of user behavior are sufficient for this purpose, e.g., task models (Giersich et al. 2007).

A decentralized approach for an automated selection and execution of device actions embedded based on user goals can be found in (Reisse and Kirste 2008).

3.4 FROM SERVICES TO EXECUTION

Once the smart environment creates a service composition, it has to inform the corresponding devices to execute the services specified in the composition in the correct order. Therefore, commands have to be communicated to the devices and executed by them.

In the first step the communication is in the focus of consideration. If the communicated commands are considered as abstract data, similar questions arise as in the process of propagating sensor data in WSNs described in Section 3.1. Thus, similar simulation approaches can be used.

When it comes to the execution of services on the devices, the devices are in focus and different aspects have to be taken into account. A device can fail or be removed from the smart environment before a service can be executed on this device. In both cases, the created service composition might become invalid. Thus, strategies for service execution are required in order to decide how the smart environment should react if services fail to be executed. Is it possible to execute the service on another device, can the service that failed be ignored, or is it necessary to create a new service composition in order to reach the specified goal state? In this context, simulation can be used to test the robustness and reliability of those execution strategies in case of service failures and the interplay with human activities.

There are special "simulators" focusing on a sophisticated visualization and animation of smart environments in terms of a 3D virtual environment, e.g., UbiREAL (Nishikawa et al. 2006) or the ResiSim 3D simulator (Youngblood et al. 2005). Those simulators can be used to test the interplay between humans and devices in smart environments. However, this kind of visualization or animation of smart environments does not strictly fall under the term experimentation, but reveals first problems and allows a kind of face validation of different strategies.

4 DISCUSSION

In consideration of the user assistance pipeline described in Section 3 simulation can be and is used for testing and evaluating strategies, models, and concrete implementations on basis of test data in every process of the pipeline apart from the perception by sensors. The test data refer to the input of the corresponding process depicted in Figure 1. According to the process, different types of input data are required reflecting different aspects on different levels of abstractions. Therefore, different simulation models should be used to generate test data. The incorporation or combination of different models could be suitable.

With reference to sensor networks, simulation can be used to evaluate network protocols for the communication of sensor data concerning their energy efficiency. The sensor nodes composing the sensor network and their intercommunication is thereby modeled and simulated. In this context, suitable and valid sensor models are required that represent the functioning of a single sensor node and the wireless communication between multiple sensor nodes. The data that is communicated is typically represented in an abstract manner, because mainly the quantity of data is important for testing protocols. Users and other devices are of little relevance at this point, so they are suppressed in the simulation. However, the physical environment should be taken into account, because it can influence the communication between devices, e.g., a thick wall has influence on the signal strength.

For testing interpretation strategies of sensor data sensors can be modeled reflecting their specifics, e.g., energy restrictions. Sensors that generate sensor data related to a user, e.g., the location of a user, should take user models into account in order to generate plausible data that can be interpreted. A model of the environment should also be considered.

For approaches of intention analysis that take symbolic beliefs as input, beliefs can be directly generated for testing. Sensors and sensor networks do not have to be considered in detail or at all in the simulation. On basis of behavior models, a virtual user can be modeled used for generating beliefs directly, because beliefs about a user refers to his previous and current activities. However, it should be taken into account that noisy sensor data can be interpreted incorrectly resulting in wrong beliefs. A behavior model should reflect this uncertainty. The state of a physical environment can be represented by a simple model of the environment. As mentioned above, there are approaches for analyzing intentions that work directly on sensor data and interpret these data implicitly. For these approaches, test data can be generated in the same way as it is generated for the interpretation of sensor data.

The strategy synthesis requires certain goals that specify the intentions of users in a smart environment. These intentions can be generated by a behavior model as well. In addition to the intentions, devices and the services they provide are of interest. Therefore, models of the devices are necessary that represent device-specific aspects and specify what service is provided by a device. If we think of mobile devices, the device model has to consider locations of the devices and spatial relations between them.

Device models are also required for testing executions strategies. Here, the behavior of devices is in focus.

Independent of the process simulation is used for, valid models are essential for reliable and usable experimentation result. However, a valid model does not have to be plausible inevitably. For instance, users that are walking through a wall are not desired. Therefore, models used for the generation of test data should be designed carefully and verified themselves.

5 CONCLUSION AND OUTLOOK

Diverse modeling and simulation approaches are being applied for designing and evaluating smart environments. A core question of smart environments is how the user can best be assisted.

We adapt the goal-based user assistance by Heider et al. to reflect the different processes that are of importance in realizing pro-active user assistance. We use this user assistance pipeline to structure the different modeling and simulation approaches currently exploited. The pipeline ranges over perception, sensor interpretation, intention analysis, strategy synthesis, and execution (actuation).

At the beginning and at the end of the pipeline, networks and devices are of interest, to evaluate and test the information gathering and execution strategies. Whereas for perception and sensor interpretation work on simulating sensors and sensor networks are exploited for systematic testing and evaluation, the application of modeling and simulation in the context of execution strategies is currently still restricted to animation approaches. In providing support for designing and evaluating the intention analysis and strategy synthesis modeling and simulation abstracts away much of the physical environment and technical details of devices and sensors. In all steps of modeling and simulation models of the user play an important role, the most central one of course in the context of intention analysis. The user models that are exploited range from simple stochastic models to fully fledged Beliefs, Desire, and Intention agent-based models.

All processes of the user assistance pipeline need to be seamlessly integrated for a functioning smart environment. Similarly, it would be important to provide means to integrate the modeling and simulation efforts that currently are exploited locally within this pipeline.

Therefore, a modeling and simulation environment is required that allows to integrate the different types of models at different levels of abstractions. In this regard, component-based approaches as pursued in SENSE (Chen et al. 2004), for the area of sensor networks, or proposed in COMO (Component-based Modeling) (Röhl and Uhrmacher 2008) are a suitable starting point for this research.

In addition, to composing models on demand that might be realized in different formalisms another requirement is a flexible execution mechanism, that supports efficient simulation as well as a software-in-the-loop simulation (Himmelspach et al. 2003). At that point, component-based simulation frameworks, such as JAMES II (Himmelspach and Uhrmacher 2007), can be beneficially exploited.

Currently, we are exploring possibilities for a component-based modeling and simulation framework for smart environments that supports the entire user assistance pipeline. It is planned to evaluate the developed concepts in a concrete setting.

ACKNOWLEDGMENTS

This research is funded by the German Research Foundation (DFG) within the context of the project MuSAMA (Multimodal Smart Appliance Ensembles for Mobile Applications).

REFERENCES

- Aipperspach, R., E. Cohen, and J. F. Canny. 2006, April. Modeling human behavior from simple sensors in the home. In *Pervasive Computing: 4th International Conference*, ed. K. P. Fishkin, B. Schiele, P. Nixon, and A. J. Quigley, Volume 3968 of *Lecture Notes in Computer Science*, 337–348. Berlin, Germany: Springer.
- Antoine-Santoni, T., J.-F. Santucci, E. de Gentili, and B. Costa. 2008a, September. Devs-wsn : A discrete event approach for wireless sensor network simulation. In *The 6th ACS/IEEE International Conference on Computer Systems and Applications*, 895–898. Los Alamitos, CA, USA: IEEE Computer Society.
- Antoine-Santoni, T., J.-F. Santucci, E. de Gentili, and B. Costa. 2008b, February. Discrete event modeling and simulation of wireless sensor network performance. *Simulation* 84 (2-3): 103–121.
- Bouron, T., J. Ferber, and F. Samuel. 1990, August. Mages: A multiagent testbed for heterogeneous agents. In *Proceedings of the 2nd European Workshop on Modelling Autonomous Agents in a Multi-Agent World*, ed. Y. Demazeau and J. P. Müller, 221–239. Amsterdam, The Netherlands: North Holland.
- Burghardt, C., and T. Kirste. 2009, August. A probabilistic approach for modeling human behavior in smart environments. In *Proceedings of the 2nd International Conference on Digital Human Modeling*, ed. V. G. Duffy, Volume 5620 of *Lecture Notes in Computer Science*, 202–210. Berlin, Germany: Springer.
- Chen, G., J. Branch, L. Pflug, Michael J. amd Zhu, and B. K. Szymanski. 2004, November. Sense: A sensor network simulator. In *Advances in Pervasive Computing and Networking* (1st ed.), ed. B. K. Szymanski and B. Yener, Chapter 13, 249–267. Berlin Heidelberg, Germany: Springer.
- Cook, D. J. 2009, January. Multi-agent smart environments. *Journal of Ambient Intelligence and Smart Environments* 1 (1): 51–55.
- Cook, D. J., and S. K. Das. (Eds.) 2004, November. *Smart environments: Technology, protocols and applications*. 1st ed. Wiley Series on Parallel and Distributed Computing. Hoboken, NJ, USA: John Wiley & Sons.
- Das, S. K., and D. J. Cook. 2006. Designing and modeling smart environments (invited paper). In *WOWMOM '06: Proceedings of the 2006 International Symposium on World of Wireless, Mobile and Multimedia Networks*, 490–494. Washington, DC, USA: IEEE Computer Society.
- Gierke, M., and A. M. Uhrmacher. 2005. Modeling elderly behavior for simulation-based testing on agent software. In *Proceedings of the Conceptual Modeling and Simulation Conference*, ed. F. Barros, A. Bruzzone, C. Frydman, and N. Giambiasi, 159–164: SCS.
- Giersich, M., P. Forbrig, G. Fuchs, T. Kirste, D. Reichart, and H. Schumann. 2007, July. Towards an integrated approach for task modeling and human behavior recognition. In *Proceedings of the 12th International Conference on Human-Computer Interaction*, ed. J. A. Jacko, Volume 4550 of *Lecture Notes in Computer Science*, 1109–1118. Berlin, Heidelberg, Germany: Springer.
- Giersich, M., and T. Kirste. 2007, November. Effects of agendas on model-based intention inference of cooperative teams. In *Proceedings of the 2007 International Conference on Collaborative Computing: Networking, Applications and Worksharing*, 456–463. Washington, DC, USA: IEEE Computer Society.
- Heider, T., and T. Kirste. 2002, July. Supporting goal-based interaction with dynamic intelligent environments. In *Proceedings of the 15th European Conference on Artificial Intelligence*, ed. F. v. Harmelen, 596–600. Amsterdam, The Netherlands: IOS Press.
- Hellenschmidt, M., and R. Wichert. 2005. Goal-oriented assistance in ambient intelligence. In *Workshop on Experience Research in Ambient Intelligence*.

- Himmelspach, J., M. Röhl, and A. M. Uhrmacher. 2003, December. Simulation for testing software agents - an exploration based on james. In *Proceedings of the 2003 Winter Simulation Conference*, ed. S. Chick, P. Sánchez, D. Ferin, and D. Morrice, 799–807. Los Alamitos, CA, USA: IEEE Computer Society.
- Himmelspach, J., and A. M. Uhrmacher. 2007. Plug'n simulate. In *Proceedings of the 40th Annual Simulation Symposium*, 137–143. Washington, DC, USA: IEEE Computer Society.
- John, M., R. Ewald, and A. M. Uhrmacher. 2008, January. A spatial extension to the pi calculus. *Electronic Notes in Theoretical Computer Science* 194 (3): 133–148.
- Lee, J. J., and P. A. Fishwick. 1997, December. Simulation-based planning for multi-agent environments. In *Proceedings of the 29th Conference on Winter Simulation*, ed. S. Andradttir, K. J. Healy, D. H. Withers, and B. L. Nelson, 405–412. Washington, DC, USA: IEEE Computer Society.
- Lesser, V., M. Atighetchi, B. Benyo, B. Horling, A. Raja, R. Vincent, T. Wagner, P. Xuan, and S. X. Zhang. 1999, January. A multi-agent system for intelligent environment control. Computer science technical report, University of Massachusetts.
- Lühr, S., H. H. Bui, S. Venkatesh, and G. A. W. West. 2003, April. Recognition of human activity through hierarchical stochastic learning. In *Proceedings of the First IEEE International Conference on Pervasive Computing and Communications*, 416–422. Los Alamitos, CA, USA: IEEE Computer Society.
- Luo, Y., and J. J. P. Tsai. 2005, December. A graphical simulation system for modeling and analysis of sensor networks. In *Proceedings of the Seventh IEEE International Symposium on Multimedia*, ed. S. Kawada, 474–482. Los Alamitos, CA, USA: IEEE: IEEE Computer Society.
- Marquardt, F., and A. M. Uhrmacher. 2008, December. Evaluating ai planning for service composition in smart environments. In *Proceedings of the 7th International ACM Conference on Mobile and Ubiquitous Multimedia*, ed. A. Zaslavsky and M. Wiberg, 48–55. New York, NY, USA: SIGMOBILE: ACM.
- Marquardt, F., and A. M. Uhrmacher. 2009a. An ai-planning based service composition architecture for ambient intelligence. In *Workshop Proceedings of the 5th International Conference on Intelligent Environments*, ed. M. Schneider, A. Kröner, J. E. Alvarado, A. G. Higuera, J. Augusto, D. Cook, V. Ikonen, P. Cech, P. Mikulecktabacckkludgey, A. Kameas, and V. Callaghan, Volume 4 of *Ambient Intelligence and Smart Environments*, 145–152. Amsterdam, Berlin, Tokyo, Washington DC: IOS Press.
- Marquardt, F., and A. M. Uhrmacher. 2009b, December. Creating ai planning domains for smart environments using pddl. In *Proceedings of International Conference on Intelligent Interactive Assistance and Mobile Multimedia Computing*, ed. D. Tavangarian, T. Kirste, D. Timmermann, U. Lucke, and D. Versick, Volume 53 of *Communications in Computer and Information Science*, 263–274. Berlin Heidelberg, Germany: Springer.
- Nishikawa, H., S. Yamamoto, M. Tamai, K. Nishigaki, T. Kitani, N. Shibata, K. Yasumoto, and M. Ito. 2006, September. Ubireal: Realistic smartspace simulator for systematic testing. In *Proceedings of the 8th International Conference on Ubiquitous Computing*, ed. P. Dourish and A. Friday, Volume 4206 of *Lecture Notes in Computer Science*, 459–476. Berlin / Heidelberg, Germany: Springer.
- Park, S., A. Savvides, and M. B. Srivastava. 2000, August. Sensorsim: A simulation framework for sensor networks. In *Proceedings of the 3rd ACM International Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, 104–111. New York, NY, USA: ACM.
- Patterson, D. J., L. Liao, D. Fox, and H. Kautz. 2003, November. Inferring high-level behavior from low-level sensors. In *Proceedings of the 5th International Conference on Ubiquitous Computing*, ed. A. K. Dey, A. Schmidt, and J. F. McCarthy, Volume 2864 of *Lecture Notes in Computer Science*, 73–89. Berlin Heidelberg, Germany: Springer.
- Pollack, M. E., C. E. McCarthy, I. Tsamardinis, S. Ramakrishnan, L. Brown, S. Carrion, D. Colbry, C. Orosz, and B. Peintner. 2002, March. Autominder: A planning, monitoring, and reminding assistive agent. In *Proceedings of the 7th International Conference on Intelligent Autonomous Systems*, ed. M. Gini, W.-M. Shen, C. Torras, and H. Yuasa. Amsterdam, The Netherlands: IOS Press.
- Rabiner, L. R. 1989, February. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE* 77 (2): 257–286.
- Rao, A. S., and M. P. Georgeff. 1992, October. An abstract architecture for rational agents. In *Proceedings of the 3rd International Conference on Principles of Knowledge Representation and Reasoning*, ed. B. Nebel, C. Rich, and W. R. Swartout, 439–449. Cambridge, MA, USA: Morgan Kaufmann.
- Rao, S. P., and D. J. Cook. 2003. Identifying tasks and predicting actions in smart homes using unlabeled data. In *Proceedings of the Machine Learning Workshop on The Continuum from Labeled to Unlabeled Data in Machine Learning and Data Mining*.
- Reisse, C., C. Burghardt, F. Marquardt, T. Kirste, and A. M. Uhrmacher. 2008, December. Smart environments meet the semantic web. In *Proceedings of the 7th International Conference on Mobile and Ubiquitous Multimedia*, ed. M. Wiberg and A. B. Zaslavsky, 88–91. New York, NY, USA: ACM.
- Reisse, C., and T. Kirste. 2008, July. A distributed action selection mechanism for device cooperation in smart environments. In *Proceedings of the 4th IET International Conference on Intelligent Environments*, 1–8.

- Röhl, M., and A. M. Uhrmacher. 2008, December. Definition and analysis of composition structure for discrete-event models. In *Proceedings of the 2008 Winter Simulation Conference*, ed. S. J. Mason, R. R. Hill, Mönch, O. Rose, T. Jefferson, and J. W. Fowler, 942–950. Los Alamitos, CA, USA: IEEE Computer Society.
- Silverman, B. G., R. Migh, R. Dubois, H. Shin, and M. Johns. 2001. Toward a human behavior models anthology for synthetic agent development. In *Proceedings of the 10th Conference on Computer Generated Forces and Behavioral Representation*, 277–285.
- Sobeih, A., W. P. Chen, J. C. Hou, L. C. Kung, N. Li, H. Lim, H. Y. Tyan, and H. Zhang. 2005, April. J-sim: A simulation environment for wireless sensor networks. In *Proceedings of the 38th Annual Simulation Symposium*, 175–187. Los Alamitos, CA, USA: IEEE Computer Society.
- Tapia, E. M., S. S. Intille, and K. Larson. 2004, May. Activity recognition in the home using simple and ubiquitous sensors. In *Pervasive Computing: Second International Conference*, ed. A. Ferscha and M. Friedemann, Volume 3001 of *Lecture Notes in Computer Science*, 158–175. Berlin, Germany: Springer.
- Thrun, S. 2003. Robotic mapping: A survey. In *Exploring Artificial Intelligence in the New Millennium* (1st ed.), ed. G. Lakemeyer and B. Nebel, The Morgan Kaufmann Series in Artificial Intelligence, Chapter 1, 1–35. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.
- Tilak, S., N. B. Abu-Ghazaleh, and W. R. Heinzelman. 2002, April. A taxonomy of wireless micro-sensor network models. *Mobile Computing and Communications Review* 6 (2): 28–36.
- Uhrmacher, A. M., and D. Weyns. (Eds.) 2009, June. *Multi-agent systems: Simulation and application*. 1st ed. Computational Analysis, Synthesis, and Design of Dynamic Systems. Boca Raton, FL, USA: CRC Press.
- Urban, C., and B. Schmidt. 2001, November. Pecs - agent-based modelling of human behavior. In *Emotional and Intelligent II: The Tangled Knot of Social Cognition*, 141–146. Menlo Park, CA, USA: AAAI Press.
- Weiser, M. 1999, July. The computer for the 21st century. *Mobile Computing and Communications Review* 3 (3): 3–11.
- Wurdel, M., C. Burghardt, and P. Forbrüg. 2009, December. Making task modeling suitable for smart environments. In *SEACUBE Workshop in conjunction with the Proc. of the International Conference on Ultra Modern Communications (ICUMT 2009)*, 1–6. Los Alamitos, CA, USA: IEEE Computer Society.
- Youngblood, G. M., D. J. Cook, and L. B. Holder. 2005, October. Seamlessly engineering a smart environment. In *Proceedings of the 2005 IEEE International Conference on Systems, Man and Cybernetics*, Volume 1, 548–553. Washington, DC, USA: IEEE Computer Society.
- Zeigler, B. P., H. Praehofer, and T. G. Kim. 2000, January. *Theory of modeling and simulation*. 2nd ed. San Diego, CA, USA: Academic Press.

AUTHOR BIOGRAPHIES

ALEXANDER STEINIGER holds a diploma in Computer Science from the University of Rostock and pursues a PhD at the Modeling and Simulation Group at the University of Rostock. His main research interests are smart environments, component-based modeling and simulation, and human behavior modeling. His email address is alexander.steiniger2@uni-rostock.de.

ADELINDE M. UHRMACHER is a Professor at the Department of Computer Science at the University of Rostock and head of the Modeling and Simulation Group. Her research interests are in modeling and simulation methodologies and their applications. Her e-mail and web addresses are lin@informatik.uni-rostock.de and www.informatik.uni-rostock.de/~lin.