AUTONOMOUS BEHAVIOR-BASED EXPLORATION OF OFFICE ENVIRONMENTS

Daniel Schmidt, Tobias Luksch, Jens Wettach and Karsten Berns
University of Kaiserslautern (TU)
Gottlieb-Daimler-Straße, 67663 Kaiserslautern, Germany

Keywords: Mobile robots, autonomous exploration, behaviour-based, indoor environments.

Abstract: Besides safe motion control the gain of environmental knowledge is a key for a reliable home or office service robot. When being set into a completely unknown environment the robot has to be able to derive a certain abstract internal representation of this world without any user interaction. This knowledge enables the robot to know how to get from its actual place in one room to a target position in another room as a prerequisite for transportation tasks for example. In this context, the combination of a behavior-based motion control system and an abstract topological map based on geometric representations of rooms seems promising. As the concept of motion and exploration behaviors facilitates to compete with noisy sensor information and geometrically imprecise maps, it has been used to develop completely autonomous exploration strategies for deriving topological representations of common indoor environments. The only prescribed world knowledge is the fact that these environments are composed of rectangular entities (rooms) which are connected by openings (doors). The developed system has successfully been tested in simulation and reality.

1 INTRODUCTION

Mobile service robots have to derive an internal representation of their working environment autonomously. When such a machine is installed in its new habitat, it has to perform a certain initialization phase for setting up its internal map before the user can give any orders to it. Of course this map building has to be executed without any user intervention.

One question is how abstract and accurate this representation has to be to facilitate safe robot motion and reliable task performance. In this paper a new behavior-based control approach is presented that enables a mobile robot to derive a topological map of indoor environments completely autonomously. The realized approach allows to compete with inaccurate sensor distance information and imprecise mapping results. The main challenge is the coordination of the behavioral network in order to guide the robot autonomously through priori unknown environments only based on 2D distance information from laser scanners and the knowledge of rooms being rectangular.

To explain the proposed exploration system the paper is organized as follows: section 2 summarizes previous work, section 3 introduces the sensor systems of the applied mobile robot. Section 4 describes the basics of the environmental representation approach and section 5 introduces the applied behavior-based control system. The exploration system is explained in section 6, followed by test results in section 7. The project achievements are summarized in section 8 and supplemented by intended future enhancements.

2 RELATED WORK

Autonomous exploration mainly depends on sophisticated exploration strategies. Common approaches include heuristic methods which calculate the probability of new sensor informations at given positions out of the actual or already retrieved sensor data.

One of such exploration methods is proposed by (Surmann et al., 2003). It utilizes so-called *next-best-view*-algorithms to calculate a position which promises a maximum of new informations about the environment. A new measurement is taken only at this position. Another strategy is pointed out by (Sim and Roy, 2005), who implemented an exploration method that tries to minimize the error of the global

Schmidt D., Luksch T., Wettach J. and Berns K. (2006).
AUTONOMOUS BEHAVIOR-BASED EXPLORATION OF OFFICE ENVIRONMENTS.
In *Proceedings of the Third International Conference on Informatics in Control, Automation and Robotics*, pages 235-240
DOI: 10.5220/0001204002350240

map by integrating the informations collected while driving a trajectory.

Other strategies have been published before. (Yamauchi et al., 1998) proposed a so-called frontier-based exploration. Here evidence grids are used for the detection of unexplored regions and for the extraction of new exploration targets. A different method attempts to close virtual chains around two-dimensional features as (Wullschleger et al., 1999) published. There an object or room is explored if all sides have been detected by the robot.

These strategies do not match the mapping system in this work (explained in section 4) so a new behavior-based method has to be created.

3 MOBILE ROBOT MARVIN

The test bed for the proposed exploration strategy is the autonomous indoor vehicle `Marvin` - its sensor systems are shown in figure 1. Above the bumper rail for safety aspects two `SICK` laser scanners are mounted, one at the front (model `S3000`) and one at the back end of the robot (model `LMS200`).

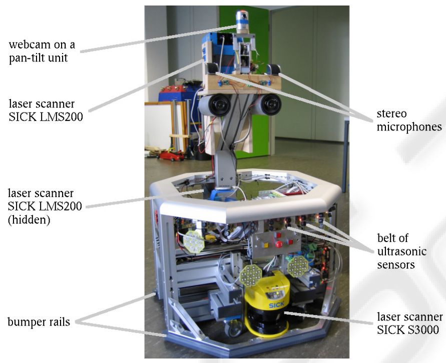

Figure 1: Front view and sensor system of mobile robot `Marvin`.

Above the laser scanners, there is a belt of ultrasonic sensors, mounted below the hexagonal top frame of the robot. The goal of this sensor belt is to detect obstacles that do not intersect the measurement plane of the laser scanners, as for example table tops, which the robot cannot underpass due to its raised arm (camera holder).

On top of the robot's arm there is another `LMS200` scanner for getting more reasonable measurement results in highly crowded areas. Soon it will be mounted on a tilt unit for 3D measurements. Besides the laser scanner, there are two microphones to detect the direction of sources of noise. Finally, there is a standard webcam mounted on a pan-tilt unit. It is used for the detection of human faces, blobs of color and straight edges in a scene.

4 ENVIRONMENTAL REPRESENTATION

In this section the recognition and representation of the environment based on 2D distance information from the laser scanners is introduced as this is helpful for understanding the exploration strategy. From the sensor data an abstract rectangular room consisting of four walls is generated and openings like doors are registered as described in (Kleinlützum et al., 2005). The dimensions of the created room comply with the measured distance information. Figure 2 shows an example of laser scanner data (a) and the generated room (b) with an already passed door on the lower side including the room local coordinate plane and three additional openings. This example will be continued in chapter 6.1.

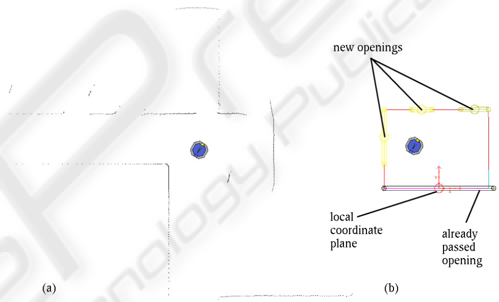

Figure 2: Robot `Marvin` surrounded by scanner data (a) and the corresponding map representation (b) (top view).

Regarding this form of description some assumptions need to be mentioned:

- All walls inside the rooms are arranged right-angled.
- Rooms can be described by using rectangles or combinations of them.
- Walls are fixed inside the environment and cannot be moved.

Correspondingly the application is limited to rectangular environments as they can be found inside office buildings. Environments with other shapes will be approximated by rectangles which may lead to inaccurate maps of the rooms. These restrictions have been made to keep the representation as abstract as possible and to make it more robust against dynamic changes or other interferences. A topology matrix specifies the connections between the respective room maps for navigation tasks.

At the moment a room is defined as well-known if the robot has explored all walls by travelling closer than two meters to each part of the walls. This aspect will change in future when new objects are added to the map and is reflected in the rating $k(R_j) \in [0, 1]$.

5 BEHAVIOR-BASED ROBOT CONTROL

The basic ideas of the behavior-based robot control system is subject of this section. Since R. Brooks invented his *Subsumption Architecture* (Brooks, 1986) the behavior-based robot control is established as one of the most powerful systems for controlling a complex robot.

For the implementation of the exploration system the behavior architecture introduced in e.g. (Albiez et al., 2003) is used. Figure 3 shows the structure of a single behavioral module. The input vector $\vec{e}$ represents sensor values or preprocessed information from other modules. This input is transformed by the transfer function $F(\vec{e}, \iota, i)$ into the output $\vec{u}$ which represents motion control values or information for other behaviors. The calculation is influenced by ι which determines the activation of the module and by the inhibiting input i which counteracts the activation. The outputs a and r represent the module's activity respectively the target rating (or satisfaction).

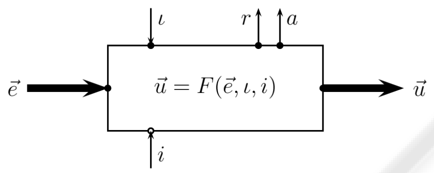

Figure 3: Structure of a behavior module.

When several behaviors take effect on the same control values, their output can be fused using two different strategies, realized by units denoted as fusion nodes. Both nodes collect the controller outputs $\vec{u}_i$ and activities a_i of the involved modules and combine them to one control vector $\vec{u}$. The maximum fusion forwards the output vector of the behavior with the highest activity to the fusion output. In contrast the weighted fusion averages all controller values regarding the different activities.

Further information on the architecture can be found in (Proetzsch et al., 2005).

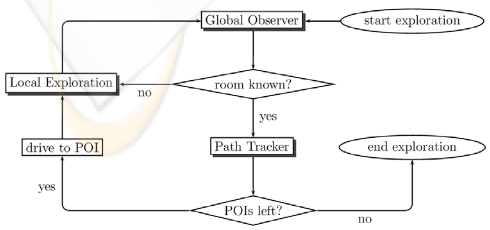

Figure 4: Process of the global exploration.

6 GLOBAL AND LOCAL EXPLORATION

This section describes the behavior based exploratory strategy that enables the mobile robot `Marvin` to gain knowledge about complex indoor environments without any user intervention. The control system has been integrated into the $AutEx^{IE}$ system (**Aut**onomous **Ex**ploration of **I**ndoor **E**nvironments) as shown in figure 5. This structure contains perceptual components for mapping and localization as well as a behavior-based steering system. It has been introduced in (Kleinlützum et al., 2005) and updated to its current state by (Schmidt, 2006).

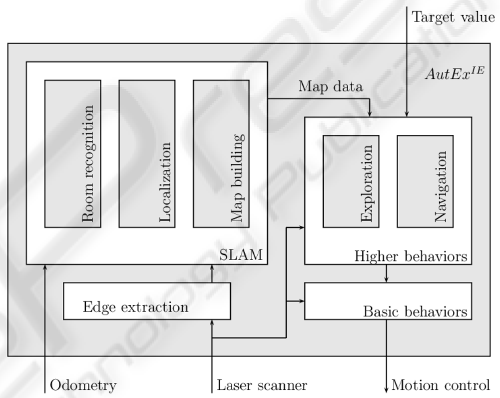

Figure 5: Exploration structure of the robot `Marvin`.

Targeting a hierarchical structure that is easy to upgrade a control system has been implemented with different grades of abstraction. The lowest level is composed of basic behaviors including safety aspects and simple movements. Exploration behaviors and additional components are implemented on higher levels. The main exploration system consists of three concurrent behaviors:

Local Exploration Collection of simple behaviors which allow the exploration of the current room.

Path Tracker Drives the robot to the closest point of interest by using a topological graph.

Global Observer Surveys the *Local Exploration* and triggers the path tracking behavior, if for a period of time no new informations have been retrieved.

Figure 4 shows a flowchart of the exploration activity based on the interaction of these three components. The idea is to separate the *Local Exploration* which depends only on laser scanner data and the global exploration strategy that has access to the more complex map data. By this means the robot can travel directly based on sensor informations with a high reactivity.

The *Local Exploration* is mostly responsible for collecting data about the environment. It is surveyed by the *Global Observer* behavior which can guide the robot by the *Path Tracker* module to an interesting point.

6.1 Local Exploration

The first part that has been implemented is the *Local Exploration.* It has been realized as a set of different simple behaviors that compete for the control of the robot `Marvin`. From the outside view this group reacts like one behavior with the interfaces introduced in section 5. All included behaviors only produce turning commands, the forward driving is done automatically but can be inhibited. The structure of the *Local Exploration* can be seen in figure 6 - the members are described in the following:

Random Cruise Simple behavior that drives the robot in a random direction. The activity is always $a_{RC} = 1$ unless it is inhibited by any other **active** local behavior - the target rating is set $r_{RC} = 0$.

Hold Distance Left / Right Two behaviors which try to keep a certain distance from obstacles on the left and the right side equivalent to wall follow behaviors. The activity depends on the minimal distances d to the nearest obstacles and two thresholds $d_l, d_u \in \mathbb{R}$ as seen in equation 1. The target rating is computed by the needed movement of the robot $r_{HD} = |\vec{m}|$.

$$a_{HD} = min\left(max\left(1 - \frac{d - d_l}{d_u - d_l}, 0\right), 1\right) \quad (1)$$

Pre Evasion Turns the robot away from frontal obstacles or towards passages in front of it. Activity a_{PE} and target rating r_{PE} are calculated as for the distance holding behaviors.

Local Door Driving Allows the robot to leave the current room by turning towards an opening inside a wall even if this opening is not in front of the robot. The activity is set depending on the needed movement $\vec{m}$ (equation 2), the target rating is set $r_{LDD} = 0$.

$$a_{LDD} = \begin{cases} 1 & |\vec{m}| > 0 \\ 0 & else \end{cases} \quad (2)$$

Narrow Driving This behavior is meant for a special situation that only occurs that way on the real robot. Inside narrow doorways several behaviors can be active and counteract each other. Therefore this behavior suppresses all other local exploration modules so that the robot cannot turn and goes straight ahead driven by the basic behaviours. The activity a_{ND} is calculated in a similar way to the distance holding behaviors, whereas the target rating is set $r_{ND} = 0$.

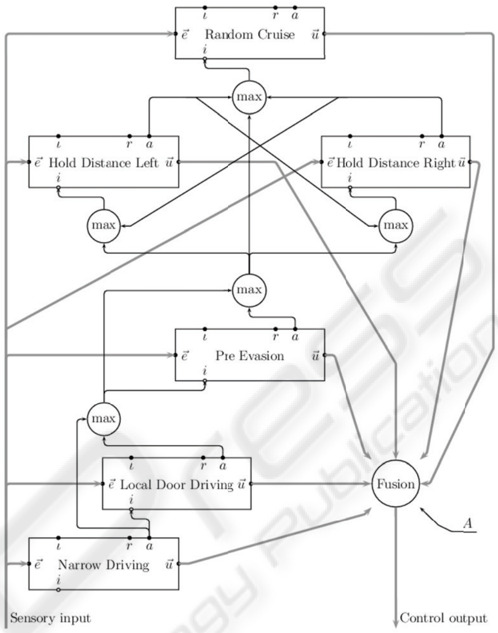

Figure 6: Interaction of the local exploration behaviors.

All these simple behaviors enable the robot to explore a local area by following walls and travelling through passages and narrow doors. The activity of this group is determined by the maximum of the internal activities $a_{LocExp} = max(a_{RC}, a_{HDL}, ..., a_{ND})$. The target rating $r_{LocExp} = 1 - k(R_j)^2$ is calculated with the knowledge rating $k(R_j) \in [0, 1]$ for the current room R_j.

As sensor input the local exploration behaviors use an abstraction from the raw laser scanner data. Therefore the space around the robot is split into several sectors that reflect spatially discretized distance information (denoted as *Robot Sectors* in the following) as shown in figure 7. These illustrations correspond to the scanner data and resulting map in figure 2. For each sector several values are determined:

1. minimal distance of the laser scanner beams inside the sector
2. maximal and minimal distances to the generated walls of the current room
3. minimal distance to already passed doors

Depending on different states of the exploration the local behaviors receive the minimum of 1. and 2. (figure 7 (a)) to keep the robot inside the room, the minimum of 1. and 3. (figure 7 (b)) so that the robot can leave the room only through an unknown opening or only the scanner data as shown in figure 7 (c).

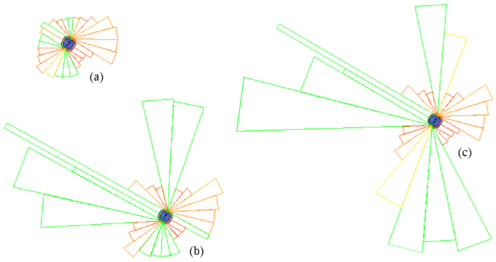

Figure 7: *Robot Sectors* at different states of the exploration inside the same room (corresponding to figure 2).

6.2 Global Exploration

In order to explore complex indoor environments autonomously additional behaviors are needed that guide the robot from room to room. This task is coordinated by the *Global Observer* module based on the *Path Tracker* (see figure 4 for an overview).

The global exploration strategy works as follows: The robot starts at an arbitrary position inside its environment by generating the first room. Then the *Local Exploration* becomes active and the robot starts driving through the surrounding area by using the laser scanner data until the room is well known. Then the robot is allowed to leave the room by a local behavior that drives it through a narrow opening.

During the local exploration the *Global Observer* surveys the actions and becomes discontented, if no new information is added to the map within a certain period of time. So the activity is set $a_{GO} = 1$ if the local exploration is active (equation 3).

$$a_{GO} = \begin{cases} 1 & a_{LocExp} > 0 \\ 0 & else \end{cases} \tag{3}$$

The target rating r_{GO} can be calculated by using equation 4 with a counter C that increases if the map data has not changed since the last check and is reset otherwise. $T_1, T_2 \in \mathbb{N}$ define how many cycles are used for becoming unsatisfied with $T_1 > T_2$. As a result the local exploration can be longer active inside poorly known rooms R_j before the rating becomes completely unsatisfied ($r_{GO} = 1$).

$$r_{GO} = min\left(1, \frac{C}{T_1 - (T_2 \cdot k(R_j))}\right) \tag{4}$$

At a maximum of dissatisfaction the *Global Observer* triggers the *Path Tracker*. This behavior creates points of interest (short: POIs) from the map data by searching for openings which haven't been passed through by the robot or for unknown parts of walls. Furthermore it constructs a graph that allows the robot to travel to each point inside every known room. The closest POI is selected by using the Dijkstra algorithm. When the robot moves along this path the *Path Tracker* stays active, until the target position is reached. To achieve this a variable t_r determines if the robot reached its target or not (equation 5).

$$a_{PT} = 1 - t_r \tag{5}$$

For surveying the robot's path the *Path Tracker* becomes discontented if it cannot reach a point on the path because of objects onto the point or a closed doorway. Therefore a counter C_c increases if the robot is closer than one meter to the next point and another counter C_i increases if it is moving away from the next point. So the target rating (equation 6) is calculated using these counters and two thresholds T_c and T_i.

$$r_{PT} = max\left(\frac{C_c}{T_c}, \frac{C_i}{T_i}\right) \tag{6}$$

In the first case the robot accepts the nearly reached position and turns to the following way point. If the other case occurs the *Path Tracker* has to change or delete the actual edge inside the graph and recalculate a new path from the actual position to the target point.

When reaching the destination the local behaviors become active again and continue the exploration.

7 TEST RESULTS

The described exploration system has been tested in simulation as well as on the robot `Marvin`. For the simulation the scanner data is generated by using a manually created map and adding noise to the values.

Figure 8 describes the values of the global behaviors during the exploration of a simulated office environment. The first three rows are the activation ι_{Loc}, activity a_{Loc} and target rating r_{Loc} of the *Local Exploration*. Below the values of the *Global Observer* and of the *Path Tracker* module are shown.

It is obvious, that most of the time the *Local Exploration* is active and controls the robot. During this simulation run the target rating of the *Local Exploration* r_{Loc} is decreasing while the robot retrieves more informations about the room. At about second 60 the *Global Observer* becomes discontented because no new information is added to the map. As the target rating r_{GO} reaches its peak the *Local Exploration* is turned off and the *Path Tracker* starts to steer the robot to the closest point of interest. Near to each waypoint the target rating r_{PT} increases as the described counter C_c is incremented. When the robot has reached the target position the *Path Tracker* loses its activation and the *Local Exploration* continues the exploration. This scheme is repeated more or less frequently. It takes about 20 minutes to explore the environment as shown in figure 9 with dimensions of about 28 times 25 meters. The duration strongly

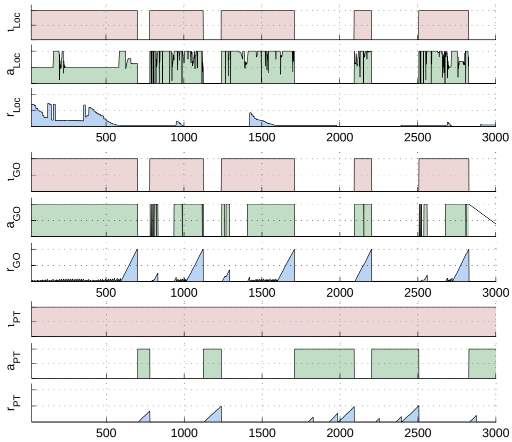

Figure 8: Activities of the applied global behaviors by the time in $^1/_{10}$ seconds.

depends on the amount of objects inside of the rooms and the number of narrow passages that slow the robot down.

In reality this exploration strategy could not be tested as good as in simulation. The reason for that is the limited possibility to detect the walls of rooms, if the sight to them is disturbed by objects on the floor. Therefore the real experiments have taken place inside an empty corridor similar to the one inside of the simulation. For these exploration runs only the local strategies excluding the door driving behavior have been used because of the limitations which are linked to mapping problems. Separately the other behaviors like *Path Tracker* or the *Local Door Driving* have been tested as well and promise good results for global exploration in reality.

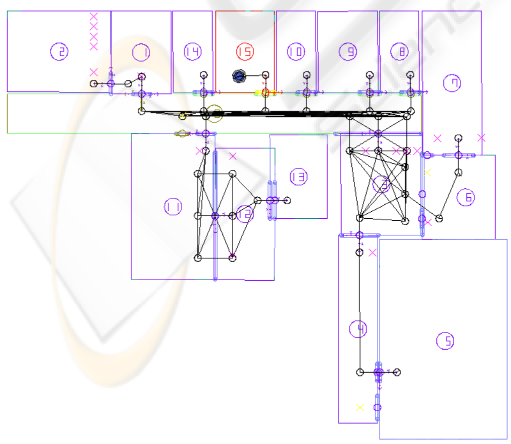

Figure 9: Autonomously explored simulated environment with navigation graph.

8 CONCLUSION

This paper introduced a hierarchical behavior-based exploration system. Three main components are shown and validated in simulated and real experiments: The *Local Exploration* to explore the current region, a *Path Tracker* behavior for navigation tasks and a *Global Observer* to fulfill a global strategy.

Future Work will mainly consist of an enhancement of the room recognition methods and the integration of more sensors. Furthermore the mapping itself could be upgraded by integrating new map objects. At last additional exploration behaviors will follow that utilize sensors like microphones, cameras or ultrasonic sensors.

REFERENCES

Albiez, J., Luksch, T., Berns, K., and Dillmann, R. (2003). A behaviour network concept for controlling walking machines. In *2nd International Symposium on Adaptive Motion of Animals and Machines (AMAM)*, Kyoto, Japan.

Brooks, R. (1986). A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, RA-2(1):14–23.

Kleinlützum, K., Luksch, T., Schmidt, D., and Berns, K. (2005). Selbstständige erstellung einer abstrakten topologiebasierten karte für die autnome exploration. In *Autonome Mobile Systeme (AMS) 2005*. (in german).

Proetzsch, M., Luksch, T., and Berns, K. (2005). Fault-tolerant behavior-based motion control for offroad navigation. In *20th IEEE International Conference on Robotics and Automation (ICRA)*, Barcelona, Spain.

Schmidt, D. (2006). Entwicklung einer verhaltensbasierten steuerung zur autonomen exploration in strukturierter umgebung von einem indoor-roboter. Diploma thesis (in german), unpublished, Robotics Research Laboratory - Kaiserslautern University of Technology.

Sim, R. and Roy, N. (2005). Global a-optimal robot exploration in slam. Technical report, Department of Computer Science, University of British Columbia.

Surmann, H., Nüchter, A., and Hertzberg, J. (2003). An autonomous mobile robot with a 3d laser range finder for 3d exploration and digitalization of indoor environments. In *Robotics and Autonomous Systems*, pages 181–198. Elsevier B.V.

Wullschleger, F., Arras, K., and Vestli, S. (1999). A flexible exploration framework for map building. Technical report, Institute of Robotics, Swiss Federal Institute of Technology ETHZ.

Yamauchi, B., Schultz, A. C., and Adams, W. (1998). Mobile robot exploration and map-building with continuous localization. In *Proceedings of the 1998 IEEE International Conference on Robotics and Automation (ICRA)*, pages pp. 3715–3720, Leuven, Belgium.