Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence

Occupancy Grid Models for Robot Mapping in Changing Environments

Daniel Meyer-Delius
KUKA Laboratories GmbH
D-86165 Augsburg, Germany
daniel.meyer-delius@kuka.com

Maximilian Beinhofer and **Wolfram Burgard**
University of Freiburg
D-79110 Freiburg, Germany
{beinhofe, burgard}@informatik.uni-freiburg.de

Abstract

The majority of existing approaches to mobile robot mapping assumes that the world is static, which is generally not justified in real-world applications. However, in many navigation tasks including trajectory planning, surveillance, and coverage, accurate maps are essential for the effective behavior of the robot. In this paper we present a probabilistic grid-based approach for modeling changing environments. Our method represents both, the occupancy and its changes in the corresponding area where the dynamics are characterized by the state transition probabilities of a Hidden Markov Model. We apply an offline and an online technique to learn the parameters from observed data. The advantage of the online approach is that it can dynamically adapt the parameters and at the same time does not require storing the complete observation sequences. Experimental results obtained with data acquired by real robots demonstrate that our model is well-suited for representing changing environments. Further results show that our technique can be used to substantially improve the effectiveness of path planning procedures.

Introduction

Accurate maps of the environment are essential for many mobile robot navigation tasks such as path planning, surveillance, or coverage. Although real-world environments generally are dynamic, most existing mapping approaches assume the environment to be static. Accordingly, they do not explicitly account for potential changes in the environment. In several navigation systems, the robots use maps that are built in an offline phase and not changed during operation (Burgard et al. 2000; Montemerlo, Roy, and Thrun 2003; Quigley et al. 2009). Although there are robust approaches that can handle inconsistencies between the map and the actual measurements, largely inconsistent models can lead to suboptimal navigation behavior or maybe navigation failure.

In this paper we present a generalization of occupancy grids (Moravec and Elfes 1985), one of the most popular mapping approaches in mobile robotics. The fundamental difference between our approach and occupancy grids is that while occupancy grids regard the state of a cell as static, our representation explicitly models state changes, which makes it particularly suited for changing environments. Our method applies cell-specific hidden Markov models (HMM) to represent the belief about the occupancy state and state transition probabilities of each grid cell. The HMM framework provides efficient algorithms for estimating the model parameters. This allows us to learn how the environment changes from observations made by the robot. In addition to the standard offline learning approach, we apply a variant that is able to learn these parameters in an online fashion and only requires the storage of a constant amount of information extracted from the observations made by the robot. Within the framework we can efficiently estimate the occupancy state of a cell from the observed evidence as it becomes available, making it possible to adapt the representation continuously over time. Accordingly, our approach provides more accurate state estimates. Furthermore, the learned parameters allow us to make predictions about the future occupancy state of a cell.

The contribution of this work is a mapping approach for changing environments that represents the occupancy of the space and explicitly characterizes how this occupancy changes over time. We describe our model, how the representation can be updated as new observations become available, and how to predict future occupancy states. Furthermore, we present two techniques, one offline and one online, to estimate the state transition probabilities of the model from observed data. We evaluate our approach in simulation and using real-world data. The results demonstrate that our model yields highly accurate maps for changing environments. Furthermore, we describe how the explicit representation of the dynamics in the environment can be used to improve the path planning performance of a robot.

Related Work

Previous work on mapping changing environments can be divided into two categories: approaches that filter out sensor measurements caused by non-static elements and approaches that explicitly model aspects of the environment dynamics. Filtering out sensor measurements is typically based on probabilistic sensor models that identify the measurements which are inconsistent with a reference model of the environment (Fox, Burgard, and Thrun 1999; Burgard et al. 2000; Hähnel et al. 2003). In contrast to these approaches, the method presented in this paper explicitly characterizes the dynamics of the environment in the representation itself instead of relying on specific sensor models.

Several authors have proposed augmented representations of the environment that explicitly represent dynamic objects. The approaches of Anguelov et al. (2002) and Biswas et al. (2002), for example, compute shape models of non-stationary objects. They create maps at different points in time and compare those maps using an EM-based algorithm to identify the parts of the environment that change. Similarly, Gallagher et al. (2009) build maps for individual objects which can then be overlaid to represent the current configuration of the environment. Petrovskaya and Ng (2007) extend occupancy grid maps with parameterized models of dynamic objects like doors and apply a Rao-Blackwellized particle filter to estimate the pose of the robot and the state of these objects. Wang et al. (2007) estimate the state of non-parametric dynamic objects using Kalman filters while at the same time constructing a map of the static objects in the environment. The above-mentioned approaches are based on the identification and modeling of dynamic objects in the environment. Our approach, in contrast, does not depend on dynamic object detection and high-level object models; it considers only the occupancy of the space at a lower level of abstraction.

The problem of modeling the occupancy of the space in changing environments has also been addressed in the past. Murphy (1999), for example, presents a grid-based representation of the environment and explicitly models occupancy state changes in the context of robot localization. Chen et al. (2006) and Brechtel, Gindele, and Dillmann (2010) use a similar grid-based representation for changing environments to track dynamic objects. These approaches, in contrast to our method, assume that the state transitions are a priori given and do not include algorithms to learn them from data observed by the robot.

Wolf and Sukhatme (2005) propose a model that maintains two separate occupancy grids, one for the static parts of the environment and the other for the dynamic parts. Biber and Duckett (2005) present an approach to represent the environment on multiple timescales simultaneously. For each timescale a separate sample-based representation is maintained and updated using the observations of the robot according to an associated timescale parameter. Konolige and Bowman (2009) describe a vision-based system where camera views are grouped into clusters that represent different persistent configurations of the environment. Changes in the environment are handled by deleting views based on a least-recently-used principle. While these approaches are also able to continuously adapt to changes over time, the fundamental difference to our method is that we provide an explicit characterization of the changes of the environment.

Occupancy Grids for Changing Environments

Occupancy grids, as they were introduced by Moravec and Elfes (1985), are based upon a regular tessellation of the space into a number of rectangular cells. They store in each cell c of this grid the probability $p(c)$ that the corresponding area in the environment is occupied by an obstacle. One particular assumption in the context of occupancy grids is that the environment is static. Accordingly, there is no explicit model about how the occupancy changes over time in this approach. The approach described in this paper overcomes this limitation. It applies HMMs (see Rabiner (1989)) to explicitly represent both the belief about the occupancy state and state transition probabilities of each grid cell.

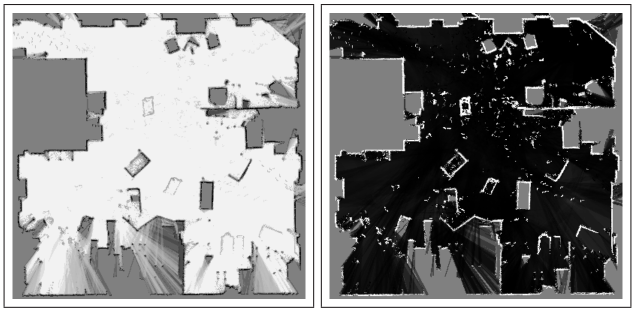

Figure 1: State transition probabilities in a small hall where people usually go from the office in the top left corner to one of the two exits on the right. The left and right images correspond to the distributions $p_c^{o|f}$ and $p_c^{f|o}$ respectively: the darker the color, the larger the state change probability.

An HMM requires the specification of a state transition probability, an observation model, and an initial state distribution. Let c_t be a discrete random variable that represents the occupancy state of a cell c at time t. The initial state distribution $p(c_0)$ specifies the occupancy probability of a cell at the initial time step $t = 0$ prior to any observation.

The state transition model $p(c_t \mid c_{t-1})$ describes how the occupancy state of cell c changes between consecutive time steps. We assume that the changes in the environment are caused by a stationary process, that is, the state transition probabilities are the same for all time steps t. Since a cell is either free (*free*) or occupied (*occ*), the state transition model can be specified using only two transition probabilities, namely

$$p_c^{o|f} = p(c_t = occ \mid c_{t-1} = free) \qquad (1)$$

and

$$p_c^{f|o} = p(c_t = free \mid c_{t-1} = occ)\,. \qquad (2)$$

Note that, by assuming a stationary process, these probabilities do not depend on the absolute value of t. Therefore, the dynamics of a cell at any time can be captured by its transition matrix

$$A_c = \begin{bmatrix} 1 - p_c^{f|o} & p_c^{f|o} \\ p_c^{o|f} & 1 - p_c^{o|f} \end{bmatrix}. \qquad (3)$$

This transition model generalizes the one utilized by Murphy (1999) and Chen et al. (2006) where $p_c^{o|f} = p_c^{f|o}$. Figure 1 depicts the transition probabilities in a small hall. The figure shows areas where highly dynamic elements like people usually move and the areas where more static elements such as furniture and walls are placed as having different state change probabilities.

The observation model $p(z \mid c)$ represents the likelihood of the observation z given the state of the cell c. In this paper, the observations correspond to range measurements obtained, for example, with a laser range scanner. For a given cell and measurement we consider two cases: the measurement ends up in the cell (a *hit*) or it goes through the cell without ending in it (a *miss*). Accordingly, the observation model can also be specified using two probabilities: $p(z = \mathit{hit} \mid c = \mathit{free})$ and $p(z = \mathit{hit} \mid c = \mathit{occ})$. We additionally take into account the situation where a cell is not covered by the measurement. This is necessary since the transition model characterizes state changes only for consecutive time steps and not every cell is observed at each time step. Explicitly considering this *no-observation* case allows us to update and estimate the parameters of the model using the HMM framework directly without having to artificially distinguish between cells that are observed and cells that are not. The concrete observation probability for a *no-observation* does not affect the results as long as the proportion between the two remaining probabilities remains unchanged. The probabilities can be specified by three matrices

$$B_z = \begin{bmatrix} p(z \mid c = \mathit{occ}) & 0 \\ 0 & p(z \mid c = \mathit{free}) \end{bmatrix} , \qquad (4)$$

where $z \in \{\mathit{hit}, \mathit{miss}, \mathit{no\text{-}observation}\}$.

From the discussion above it can be seen that occupancy grids are a special case of our model where the transition probabilities $p_c^{o|f}$ and $p_c^{f|o}$ are 0 for all cells c.

Occupancy State Update

The update of the occupancy state of the cells follows a Bayesian approach. The goal is to estimate the posterior distribution $p(c_t \mid z_{1:t})$ over the current occupancy state c_t of a cell given all the available evidence $z_{1:t}$ up to time t. The update formula is:

$$p(c_t \mid z_{1:t}) = \\ \eta \, p(z_t \mid c_t) \sum_{c_{t-1}} p(c_t \mid c_{t-1}) \, p(c_{t-1} \mid z_{1:t-1}) , \qquad (5)$$

where η is a normalization constant. The structure of our particular HMMs allows for a simple and efficient implementation of this recursive approach. Utilizing the matrix notation and defining the posterior at time t as the vector

$$Q_t = [p(c_t = \mathit{occ} \mid z_{1:t}) \quad p(c_t = \mathit{free} \mid z_{1:t})] , \qquad (6)$$

one can easily compute the posterior at time $t + 1$ as

$$Q_{t+1} = Q_t \, A_c \, B_{z_{t+1}} \, \eta . \qquad (7)$$

Note that replacing the sum in (5) with the posterior $p(c_t \mid z_{1:t-1})$, or equivalently, the matrix A in (7) with the identity matrix I, we obtain the map update approach for occupancy grids described by Thrun, Burgard, and Fox (2005), which is widely used in mobile robotics.

State Prediction

Equation (7) can be used directly to estimate the future state of a cell or estimate the current state of a cell that has not been observed recently. Updating Q_t with k consecutive *no-observation* readings results in an unbiased estimator for Q_{t+k}. As k tends to infinity, the occupancy value of a cell converges to a unique stationary distribution π_c. As in (7) $B_{\mathit{no\text{-}observation}} \, \eta = I$, the stationary distribution must satisfy

$$\pi_c = \pi_c \, A_c . \qquad (8)$$

From (8), the stationary distribution for our concrete HMMs can be computed as

$$\pi_c = \frac{1}{p_c^{o|f} + p_c^{f|o}} \begin{bmatrix} p_c^{o|f} & p_c^{f|o} \end{bmatrix} . \qquad (9)$$

The time needed for the occupancy value of a cell to converge to π_c its called the *mixing time*. In this paper we consider the *total variation distance* (see Levin, Peres, and Wilmer (2006)) to measure the distance between the two distributions. Since our HMMs have only two states, the total variation distance Δ_k between the stationary distribution π_c and the occupancy distribution $p(c_t)$ at time t can be specified as

$$\Delta_k = \left|1 - p_c^{o|f} - p_c^{f|o}\right|^k \Delta_0 , \qquad (10)$$

where

$$\Delta_0 = |p(c_t = \mathit{occ}) - \pi_c(\mathit{occ})| = |p(c_t = \mathit{free}) - \pi_c(\mathit{free})|$$

is the difference between the occupancy probability $p(c_t)$ of a cell at time t and its stationary distribution π_c.

Based on the total variation distance, an approximation of the mixing time $k_{\mathit{mix}}(\epsilon)$ can then be defined as the smallest k such that Δ_k is less than a given value ϵ. For a given ϵ, it can be analytically computed as

$$k_{\mathit{mix}}(\epsilon) = \left\lceil \frac{\ln(\epsilon/\Delta_0)}{\ln(|1 - p_c^{o|f} - p_c^{f|o}|)} \right\rceil , \qquad (11)$$

where $\lceil \cdot \rceil$ is the ceiling operator. Equation (11) can be derived from (10) by straightforward algebraic operations. Note that this computation is valid only when $\Delta_0 > 0$, for the opposite case, $k_{\mathit{mix}}(\epsilon)$ is trivially 0.

Being able to compute the mixing time of a cell has important practical implications, since it renders the computation of predicted occupancy values beyond the mixing time unnecessary. This can be exploited to update successively unobserved parts of the grid map in an efficient way.

Parameter Estimation

As mentioned above, an HMM is characterized by the state transition probabilities, the observation model, and the initial state distribution. We assume that the observation model only depends on the sensor used and not on the location. Therefore it can be specified beforehand and is the same for each HMM. We also assume the same uniform initial state distribution for each cell. Thus, the parameters to be learned are the state transition probabilities of the cell.

One of the most popular approaches for estimating the parameters of an HMM from observed data is an instance of the expectation-maximization (EM) algorithm. The basic idea is to iteratively estimate the model parameters using the observations and the parameters estimated in the previous iteration until the values converge. Let $\hat{\theta}^{(n)}$ represent the parameters estimated at the n-th iteration. The EM algorithm results in the following re-estimation formula for the transition model of cell c:

$$\hat{p}(c_t = i \mid c_{t-1} = j)^{(n+1)} = \frac{\sum_{\tau=1}^{T} p(c_{\tau-1} = i, c_\tau = j \mid z_{1:T}, \hat{\theta}^{(n)})}{\sum_{\tau=1}^{T} p(c_{\tau-1} = i \mid z_{1:T}, \hat{\theta}^{(n)})}, \quad (12)$$

where $i, j \in \{free, occ\}$ and T is the length of the observation sequence. Note that the probabilities on the right-hand side are conditioned on the observation sequence $z_{1:T}$ and the previous parameter estimates $\hat{\theta}^{(n)}$. The probabilities in (12) can be efficiently computed using a forward-backward procedure (Rabiner 1989).

The mentioned algorithm, however, is an offline approach that requires storing the complete observation sequence for each cell. An online version of the algorithm was derived by Mongillo and Deneve (2008). To calculate the transition probabilities in (12), this algorithm only needs to store the sufficient statistics

$$\phi_{ijh}(t; \hat{\theta}) = \frac{1}{t} \sum_{\tau=1}^{t} \delta(z_\tau, h)\, p(c_{\tau-1} = i, c_\tau = j \mid z_{1:t}, \hat{\theta}), \quad (13)$$

where $i, j \in \{\mathit{free}, \mathit{occ}\}$, $h \in \{\mathit{hit}, \mathit{miss}, \mathit{no\text{-}observation}\}$, and $\delta(z_\tau, h) = 1$ if $z_\tau = h$ and 0 otherwise. Dropping the dependence on $\hat{\theta}$, only 16 values have to be stored. The algorithm uses ϕ_{ijh} instead of the probabilities computed with the forward-backward procedure to estimate the transition model. Therefore it implements only a partial expectation step, while the maximization step remains exact.

Besides being an online approach with small storage requirements, the pre-factor $1/t$ in (13) allows the algorithm to handle non-stationary environment dynamics. Additionally, the algorithm updates with each observation the occupancy state. These properties make the online version of the EM algorithm an attractive alternative for systems operating over extended periods of time.

Experimental Evaluation

We implemented our proposed model and tested it in simulation and using data obtained with a real robot. The goal of the experiments was to evaluate the quality and usefulness of our representation.

Accuracy of the Representation

In a first experiment we evaluated the accuracy of our proposed mapping approach. We steered a MobileRobot Powerbot equipped with a SICK LMS laser range finder through a parking lot. We recorded twelve data sets $\{d_1, \ldots, d_{12}\}$ that consist of laser range data and odometry collected at every full hour from 7am until 6pm during one day. Accordingly, the data correspond to twelve different configurations

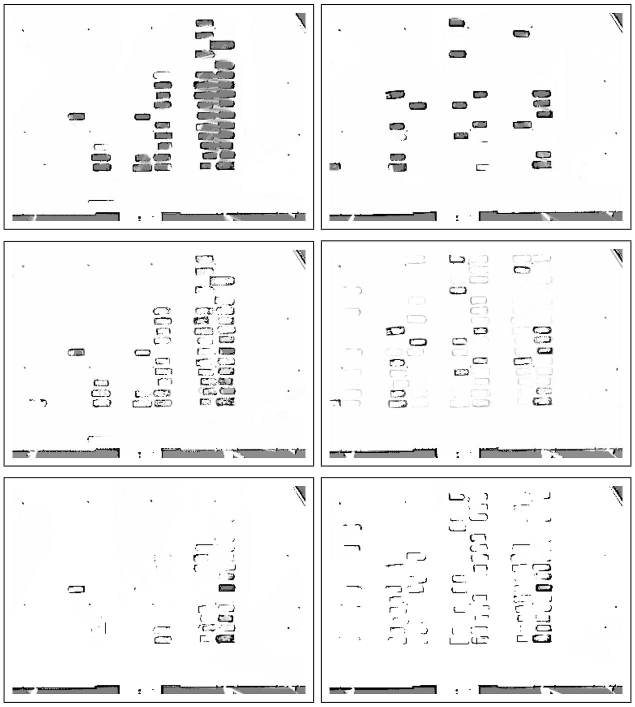

Figure 2: Qualitative evaluation of our proposed model. Shown are the ground truth maps m_3 (top left) and m_{12} (top right) , our maps $m^\star_{1:3}$ and $m^\star_{1:12}$ (middle row, left and right), and standard occupancy grids $m_{1:3}$ and $m_{1:12}$ (bottom row, left and right).

of parked cars. We used a simultaneous localization and mapping (SLAM) approach (Grisetti, Stachniss, and Burgard 2005) to correct the odometry of the robot and to determine its trajectory, which was then used together with the range measurements to generate the maps for our experiments. Range measurements were sampled at about 1 Hz. The path and the velocity of the robot in each of the runs were approximately the same to avoid a bias in the complete data set. We evaluated the long-term maps $m^\star_{1:i}$ built from the data sets $d_1, \ldots, d_i$ with our approach against the long-term maps $m_{1:i}$ built from the data sets $d_1, \ldots, d_i$ with the standard occupancy grid mapping approach described by (Thrun, Burgard, and Fox 2005). Since, during each of the individual runs, the parking lot did not change considerably, we used the standard occupancy grid maps m_i built from the data set d_i of the single runs as ground truth.

Figure 2 shows a qualitative comparison between our representation and standard occupancy grids. Here, we used the online EM approach to learn the parameters of our model. In the figure, the ground truth grid-maps m_3 and m_{12} are shown in the first row. While the second row contains the maps $m^\star_{1:3}$ and $m^\star_{1:12}$ that were obtained with our method, the third row depicts the standard occupancy grids $m_{1:3}$ and $m_{1:12}$. As can be seen our representation quickly adapts to the changes in the parking lot. Thus it constitutes a more accurate representation of the environment than standard occupancy grid maps at any point in time. Additionally, our

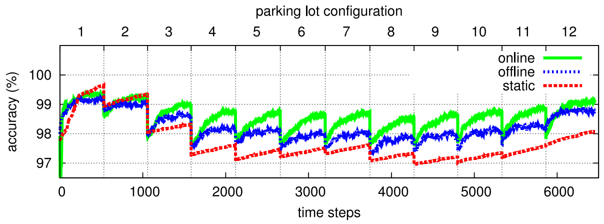

Figure 3: Accuracy of the representation over time for the parking lot data. Both online and offline parameter estimation approaches were evaluated.

model also represents the probable occupancy of areas that have not been recently observed. The light-gray rectangles in the map $m^{\star}_{1:12}$, for example, correspond to predicted reappearances of cars in the parking lot.

To quantitatively evaluate the accuracy of our representation, we computed its accuracy with respect to the ground truth maps. In this context, accuracy is defined as the number of correctly classified cells divided by the total number of classified cells. A cell c was classified as *occupied* if $p(c) > 0.5$ and as *free* if $p(c) < 0.5$. This accuracy indirectly reflects the quality of the learned model parameters. Figure 3 compares the accuracy of a standard occupancy grid (*static*) against that of our model obtained using the online (*online*) and offline (*offline*) parameter estimation approaches. The figure plots the accuracy of the grids over time for the parking lot data. After each configuration change, the accuracy of our model quickly starts to increase as the representation adapts to the new configuration. Occupancy grids adapt relatively quickly at first, but their adaptability decreases with the number of observations already integrated into the map. Note that our mapping approach is not restricted to slowly changing or semi-static environments like the parking lot and can also be utilized in more dynamic environmets like the one shown in Figure 1.

Influence of the Environment's Dynamics

The goal of this experiment was to evaluate the accuracy of our proposed representation for different environment dynamics. In the context of occupancy grids, the dynamics of the environment are characterized by the number of dynamic cells and their state change probabilities. To generate data corresponding to different environment dynamics we used a 50×50 grid map and changed the fraction of dynamic cells and state change probabilities. We randomly selected the locations of the dynamic cells (a more involved simulation would have yielded the same results) and estimated the state transition probabilities using both the online and offline approaches for different data sets. Figure 4 compares the resulting model accuracy against that of a standard occupancy grid for two different settings: one relatively static and another more dynamic. The curves correspond to the mean and standard deviation for 10 repetitions of the experiment. As can be seen, our model represents the environment more accurately than standard occupancy grids even for moderately dynamic environments.

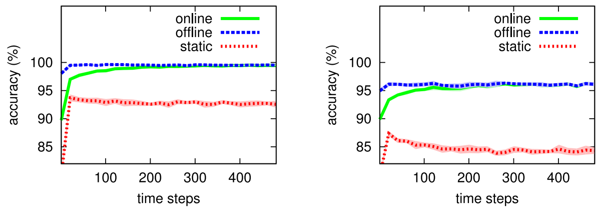

Figure 4: Accuracy of the representation for different configurations of dynamic cells in a simulated grid map. Left: $5\,\%$ dynamic cells and $5\,\%$ state change probability. Right: $25\,\%$ dynamic cells and $25\,\%$ state change probability

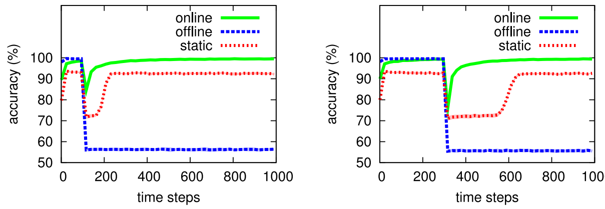

Figure 5: Accuracy of the representation when the dynamics of the environment changes. The left and right plots correspond to changes after 100 and 300 time steps respectively.

Parameter Estimation

As can be seen in Figure 4, the offline approach produces more accurate results at first, but the difference between the results of the two approaches decreases over time. This suggests that, regarding the accuracy of the representation, both parameter estimation techniques are comparable.

Although the offline approach produces good results from the beginning, it requires storing all observations for each cell in the grid for an a priori training phase. Furthermore, being an offline approach, once the parameters have been estimated, they remain fixed. This makes the offline approach less appropriate for environments in which the assumption that the dynamics are stationary does not hold. In such situations the online approach provides better results, since it continually adapts its parameters as new observations become available. Figure 5 illustrates the effects of a change in the environment dynamics on the accuracy of the representation. The model parameters were obtained with the online and offline approaches for $5\,\%$ dynamic cells and $5\,\%$ state change probability (corresponding to the left plot in Figure 4). For the experiment, the change consisted in selecting a new set of dynamic and static cells. The number of dynamic cells and their state change probabilities remained the same, but we obtained similar results when these parameters were changed as well. As can be seen in the figure, using the online approach, the accuracy of the representation quickly returns to its value before the change. This is the consequence of the model parameters adapting to the new environment dynamics. The accuracy of the representation, whose parameters were estimated offline, drops when the dynamics change and remains low. We also evaluated

values in %	*static*	*online*	*offline*	*random*	*optimal*
short (optimal)	11.25 (±16.35)	25.50 (±22.24)	40.75 (±20.15)	24.25 (±10.17)	40.50 (±19.99)
long (optimal)	35.75 (±14.80)	27.00 (±18.38)	15.00 (±9.03)	24.25 (±13.70)	17.25 (±9.39)
indirect long	4.75 (±1.12)	19.50 (±12.24)	30.50 (±8.72)	23.50 (±9.33)	27.75 (±9.24)
unnecessary indirect long	0.25 (±1.12)	1.50 (±2.86)	2.75 (±3.02)	1.50 (±2.86)	2.25 (±3.02)
unnecessary direct long	48.00 (±16.89)	26.50 (±21.34)	11.00 (±5.28)	26.50 (±13.19)	12.25 (±6.38)
effectiveness	47.00 (±16.89)	52.50 (±16.10)	55.75 (±13.21)	48.50 (±10.14)	57.75 (±14.09)

Table 1: Occurrences of different types of planned paths for different environment models and parameter estimation approaches.

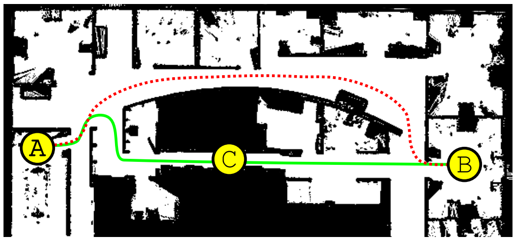

Figure 6: Experimental setup for the path planning experiment. The task consists in navigating between A and B. At C we added a virtual door that changed its state over time.

the behavior of a standard occupancy grid. As expected, the number of observations needed by the occupancy grid to correctly represent the new static cells is approximately the same as the number of previous observations.

Path Planning

In the previous experiments we showed that our representation readily adapts to changes in the environment. The goal of this experiment was to show that this adaptability can be used to improve the path planning performance of a robot. The experiment was performed in simulation within the environment shown in Figure 6. The task of the robot was to navigate between positions A and B. We added a virtual door at position C along the shortest path between A and B. To generate the a priori map needed for path planning and obtain data for estimating the parameters of our model, we steered the robot through the relevant parts of the environment in an offline phase.

We applied the $A^\star$ algorithm and performed re-planning at every time step. The traversal cost for each cell was computed based on its occupancy state at the corresponding moment (1 for *free* and ∞ for *occupied* cells after a clipping operation). We then performed 20 repetitions of the experiment. In each of them, the robot executed 20 runs from one position to the other.

To abstract from the specific structure of the environment, we performed our evaluation based on a classification of the trajectories followed by the robot into five different types and compared the numbers of occurrences of each different class. We opted for this qualitative evaluation criterion since other performance measures, like traveled distance or execution time, largely depend on the particular environment used in the experiment, which can arbitrarily bias the results in either direction. We considered the following five classes: When the robot took the shortest path: *short*. When the robot took the longer path: *long*. In this case, the longer path was the optimal choice. When the robot tried to take the shortest path first, found the door closed, and then took the longer path: *indirect long*. In this case, taking the longer path from the beginning would have been the optimal choice. When the robot tried to take the shortest path first, found the door closed, and then followed the longer path: *unnecessary indirect long*. In this case, however, continuing to follow the shortest path would have been the optimal choice since the door would have opened for the robot to pass. When the robot took the longer path and no attempt was made to follow the shortest path: *unnecessary direct long*. In this case, taking the shortest path from the beginning would have been the optimal choice.

The values in Table 1 correspond to the occurrences (average and standard deviation) of the different trajectory types during the repetitions of the experiment. We compared the path planning performance when using a standard occupancy grid (*static*), our model obtained using the online approach (*online*), and using the offline approach (*offline*). The number of occurrences of *short* and *long* trajectories in the table indicate that the information about the state change probability of the door represented in our model leads to better path planning performances. Once the door is represented as closed in the occupancy grid (*static*), the robot never attempts to follow the shortest path again. This can be seen in the table by the small number of *short*, *indirect long*, and *unnecessary indirect long* trajectories and explains the large number of *long* trajectories. Using our model, on the other hand, the state of the (unobserved) door in the map changes over time according to the learned state transition probabilities. Whenever the cells corresponding to the door are classified as free, the robot attempts to follow the shortest path.

We additionally implemented two baseline path planning policies for comparison. In the first (*random*) the robot, when in A or B, randomly chooses between the two possible paths. In the second (*optimal*) it has perfect knowledge about the state change probability of the door and based on its internal belief about the state of the door chooses the expected optimal path. The percentage of runs in which the optimal path was followed (*effectiveness*) by this robot is an empirical upper bound for the effectiveness achievable in this experiment. One-sided t-tests revealed that the effectiveness of *offline* was significantly higher than that of *static* and *ran-*

dom on a 5% level. Between the effectiveness of *offline* and *optimal* we could not find a significant difference. The results indicate that, even using a standard planning approach, our maps lead to an improved path planning performance.

Conclusions

In this paper we described an approach to occupancy grid mapping that utilizes HMMs to represent the occupancy of each grid cell and explicitly models how this occupancy changes over time. We described an offline and an online learning technique to estimate the model parameters from data. We furthermore discussed how our maps can be updated as new observations become available and how to predict the future state of a cell. We evaluated our approach in simulation and using real-world data. The results demonstrate that our model is well-suited for representing changing environments and it can also be utilized to improve path planning performance of a mobile robot.

In our future work we want to relax the assumption that the pose of the robot is known when the robot is building the map. More specifically, we plan to integrate our model into a simultaneous localization and mapping approach. Additionally, we will investigate how to exploit the information encoded in the state transition probabilities of the HMMs in a customized path planning approach.

Acknowledgements

This work has been partially supported by the European Commission under contract numbers FP7-231888-EUROPA and FP7-260026-TAPAS, and by the German Research Foundation within the Research Training Group 1103.

References

Anguelov, D.; Biswas, R.; Koller, D.; Limketkai, B.; Sanner, S.; and Thrun, S. 2002. Learning hierarchical object maps of non-stationary environments with mobile robots. In *Proc. of the Conference on Uncertainty in AI (UAI)*.

Biber, P., and Duckett, T. 2005. Dynamic maps for long-term operation of mobile service robots. In *Proc. of Robotics: Science and Systems (RSS)*.

Biswas, R.; Limketkai, B.; Sanner, S.; and Thrun, S. 2002. Towards object mapping in non-stationary environments with mobile robots. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*.

Brechtel, S.; Gindele, T.; and Dillmann, R. 2010. Recursive importance sampling for efficient grid-based occupancy filtering in dynamic environments. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*.

Burgard, W.; Cremers, A.; Fox, D.; Hähnel, D.; Lakemeyer, G.; Schulz, D.; Steiner, W.; and Thrun, S. 2000. Experiences with an interactive museum tour-guide robot. *Artificial Intelligence* 114(1-2).

Chen, C.; Tay, C.; Laugier, C.; and Mekhnacha, K. 2006. Dynamic environment modeling with gridmap: A multiple-object tracking application. In *Proc. of the IEEE Int. Conf. on Control, Automation, Robotics and Vision (ICARCV)*.

Fox, D.; Burgard, W.; and Thrun, S. 1999. Markov localization for mobile robots in dynamic environments. *Journal of Artificial Intelligence Research* 11:391–427.

Gallagher, G.; Srinivasa, S. S.; Bagnell, J. A.; and Ferguson, D. 2009. GATMO: A generalized approach to tracking movable objects. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*.

Grisetti, G.; Stachniss, C.; and Burgard, W. 2005. Improving grid-based SLAM with Rao-Blackwellized particle filters by adaptive proposals and selective resampling. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*.

Hähnel, D.; Triebel, R.; Burgard, W.; and Thrun, S. 2003. Map building with mobile robots in dynamic environments. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*.

Konolige, K., and Bowman, J. 2009. Towards lifelong visual maps. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*.

Levin, D. A.; Peres, Y.; and Wilmer, E. L. 2006. *Markov chains and mixing times*. American Mathematical Society.

Mongillo, G., and Deneve, S. 2008. Online learning with hidden Markov models. *Neural Computation* 20:1706–1716.

Montemerlo, M.; Roy, N.; and Thrun, S. 2003. Perspectives on standardization in mobile robot programming: The carnegie mellon navigation (CARMEN) toolkit. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*.

Moravec, H., and Elfes, A. 1985. High resolution maps from wide angle sonar. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*.

Murphy, K. 1999. Bayesian map learning in dynamic environments. *Advances in Neural Information Processing Systems NIPS* 12:1015–1021.

Petrovskaya, A., and Ng, A. Y. 2007. Probabilistic mobile manipulation in dynamic environments, with application to opening doors. In *Proc. of the Int. Conf. on Artificial Intelligence (IJCAI)*.

Quigley, M.; Gerkey, B.; Conley, K.; Faust, J.; Foote, T.; Leibs, J.; Berger, E.; Wheeler, R.; and Ng, A. 2009. ROS: an open-source robot operating system. In *Proc. of the Workshop on Open Source Software at IEEE Int. Conf. on Robotics & Automation (ICRA)*.

Rabiner, L. 1989. A tutorial on hidden Markov models and selected applications in speech recognition. In *Proceedings of the IEEE*, volume 77 (2), 257–286.

Thrun, S.; Burgard, W.; and Fox, D. 2005. *Probabilistic Robotics*. MIT Press.

Wang, C.-C.; Thorpe, C.; Thrun, S.; Hebert, M.; and Durrant-Whyte, H. 2007. Simultaneous localization, mapping and moving object tracking. *The International Journal of Robotics Research* 26(9):889–916.

Wolf, D. F., and Sukhatme, G. S. 2005. Mobile robot simultaneous localization and mapping in dynamic environments. *Autonomous Robots* 19(1):53–65.