

How Agents Use Breadcrumbs to Find Their Way

Christian Brandstaetter*, Samer Schaat, Alexander Wendt, Martin Fittner

Institute of Computer Technology at Vienna University of Technology, Gußhausstraße 27-29, 1040 Vienna.

* Corresponding author. Email: brandstaetter@imp.ac.at

Manuscript submitted October 5, 2015; accepted December 30, 2015.

doi: 10.17706/jcp.12.1.89-96

Abstract: An autonomous agent must know where it is, which objects are in the nearby surrounding area and where it can find already seen, far away objects. On the basis of the existing cognitive architecture Simulation of Mental Apparatus and Applications (SiMA) at the Institute of Computer Technology (ICT) we show, how cognitive architectures can get extended by location awareness and navigation. This paper deals with the problem of self-localization and path finding. Therefore, landmarks, locations and regions will be used. Actions lead the agent from one location to another. Locations are hierarchically grouped together to regions. The functionality was implemented exemplary in a multiagent simulation using simplified scenarios. The results confirm the prospects. Now we can start to introduce the functionality in our full featured main simulator.

Key words: Artificial general intelligence, artificial recognition system, cognitive architectures, cognitive automation, landmarks, localization, navigation, psychoanalytically inspired AI, software agents.

1. Introduction

As well as it is challenging for a child to avoid becoming lost in the shopping center, simulated agents need some kind of localization. It is a general problem that agents without location awareness are less efficient in its operation than agents with knowledge about their own location. For example automatic lawn mowers [1] with a positioning system can accomplish its task much faster than mowers which move and turn only randomly. Hence, a clear demand exists for a method to put the perception and experiences into locational relation to each other. Humans would not be that successful with a lack of localization and navigation skills. We discuss in this paper how a psychoanalytically inspired agent could solve the problem of localization and navigation.

Following a bionic approach, Psychoanalysis is used as a framework, since it provides a functional model of the Artificial General Intelligence (AGI) approach. Technically the Simultaneous Localization and Mapping (SLAM) Algorithm solves this problem of localization. But in our special scope we cannot use these methods, because they are not compatible with the way the human mind operates. In our model there are no exact distances or angles. Relations are rather encoded relatively. Details can be found in the section "The SiMA Project". Therefore we have to find alternatives for these methods. We extend an existing multi-agent system by the ability to fulfill tasks of localization and navigation in the style of a human being. A data structure is developed to store navigational data with backing to the existing data structures in the SiMA project [2].

2. State of the Art

Currently there are lots of algorithms for mapping, selflocalization (Kalman filter, Monte Carlo localization,

Particle filter) and path finding (Dijkstra's algorithm, A* search algorithm). But they fit only very limited into the approach of a psychoanalytically inspired autonomous agent given in the SiMA project after the theory of Sigmund Freud.

2.1. Algorithms for Mapping and Self-Localization

Kalman filter: Robots use lots of sensors to measure characteristics of an area. In outdoor environment Global Positioning System (GPS) and a compass is used to determine the position of the robot. For indoor environments cameras, sonars or laser scanners are used. All these sensors are afflicted by measurement errors. To get values which can be used for calculating the robots position these measurement errors must be compensated. Kalman filters [3] can be used for this purpose.

Monte Carlo localization: To estimate a robot's current pose from ultrasonic or laser range sensors data the Monte-Carlo Localization (MCL) [4], a very popular sampling-based localization algorithm can be used. This localization technique is an enhancement of the known Monte Carlo method originally developed for object tracking. The MCL applies the conditional density propagation [5] algorithm to the problem of tracking the motion of the robot. MCL is an algorithm which persists on the particle filter algorithm [6].

Particle filter: Particle filters are a sample based variant of the Bayes filter [6]. The key idea of the particle filter is to represent the belief by a set of samples. The samples of a distribution are called particles. [7].

2.2. Algorithms for Path Finding

For path finding usually the environment is reduced to a weighted graph composed of nodes and edges. A typical path finding algorithm then solves the problem of finding a link between a given starting node and a given destination node in the graph.

Dijkstra's algorithm: The problem of finding the shortest path to all nodes from a single source node in a network was solved with the Dijkstra's algorithm [8]. There is the precondition that the path costs must be positive. This algorithm is often used in Personal Navigation Assistants (PNA). Even modern Network Routing Protocols like IS-IS or Open Shortest Path First (OSPF) need to find the shortest path within a network. Dijkstra's original algorithm runs in $O(|V|^2)$ where V is the number of vertexes (nodes).

A* search algorithm: A widely used algorithm in path finding and graph traversal is the A* search algorithm. This Algorithm enjoys widespread use because of its performance and accuracy. In the year 1968 Peter Hart, Nils Nilsson and Bertram Raphael of Stanford Research Institute first described the A* search algorithm [9]. The A* search algorithm is an extension of the Dijkstra's algorithm. By using heuristics A* achieves even better performance.

2.3. Cognitive Mapping

The concept of spatial knowledge representation is originated in the area of cognitive sciences. Initially described by Edward Tolman [10]. It is named there as cognitive map. Since that time the concept of cognitive map has become a well-accepted standard [11]. Some Projects are using landmarks as navigational aids on existing street maps [12].

3. The SiMA Project

One of the biggest challenges for modern computer systems is the application in the real world. Current methods of artificial intelligence have still limited success in such scenarios. The SiMA Project follows a bionic approach to improve technical systems by analyzing the human recognition and decision making process. The current implementation of the model is the SiMAi15-Framework [13] for automatic decision making for autonomous agents. It is a technical realized specification of the metapsychological description of the psychic apparatus [14]. The functional components of the human decision making process have been

extracted and specified by functions and interface definitions in a top down analysis and implemented in a decision making process of an autonomous agents. The result is psychoanalytically inspired architecture shown in Fig. 1. It consists of a primary and a secondary process. The primary process is completely unconscious while in the secondary process thoughts are conscious. The current SiMAi15-Framework and the integrated psychoanalytical theories within it are tested in a simulated autonomous agent. The result is a modular framework with an integrated evaluation system which considers internal and external assessment criteria. The implementation of the agent in the simulated environment (Fig. 2) shows the strengths of the system especially when it comes to data interpretation and evaluation using internal and external performance indicators. A more detailed description can be found in [15].

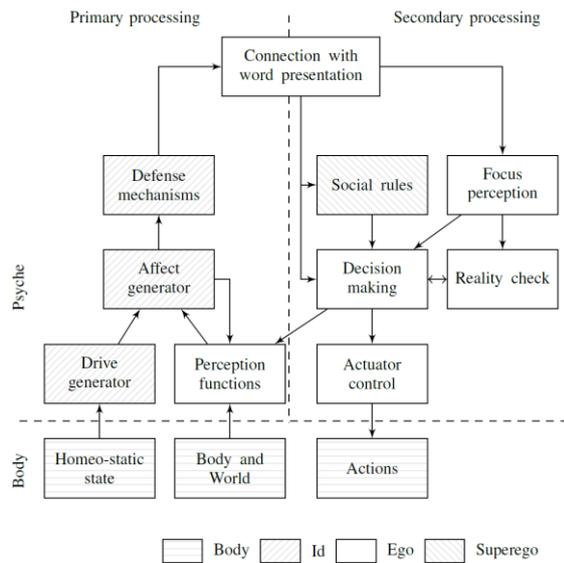


Fig. 1. Psychoanalytically inspired architecture. Adapted from [16].

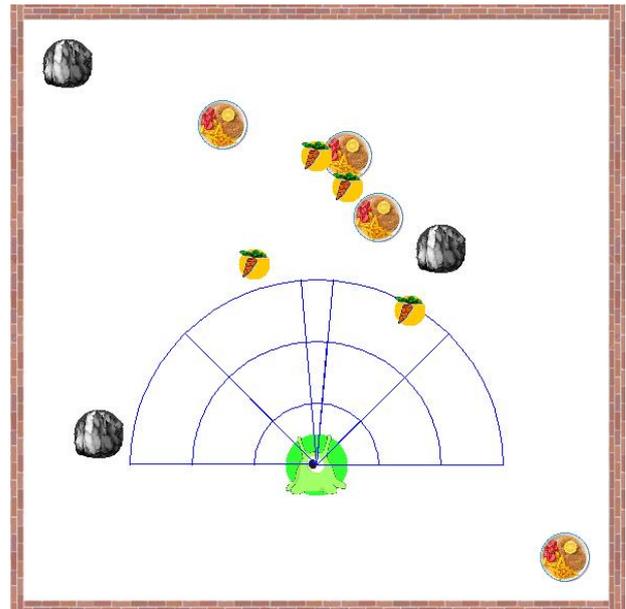


Fig. 2. SiMAi15-Framework simulated environment.

4. A Human-Inspired Localization and Navigation Model

The navigation module is a part of the secondary process of the SiMA, closely associated with the decision module, that is activated whenever there is a destination to be reached, providing the decision module with navigation information (see Fig. 3).

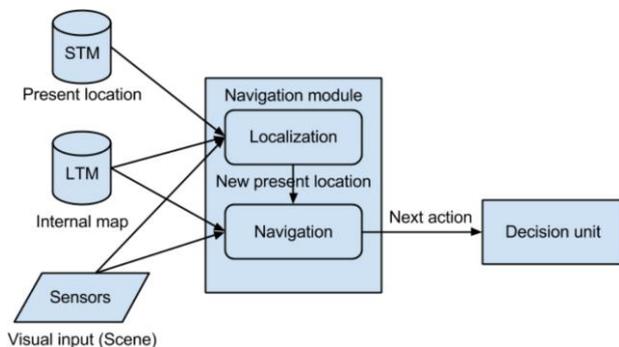


Fig. 3. A flow diagram of the navigation module.

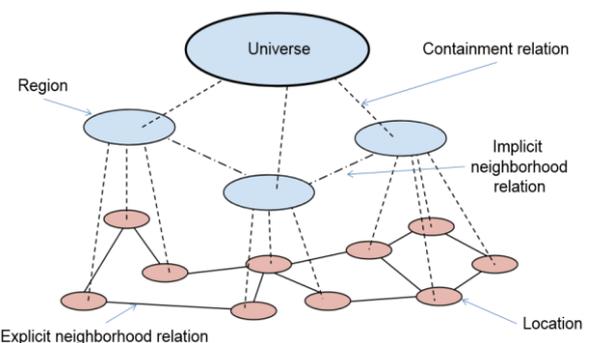


Fig. 4. A graphical representation of the internal map.

In the Psychoanalytically inspired architecture (Fig. 1) the localization module is located between the module "Connection with word presentation" and the module "Focus perception". The navigation module is

part of the module “Decision making”.

The module relies on sets of visual cues passed from visual sensor of the agent, so-called landmarks, that facilitate localization and navigation of the agent in the same way as Hansel and Gretel [17] have used their breadcrumbs.

The module is responsible for long-distance navigation only, i.e. reaching destinations that most often are outside of the agent’s field of view. Any other type of conscious agent movement, such as obstacle avoidance, is delegated to other modules.

4.1. Internal Map

Internal map is the internal representation of the environment and therefore the most important data structure of the model (see Fig. 4). It is designed as a discrete topological representation of the environment with multiple layers of abstraction that facilitates intuitive agent navigation by simplifying long-range routes into a shorter sequence of steps.

The concept of spatial knowledge representation is originated in the area of cognitive sciences. Details can be found in the section “Cognitive mapping”. The bottommost layer of the map consists of all locations in the environment known to the agent connected together in a simple directed graph, in which vertices represent locations and edges represent the actions one needs to take in order to reach one location from the other. The term Location used by us is similar to the Place used by Stephan Winter and Christian Freksa [18].

Location is the smallest perceived division of environment and therefore the most specific answer to the question “Where am I?” Locations in landmark-based localization are distinguished from each another using scenes. A scene consists of a set of landmarks visible from that location and a set of relations between landmarks themselves and between landmarks and the agent. Locations can be grouped into regions based on certain more or less specific criteria such as vicinity and/or similarity. These regions can then further be grouped into superregions, etc., until all locations and regions are grouped into a single region containing and therefore representing the entire environment. Locations and regions are together referred to as map nodes.

Regions can also have neighborhood relations; they are not explicitly defined, but instead inferred based on the neighborhood relations of the underlying locations. The model permits them being precalculated and stored until the map configuration changes. A region can contain both kinds of map nodes simultaneously. A necessary condition that every region must satisfy is that every child node of a region must be reachable from any other child node of the same region.

4.2. Localization

Localization is a process by which the agent attempts to determine its present location within the environment based on several kinds of input: symbolic sensory input containing only localization-relevant information, internal mental map of the environment stored in the long-term memory, and the last known location stored in the short-term memory. Information about the last known location is not mandatory, but allows for significant speedup of the process by restricting the candidate location set to the previous known location and its neighbors. There are many possible relations that can be taken into account when defining a scene, all of them are defined with respect to the agent’s current position. These are, for example:

- Qualitative distance of a landmark from the agent (e.g. “far”, “near”, “close”, etc.).
- Relative position of two landmark on an axis seen from the agent’s viewpoint, i.e. “A is left of B”, or “A is behind B”. The second allows the localization algorithm to account for the parallax effect [19].
- With a fixed Field of View (FoV) angle one can divide the landmark set into those within the FoV when facing a fixed point in the environment (e.g. “when directly facing landmark A, landmarks B, C, and F are also visible”).

The complexity of the data structure and the complexity of the localization algorithm grow as the number of considered relations increase. The goal is to find a set of said relations such that it allows for acceptable detection while keeping the complexity of the scene as well as the matching algorithm to a minimum. The localization algorithm is basically a pattern-matching algorithm that attempts to match currently visible scene to exactly one of the scenes stored within the internal map. The set of all location scenes that will be accepted here is called as the candidate set. This set can potentially include all locations contained in the map; the algorithm must therefore perform filtering operations to reduce the number of the scenes to a minimum. The first step is to retain all those locations whose scenes contain at least one of the currently visible landmarks. This can be a very effective strategy, especially if all landmarks are distinct and globally unique. If the present location has been determined, the candidate set can be reduced by intersecting it with the set of neighbor locations. This step assumes an agent can only either remain in the present location or move to one of the neighboring location at any given moment. We do not expect our agent to be able to beam himself to remote locations.

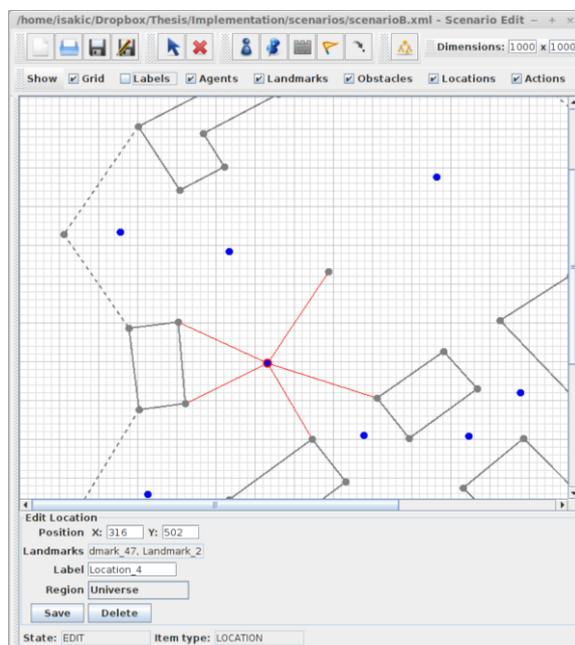


Fig. 5. Scenario editor.

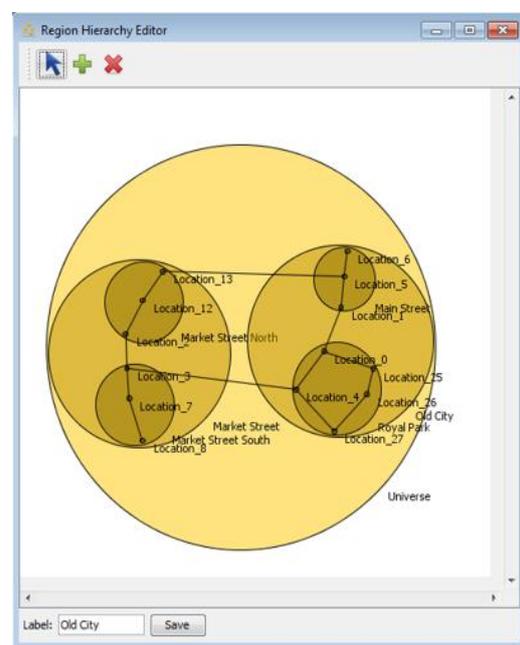


Fig. 6. Hierarchy editor.

4.3. Navigation

Navigation is a process by which the agent attempts to discover a path between its present location and a given destination based on several kinds of input:

- symbolic sensory input containing only localization relevant information
- internal mental map of the environment stored in the long-term memory
- present location provided by the process of localization
- the destination location provided by the decision module of the general model

Presented in Fig. 7 is a navigation algorithm that relies on the hierarchical nature of the internal map to efficiently navigate through the set of known Regions. Note that the path finding algorithm can be any path finding algorithm and need not give the shortest possible path. The advantages of such an algorithm are:

- It limits the part of the map considered to the region that contains both the present location and the given destination.
- The number of items present in the working memory at any time is limited to the maximum number of

children of any region.

4.4. Sensors, Cameras and Image Recognition

The area of sensors, cameras and image recognition is not covered by this work. In the SiMA-model we assume that objects are already recognized distinct before entering the localization part of the model. We also assume that in the short term memory of the agent a virtual 360 degree vision of the agent is synthesized.

5. Simulation

The current implementation of the model is still in testing and calibration at the moment, especially the localization process. It has so far been tested on several instances of landmark-rich environments and given satisfiable results even with the simplest scene-matching algorithms. One example of such an environment is shown in Fig. 5. The according hierarchy is shown in Fig. 6. Further testing will involve testing of its robustness on several landmark-scarce instances to determine its weaknesses and suggest further improvements. For example the self-localization is sometimes inaccurate. To test drive the self-localization the agent has been manually placed in every position of a given map and was asked: "Hey, in which position do you think you are?". Results are visualized by color coding the map in Fig. 8. There the self-localization delivers non optimal results for the boundary between locations 1/5 and 7/8, but that is ok. To err is human. Everyone might remember a situation where he was sure to be in known, remembered location and after moving along a little bit, he detected to be wrong. For implementation purposes a simplified version of the above model has been selected. Landmark relations within a scene have been limited to position "left-of" and "right-of" relations.

```

Input: present location  $L_{loc}$ , destination  $N_{dest}$ 
Determine common region  $R_{common}$  that contains
both  $L_{loc}$  and  $N_{dest}$ 
repeat
  Determine two children nodes of  $R_{common}$ ,  $N_{loc'}$ ,
 $N_{dest'}$ , containing (or being)  $L_{loc}$  and  $N_{dest}$ , respec-
tively
  Find a path between  $N_{loc'}$  and  $N_{dest'}$  and determine
the next node along the path,  $N_{next}$ 
  Determine one child node of  $N_{loc'}$ ,  $N_{dest''}$  that
neighbors any child node of  $N_{next}$ 
  Determine one child node of  $N_{loc'}$ ,  $N_{loc''}$  that con-
tains or is  $L_{loc}$ 
  Set  $R_{common} = N_{loc'}$ ,  $N_{loc'} = N_{loc''}$ ,  $N_{dest'} =$ 
 $N_{dest''}$ 
until  $N_{next}$  is a location
return Action A that leads from  $L_{loc}$  to  $N_{next}$ 

```

Fig. 7. Pseudocode of the navigation algorithm.

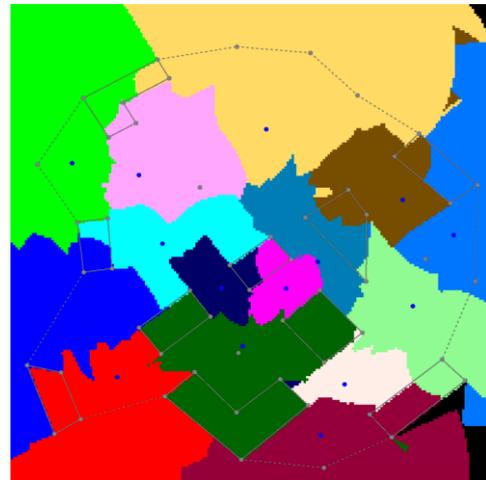


Fig. 8. Location Detection (calculated on per pixel base).

6. Conclusion an Outlook

We have shown a localization and navigation model to enhance the SiMA agent capabilities in showing human behaviour. The agent will search for desired objects in a better way but will also make failures or act inefficient because getting confused by moving objects or missing links between locations. As the agent must change its internal map while operating, for future work we have to extend the model with learning and oblivion capabilities for the internal map. The human brain, as it processes everything in parallel is far more powerful than our search algorithms. There is currently plenty of space for optimizing the used methods in terms of efficiency and speed.

References

- [1] Huang, Y. Y., Cao, Z. L., Oh, S. J., Kattan, E. U., & Hall, E. L. (1987). *Automatic Operation for a Robot Lawn Mower*, 0727, 344–354.
- [2] Dietrich, D., Bruckner, D., Zucker, G., Mller, B., & Tmej, A. (2009). Psychoanalytical model for automation and robotics. *Proceedings of the 9th IEEE AFRICON* (pp. 1–8).
- [3] Hargrave, P. J. (1989). A tutorial introduction to kalman filtering. *Proceedings of IEEE Colloquium on Kalman Filters: Introduction, Applications and Future Developments: Vol 1* (pp. 1–6).
- [4] Dellaert, F., Fox, D., Burgard, W., & Thrun, S. (1999). Monte carlo localization for mobile robots. *Proceedings of IEEE International Conference on Robotics and Automation: Vol. 2* (pp. 1322–1328).
- [5] Isard, M., & Blake, A. (1998). Condensation conditional density propagation for visual tracking. *International Journal of Computer Vision*, 29(1), 5–28.
- [6] Hiemstra, P., & Nederveen, A. (2007). *Monte Carlo Localization*.
- [7] Van Der Merwe, R., Doucet, A., De Freitas, N., & Wan, E. (2001). The unscented particle filter. *Advances in Neural Information Processing Systems*, 584–590.
- [8] Chen, J.-C. (2003). Dijkstra's shortest path algorithm. *Journal of Formalized Mathematics*, 15, 2003.
- [9] Hart, P., Nilsson, N., & Raphael, B. (1968). A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2), 100–107.
- [10] Tolman, E. C. (1948). Cognitive maps in rats and men. *Psychological Review*, 55(4), 189.
- [11] Golledge, R. (1999). *Wayfinding Behavior: Cognitive Mapping and Other Spatial Processes*. Johns Hopkins University Press.
- [12] Raubal, M., & Winter, S. (2002). Enriching wayfinding instructions with local landmarks. *Proceedings of the Second International Conference on Geographic Information Science: Vol. 2478* (pp. 243–259).
- [13] Brandstatter, C., Dietrich, D., et al. (2014). Naturwissenschaftliches, psychoanalytisches modell der psyche fuer simulation und emulation. Institute of Computer Technology, Vienna University of Technology, wissenschaftlicher Bericht 2.
- [14] Dietrich, D., Schaat, S., Bruckner, D., Doblhammer, K., & Fodor, G. (2013). The current state of psychoanalytically-inspired AI. *Proceedings of 39th Annual Conference of the IEEE on Industrial Electronics Society* (pp. 6666–6671).
- [15] Lang, R. (2010). A decision unit for autonomous agents based on the theory of psychoanalysis.
- [16] Deutsch, T., Bader, M., Zeilinger, H., Lang, R., Vincze, M., & Muchitsch, C. (2011). Cognitive decision unit applied to autonomous biped robot nao. *Proceedings of 9th IEEE International Conference on Industrial Informatics*.
- [17] Grim, W., & Grim, J. (1812). H'ansel und Gretel. *Kinder-und Hausmärchen*.
- [18] Winter, S., & Freksa, C. (2012). Approaching the notion of place by contrast. *Journal of Spatial Information Science*, 5(5), 31–50.
- [19] Blank, A. A. (1958). Axiomatics of binocular vision the foundations of metric geometry in relation to space perception. *JOSA*, 48(5), 328–333.



Christian Brandstätter was born in 1979 at Tulln, Austria, he studied informatics and informatics-management at the Vienna University of Technology and finished both studies with the degree Master of Science.

He works as a system administrator for general IT in a pharmaceutical research institute. His responsibility there is the IT infrastructure (network, storage and virtualization). Currently he works on his dissertation and is a member of the Institute of

Computer Technology at the Vienna University of Technology.



Samer Schaat was born in 1982 in Vienna, Austria, he studied medical informatics at the Vienna University of Technology and finished with the degree of a Master of Science.

He works as a project leader at the Institute of Computer Technology at the Vienna University of Technology. Currently he is PI for the national funded project Cognitive Multi-Agent System Supporting Marketing Strategies of Environmental-Friendly (CogMAS) and Co-PI for the national funded project Energy-efficient Cognitive Autonomous Building Automation (ECABA).



Alexander Wendt was born in 1981 in Sweden. After high school and a year of study at the University of Linkping, Sweden, in 2002 he began studying technical physics at the Vienna University of Technology. His focus was applied and computer-supported physics. The degree program was completed in 2007.

He worked between 2007 and 2010 as a safety engineer at Frequentis AG. Since 2010 he is working on his dissertation in Cognitive Automation and situational awareness. His research focuses on the areas of artificial intelligence, knowledge representation, and database systems. In cooperation with the company Frequentis AG, he was active in the project GÖPL (Public Private Joint Situation Report), with the task of evaluating data base systems. Since mid-2011, he has worked in the field of Smart Grids in a project in cooperation with Siemens AG, where he is responsible for the development of network communication.



Martin Fittner was born in 1984 at Mistelbach, Austria, he studied embedded systems at the University of Applied Science Technikum Wien in Vienna and finished with the degree master of science.

He works as a senior developer for software and system design for automotive electronic systems at the Robert Bosch AG. He is a member of the SiMA (Simulation of the Mental Apparatus and Applications) team. Currently he works on his dissertation at the Vienna University of Technology in the field of the SiMA project.