Environment Simulation for Scenario Perception Models

Ch. Roesener, H. Hareter, W. Burgstaller, and G. Pratl
Institute of Computer Technology
Gusshausstrasse 27-29
A-1040 Vienna, Austria
{roesener, hareter, burgstaller, pratl}@ict.tuwien.ac.at

Abstract

The evolution of automation systems has reached a point where the increasing complexity requires new mechanisms to keep control of design and operation. New automation systems models have to be designed, which are capable to perceive and react based on global situations. Perception requires scenario recognition, which again is based on the continuous input of many different types of sensors; in future all control systems will have to cope with massive data quantities that are collected from the environment. A stable system that reliably perceives situations relies on the diversity and amount of sensor inputs. The approach discussed in this paper is the simulation of big amounts of sensors and actuators in order to verify this scalability and to evaluate and verify models based on scenario perception and recognition.

1. Introduction and motivation

Future automation system will have to cope with massive amounts of data: The number of sensors within control systems has increased dramatically in the last years. Nowadays office and industrial buildings are equipped with up to 70.000 nodes [1]. Due to developments of nanotechnology this number might in future be integrated even into a single room. Unfortunately present automation models do not seem to be ready to handle such masses of information. In order to process the expected quantities of data the need for new methods is obvious. Today it is not possible to oversee all necessary situations within a production line and its environment, where e.g. mobile robots might block within a production line causing dangerous situations for humans nearby. It is not possible anymore to program and maintain all scenarios in standard data processing. Future systems have to be capable to process lots of data and have to learn adapting appropriate reactions to new situations.

Looking at nature and the evolution so far we find that this situation is pretty old and well known: The human nervous network is one of the most complex communication networks that can be found. The peripheral nervous system feeds the human brain with information in the order of 10^9 to 10^{10} bits/s; the conscious mind however only receives 10 to 10^2 bits/s for processing, all other information is filtered and preprocessed at earlier stages [2]. The biological archetype can give inspiration for new methods of data acquisition and handling based on scenario perception. New models convert large amounts of data on low abstraction levels into complex symbols. Therefore enormous amounts of sensors have to be installed, configured and tested, before the "behavior" of such a system can be thoroughly tested.

The goal of this project is to simulate future buildings and installations, e.g. high rack warehouses, office buildings or factory floors, containing millions of sensors (and actuators), which continuously generate enormous amounts of sensory data. This simulation shall help to build up new data processing models to learn and be prepared for this big amount of data. Such a system cannot deal with single values, but has to react based on the recognition of whole scenarios. The first model tested on this simulation is the *Perceptive Awareness Model* (PAM, see section 3.1), which was designed in a former project, the *Smart Kitchen Project* (SmaKi) [3]. An important issue here is the scalability of sensory input. The simulation shall give an answer about the number of data points up to which a system will be capable to react efficiently. Furthermore the simulation can give an answer about the optimal number of sensory data points covering a maximum of information within a certain room or building.

2. Modeling and Simulation So Far

The Smart Kitchen project was initiated at the Institute of Computer Technology (Vienna University of Technology, Austria) in 2000. The goal of this project was to find solutions for new automation systems by applying the methods of a new model, the Perceptive Awareness Model: Aside of promising results the project also showed the restrictions of present automation models and the need for a new concept [4].

The model developed during this project allows the handling of different industries integrated into one system, where today's methods are on their limits. The first ideas about the new concept have been published in [5], [6], and [7].

In [8] prediction methods have been used for the new model because of the adaptability of the case-based reasoning architecture. The first and major work about prediction has been published by A. Aamodt in [9], concerning e.g. case based reasoning. The basic idea was to use the recognition of situations with the focus on finding a proper reaction to the situation. One of the following various researches, which has been presented by T. Kitamura [10] is dealing with Case-Based Architecture.

One of the major problems identified in the SmaKi project was the rigid assignment of devices to specific applications of different industries. Extending these applications and upgrading devices to ensure availability to other systems meant high effort. Additionally, these specific extensions complicated the maintenance as they caused more and more devices to be in non-obvious relations to other components in other systems.

The environment that is employed here is not based on real-world sensors and actuators in huge amounts, but is rather simulated to evaluate a system that is based on the PAM model. For once, sensor systems containing these amounts of sensor are very difficult and costly to realize; therefore simulation is an acceptable alternative to get suitable inputs to study the behavior of systems based on PAM. Seconds, simulation may not only predict the behavior of a model, but might also provide additional information about how to improve a installation later on [11]. The simulation used here is based divided in models of local interaction for e.g. surface reflections and models of global interaction [12]. The roots of these methods date back to the 1970s; within this project we will enhance these methods not only for light interactions, but we will also simulate other physical properties like temperature, pressure fields, and gas distribution.

3. Project Concept

In order to thoroughly evaluate and optimize a system that uses scenario recognition based on sensor diversity and redundancy we build a simulated environment of sensors and actuators; this allows for full control of the environment that the system has to handle. The perceptive awareness is based on the Perceptive Awareness Model (PAM) that has been introduced in [13]; this project modifies and extends the PAM to further improve the model. Furthermore, we define a concept for the simulation and visualization of huge amount of sensors as well as the interface between the simulation and the Modified PAM.

3.1. Modified PAM

The system (Figure 1) consists of five main layers. The two lower layers contain sensors and actuators (to detect and influence the environment) and a network, which connects these devices. They implement a communication protocol, which is usually described by the seven layers ISO/OSI Reference model. Some fieldbus systems like LonWorks [14] additionally define profiles, which extend the communication protocols above OSI layer 7.

In the "Symbols" layer, which is the lower layer of the Modified PAM the sensor data are combined to symbols. In the "Semantics" layer, scenarios are generated out these symbols. In the "Reaction" layer on the top of the model, appropriate actions are derived from the scenarios. The actions are passed through all layers down to the actuators.

The lower layers are simulated due to the lack of availability of the necessary sensors for the bionic approach. However, in order to be able to replace the simulation by a real control network in future an interface between the network and the symbol layer is defined; this interface has to be designed carefully in order to guarantee seamless migration from simulation to real-world implementation.

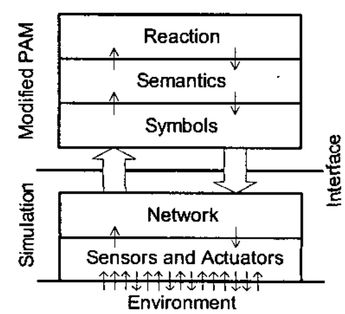

Figure 1: System Design with Modified PAM

3.2. Environment Simulation

The simulations of factory buildings have to deal with different types of sensors like optical sensors (e.g. cameras or photo sensors), thermal sensors, and pressure transducers. The simulation aims at getting data from the environment; the sensory data is in the first step created by idealized sensors, not taking the properties of each sensors into account.

On the first step a three dimensional static model of a room or an assembly hangar is built. The room can contain equipment like cupboards, tables, chairs and a stove. The assembly hall can contain a conveyer belt, production lines and so on. There are several sources of light and heating elements in both environments. Sensors – or better fields of sensors – are integrated in the walls and the equipment. For visualizing the model techniques from computer graphics and CAD are used [15].

The model of the environment is object oriented and distinguishes between different kinds of objects on different abstraction levels. The global object in the simulation is in the first step a single room. It has global properties like time, global temperature and the static position of other objects representing e.g. equipment located in the room. The equipment represents another class of objects. These objects have no sensors integrated, but contain special properties like temperature, weight or speed, which influence other objects. Movements of these objects are measured and lead to an update of their position in the global room object. Other objects like walls, the cupboard or the conveyer belt are fixed in the room and have sensors integrated.

To calculate the range of influence for each object on its environment, techniques of ray tracing and rendering (like z-buffer and scanline algorithms) will be used [16]: A stove, for example, emits heat and leads to a significant increase of the temperature of all objects nearby. This influence will be calculated after initialization of the simulation system. Beside the current situation long-term sequences and their continuous influence have to be calculated. When a heating element is heating for a long time a global increase of temperature in the whole room has to be simulated. Each object in the room will have a significantly higher temperature during this sequence. As the situation changes only slowly the global situation appears "constant".

The properties of the room and its objects can be changed in two ways. In the first step of the project changes of environment can be forced only within an initial process where the properties of each object have to be set. This means that the physical location of all objects and special properties (like the temperature of heating objects) are defined in the beginning. In a next step the user of the simulator is able to control the parameters via a graphical user interface, that consists of a 3D-view of the room and all objects like it is used in common CAD applications or by a script file, which contains all necessary changes. Parameters and properties of objects can be changed manually and therefore have to be validated by the simulator. The model itself affects the environment by the simulated actuators.

The next phase in the project deals with the dynamic processes like movement of objects (and humans) in the room or the change of properties of objects by actuators; this initiates a dynamic recalculation of the range of influence and the characteristics of the objects. These calculations are initiated by the room-object, which contains the information about the accurate movement and changes of properties, which might be caused by actuators or by an object itself. To reduce calculation to a minimum the simulation has to differ between fast and slow changes. If, for example, a person walks through the room, the range of view for the optical sensors has to change fast enough to recognize the movement. On the other hand the slow change of temperature of a heating element demands only slow recalculation.

Ray-tracing-like algorithms are used to generate the optical image of the room [17], [18] for the graphical user interface. In this case it is not necessary to produce a photorealistic image. Therefore calculation will be reduced in color and detail. Additionally, the time for calculation depends on how fine-grained room and objects have to be simulated, which depends on the type and surface of an object: Walls that are not in the scope of interest can be simulated rather roughly, while small objects or objects that are currently in the focus of the system require higher resolution. This also depends on the application of the system and changes with the focus of attention of the system. The optical simulation may not be able to generate images to identify a certain person, but it should be able to detect whether or not anybody is in the room. Additionally, the system should be able to zoom into objects of interest.

Another goal for simulation is the dynamic change of sensors and actuators. The maximum, optimal and minimum number of data points that the scenario recognition requires to reproduce an efficient image of the environment and to derive the symbols shall be found. Based on the results of simulation the minimum number of sensors with which the system can still operate is significant for optimizing the PAS.

3.3. Simulation Interface to PAS

The interface between sensor data and the system analyzing it has to meet several requirements: It has to deal with considerable amounts of data, it has to provide this data in a time resolution that is small enough for the system to react appropriately, and it has to provide not only current data, but also historical data to allow for scenario recognition.

Ideally, the PAS would require random access to all field data at any point in time; additionally, data should be congruent in time (i.e. all sampled at the exact same point in time). Although this can be done when simulating data, it is in general not possible in a real environment. Data will always be distributed over time, depending on when it has been sampled (although it has to be a goal to keep this "jitter" small). This fact is taken into account in the interface.

Aside of this jitter there is also the delay from the time when a value has been sampled to the time when it is actually available at the PAS. This propagation delay must be short, so that the system can react in time.

For reasons of modularity the data exchange is made network transparent, meaning that all communication is done by a protocol that can be transported over the network. This allows distributing the different tasks to separate machines, each with separate resources; additionally, it provides a clean cut between different tasks, because the network protocol is the only way the modules can communicate with each other. A drawback, however, is the increased propagation delay, which is introduced by the additional effort for transporting data over the network. The decision about the protocol has not yet been made; it can either be a proprietary protocol or a standardized solution.

Since the PAS requires fast access to data, the system can not start requesting data from the field level or the simulation when the Modified PAM requests the data; Data has to be buffered to be available immediately. And because we need not only current data, but also historic data, a data base is employed to provide this buffer (or process image). The PAS therefore communicates only with the database, requesting data at will, while the data in the data base is asynchronously updated from the simulation (or field level) side. Again, the drawback here is an increased propagation delay. When looking at a final implementation, it may be necessary to eliminate both database and network communication and optimize the system towards a monolithic, stand-alone solution. But since we are still in the research phase, such an optimization is currently not feasible.

4. Conclusion and Outlook

We have presented the concept of a simulation approach for a scenario recognition model. Since the diversity and redundancy of sensory input is a key issue for a system to "perceive" scenarios instead of measuring values, we focus on providing the system with an environment that simulates this sensory input with full control of all the parameters that are necessary to optimize the Modified PAM. In order to be able to migrate from the simulation back to a real-world implementation the design of the interface between the perceptive awareness system and the simulated environment is another vital concern, which is to be thoroughly investigated. Based on the work that has already been done at the Institute of Computer Technology the simulation is another piece in the puzzle of the ongoing research.

References

[1] W. Kastner, P. Palensky, T. Rausch, Ch. Roesener, "A Closer Look on Today's Home and Building Networks" *IEEE Africon*, Botswana, to be published, 2004.

[2] K. Kunsch, *Der Mensch in Zahlen, Datensammlung in Tabellen*, Spektrum Akademischer Verlag Heidelberg, 2000

[3] S. Soucek, G. Russ, C. Tamarit, "The Smart Kitchen Project – An Application of Fieldbus Technology to Domotics", *Third International Workshop on Networked Appliances*, 2001.

[4] D. Dietrich, W. Kastner, T. Maly, Ch. Roesener, G. Russ and H. Schweinzer, "Situation Modeling", *5th IEEE International Workshop on Factory Communication System*, Austria, to be published, 2004.

[5] C. Tamarit, D. Dietrich, K. Dimond, G. Russ, "A definition and a model of a Perceptive Awareness System (PA)", *FeT'2001 Proceedings, 4th IFAC International Conference on Fieldbus Systems and their Applications*, Nancy, France, 2001.

[6] G. Russ, D. Dietrich, C. Tamarit, "Situation Dependent Behavior in Building Automation", *Workshop Proceedings of EurAsia-ICT 2002 Advances in Information and Communication Technology*, Shiraz, Iran, 2002.

[7] C. Tamarit, G. Russ,"Unification of Perception Sources for Perceptive Awareness Automatic Systems", *Africon'02 Proceedings*, 2002.

[8] G. Russ, *Situation Dependent Behavior in Building Automation*, PhD thesis, Institute of Computer Technology, TU-Vienna, 2003.

[9] A. Aamodt, E. Plaza, "Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches", AI Communications. IOS Press, Vol. 7: 1, 1994.

[10] T. Kitamura, T. Tahara, K. I. Asami, "How can a robot have consciousness?", *Advanced Robotics*, Vol. 14, No. 4, 2000.

[11] P. Bratley, *A guide to simulation*, Springer, 1983.

[12] A. Watt, *3D Computer Graphics*, Addison-Wesley, 2000.

[13] C. Tamarit, *Automation System Perception*. PhD thesis, Institute of Computer Technology, TU-Vienna, 2003.

[14] D. Loy, D. Dietrich, H. J. Schweinzer, *Open Control Networks-LonWorks/EIA 709 Technology*, Kluwer Academic Publishers, 2001.

[15] P. Brunet, *CAD tools and algorithms for product design*, Springer, 2000.

[16] T. Haenselmann, *Raytracing*, Addison-Wesley, 1996.

[17] A. Watt, M. Watt, *Advanced animation and rendering techniques: theory and practice*, Addison-Wesley, 1992.

[18] T. Möller, E. Haines, *Real-time rendering*, Peters, 2000.