2014 IEEE International Conference on Robotics & Automation (ICRA)
Hong Kong Convention and Exhibition Center
May 31 - June 7, 2014. Hong Kong, China

A Probabilistic Approach to High-Confidence Cleaning Guarantees for Low-Cost Cleaning Robots

Jürgen Hess   Maximilian Beinhofer   Wolfram Burgard

***Abstract*— Cleaning is widely regarded as one of the most relevant applications of autonomous service robots. The goal of robotic cleaning is to achieve low dirt levels in the whole environment. Low cost consumer robots, however, are typically prone to high motion and sensor uncertainties. Additionally, their cleaning units do not always remove the dirt entirely. As a result, there is a substantial probability that some parts of the environment are not cleaned sufficiently. In this paper, we propose an approach to robotic cleaning that guarantees that in the whole environment, the dirt levels after cleaning are reduced below a user-defined threshold with high confidence. We introduce a novel probabilistic model for jointly estimating the trajectory of the robot and the current dirt distribution in the environment. Based on this estimate, we adapt the future cleaning path during operation such that the robot re-visits areas in which high dirt levels are still likely. We demonstrate the effectiveness of our approach in extensive experiments carried out both in simulation and with a real vacuum cleaning robot, also in comparison to previous approaches.**

I. Introduction

Cleaning robots are one of the most popular consumer products applying robotics technologies. They aim at removing the dirt in the entire environment. Low cost consumer robots, however, are typically subject to high motion and sensor uncertainties and therefore might miss certain areas. Additionally, these robots do not always remove the dirt entirely on the first pass. Thus, some parts of the environment may need repeated cleaning. Obviously, if the robot knows which trajectory it took thus far and what the results of its cleaning actions were, it can more easily identify such parts.

In this paper, we propose a probabilistic model for jointly estimating the trajectory of the robot and the current distribution of dirt in the environment. We furthermore show how this model can be used to achieve low dirt levels. We consider a cleaning robot with a range sensor to perceive its environment and a dirt sensor to measure the amount of dirt currently cleaned. The robot has access to an occupancy map of its environment and a prior estimate of the dirt distribution in the environment. Based on the prior dirt distribution, we select the parts of the environment that need to be cleaned (see Fig. 1 for an example) and compute an initial cleaning path. We model the execution of the cleaning path in a probabilistic framework. In addition to the uncertainties of the odometry and the range sensor, we explicitly model the uncertainties of the cleaning unit and the dirt sensor. Using this framework, we jointly estimate the trajectory of the robot and the current dirt distribution. This estimate enables the robot to identify the parts of the environment that have not yet been sufficiently cleaned. We exploit this knowledge for iteratively adapting the cleaning path during operation, until the maximum dirt level in the environment is below a user-defined threshold with high confidence.

All authors are with the University of Freiburg, Germany. This work has partially been supported by the German Research Foundation (DFG) within the Research Training Group 1103 and under contract number SFB/TR 8.
hess@informatik.uni-freiburg.de

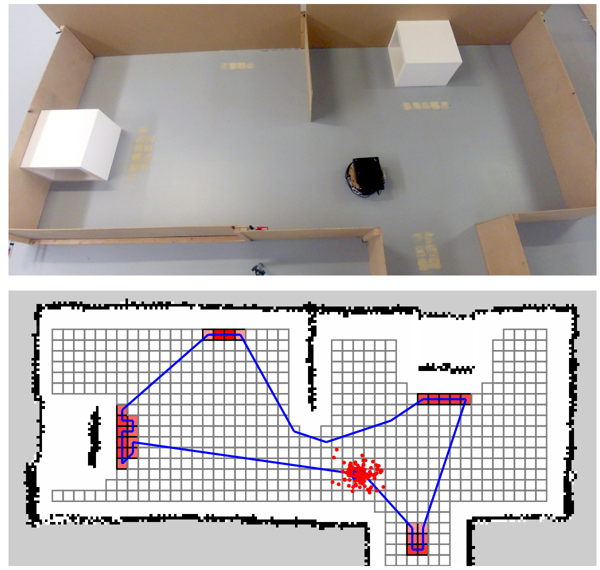

Fig. 1. The Roomba robot in the experimental environment (top) and its planned path (blue line) in the map (bottom). The bottom part shows a map of the environment (black obstacles) together with a set of particles (red dots) representing the probability distribution about the pose of the robot. The grid denotes the dirt distribution with dirty (red) and clean cells (white).

The contribution of this paper is an approach for robotic cleaning that guarantees low dirt levels in the whole environment with high confidence. Depending on the actual distribution of dirt, it typically needs considerably less time than approaches that always traverse the entire environment. Due to the estimation of the uncertainties in the task execution, our approach even works on low cost consumer robots with high noise levels. As a side effect, the integration of dirt measurements into the estimation can significantly improve the localization performance of the robot. Extensive experiments, both in simulation and with a real cleaning robot, demonstrate the above claims in practice.

II. Related Work

In the area of robotic cleaning, many systems have been developed [10], [11], [12], [15], ranging from cleaning robots for chain stores [12] to pipe cleaning robots [10]. Katsuki *et al.* [11], for example, developed a window cleaning robot

978-1-4799-3684-7/14/$31.00 ©2014 IEEE

with a dirt detection sensor and aim at guaranteeing that the robot cleans the complete window using a specialized motion control method. However, in contrast to our approach, they do not consider motion and sensor uncertainties.

Most robotic cleaners either perform random movements or do coverage path planning, i.e., they seek for the shortest path that covers the entire environment. Choset *et al.* [3] give an overview of existing coverage path planning techniques. Bretl *et al.* [2] prove that if the position and velocity errors of the robot are bounded, complete coverage can be guaranteed for a particular choice of the coverage path. In contrast, we do not aim at complete coverage, considering worst case errors beforehand, but re-plan the path during operation according to our estimate of the remaining dirt.

To estimate the joint distribution of the trajectory of the robot and the dirt in the environment, we factor the distribution similar to approaches in the area of simultaneous localization and mapping (SLAM) [17]. There exist several SLAM approaches for cleaning robots [6], [9], [18]. Gutmann *et al.* [6], for example, simultaneously estimate a vector field induced by stationary signal sources and the robot pose in the vector field. Jeong and Lee [9] propose a landmark-based SLAM algorithm for a single camera pointing towards the ceiling. The approach of Zhang *et al.* [18] shows that SLAM is also solvable even with limited range sensors like bumpers. In contrast to typical SLAM approaches, which initially have no information about the environment and estimate a static map, our approach uses an initially available occupancy grid map and an initial guess of the dirt distribution. It simultaneously estimates the pose of the robot and the changing state of the dirt distribution. Closer to our estimation framework is the work of Stachniss and Burgard [16], who jointly estimate the pose of the robot and the actual configuration of dynamic areas.

In this paper, we derive probabilistic models for the dirt actuator (cleaning unit) of the robot and for the dirt sensor inside the cleaning unit. There exists a large body of literature on modeling novel sensors and actuators for robots, e.g., air flow sensors [13], whisker sensors [14], or WiFi receivers [4]. However, to the best of our knowledge, there exists no other approach that models dirt actuators and sensors inside a probabilistic framework.

Our previous work [8] introduced a probabilistic approach for representing the distribution of dirt in the environment and proposed efficient cleaning policies. It furthermore assumed that the execution of the cleaning task is deterministic. Compared to this previous method, the approach proposed here explicitly considers the uncertainties in the entire cleaning task. Thus, it can guarantee low dirt levels with high confidence even for robots with high noise levels (see Sec. V for an experimental comparison).

III. Probabilistic Models for Localization and Cleaning

In the following, we consider a mobile cleaning robot equipped with a cleaning unit (dirt actuator), a sensor inside the cleaning unit that measures the amount of dirt sucked in at the moment (dirt sensor), and a sensor measuring the distances to the obstacles in the environment (i.e., a depth camera or a laser scanner). We assume that the robot has a geometric map of its environment (e.g., an occupancy map), which it uses to match its distance measurements against. Additionally, the robot has access to an estimate of the dirt distribution in the environment at the beginning of the cleaning cycle. To represent the dirt distribution, we follow the approach of Hess *et al.* [8] and use a regular tessellation of the environment into M grid cells. We assume that, for every grid cell c_i, the amount of dirt in the cell at time t is Poisson distributed with parameter λ_t^i. Together, the values λ_t^i for all cells c_i define the dirt distribution $\mathbf{d}_t = (\lambda_t^1, \ldots, \lambda_t^M)$. Hess *et al.* [8] show how to estimate this distribution, its dynamics over time and how it can be updated during and between cleaning runs. At the time of cleaning, we also obtain the initial dirt distribution $\mathbf{d}_0$ using these techniques.

We will now describe how to jointly estimate the trajectory already traveled by the robot and the current dirt distribution from the observations of the robot. Furthermore, we will introduce probabilistic models describing the cleaning unit and the dirt sensor of the robot.

A. Estimation of the Trajectory and the Dirt Distribution

Our goal is to estimate the joint posterior distribution of the trajectory of the robot and the dirt distribution that is

$$p(\mathbf{x}_{0:t}, \mathbf{d}_{0:t} \mid \mathbf{u}_{1:t}, \mathbf{z}^{\mathrm{r}}_{1:t}, \mathbf{z}^{\mathrm{d}}_{1:t}), \tag{1}$$

where $\mathbf{x}_{0:t}$ and $\mathbf{d}_{0:t}$ are the full histories of the pose of the robot and the dirt distribution starting from the beginning of the current cleaning cycle. Here, $\mathbf{u}_{1:t}$ are the motion controls of the robot, $\mathbf{z}^{\mathrm{r}}_{1:t}$ are the range measurements, and $\mathbf{z}^{\mathrm{d}}_{1:t}$ are the dirt measurements. We assume that while the robot cleans the environment, the values of the dirt distribution are only changed through the cleaning actions of the robot. Thus, the current dirt distribution $\mathbf{d}_t$ depends only on the previous dirt distribution $\mathbf{d}_{t-1}$ and the path taken by the cleaning robot. Similarly, the dirt measurement $\mathbf{z}^{\mathrm{d}}_t$ depends only on the change of the dirt distribution from time $t-1$ to time t. We describe the resulting dynamical system by the graphical model shown in Fig. 2.

Applying Bayes' rule and the rules of d-separation [1] to the graphical model yields the following factorization of Eq. (1):

$$\begin{aligned}
&p(\mathbf{x}_{0:t}, \mathbf{d}_{0:t} \mid \mathbf{u}_{1:t}, \mathbf{z}^{\mathrm{r}}_{1:t}, \mathbf{z}^{\mathrm{d}}_{1:t}) \\
=&\eta \ \underbrace{p(\mathbf{z}^{\mathrm{r}}_t \mid \mathbf{x}_t)}_{\text{range sensor model}} \ \underbrace{p(\mathbf{z}^{\mathrm{d}}_t \mid \mathbf{d}_{t-1}, \mathbf{d}_t)}_{\text{dirt sensor model}} \\
&\underbrace{p(\mathbf{d}_t \mid \mathbf{d}_{t-1}, \mathbf{x}_{t-1}, \mathbf{x}_t)}_{\text{dirt actuator model}} \underbrace{p(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \mathbf{u}_t)}_{\text{robot motion model}} \\
&\underbrace{p(\mathbf{x}_{0:t-1}, \mathbf{d}_{0:t-1} \mid \mathbf{u}_{1:t-1}, \mathbf{z}^{\mathrm{r}}_{1:t-1}, \mathbf{z}^{\mathrm{d}}_{1:t-1})}_{\text{prior distribution}},
\end{aligned} \tag{2}$$

where η is a normalization constant. Eq. (2) defines a recursive update rule, which allows to compute the desired distribution at time t from the distribution at time $t-1$,

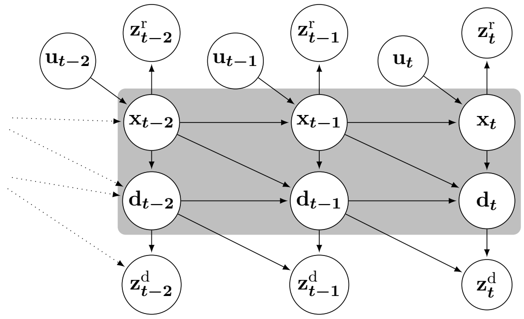

Fig. 2. Graphical model of the dynamical system describing the evolution of the robot pose $\mathbf{x}_t$ and the dirt distribution $\mathbf{d}_t$.

given all sensor and actuator models. For the range sensor and the robot motion, we employ standard models from the literature [17]. In the following, we introduce models for the dirt actuator and the dirt sensor.

B. Dirt Actuator Model

The dirt actuator model describes how the values in the dirt distribution change from one time step to the next, depending on the area covered by the cleaning unit, i.e., the path traveled by the robot. Every speck of dirt that the cleaning unit passes over is either sucked in or left on the floor. Every such event can be modeled by a Bernoulli distribution with a probability p for cleaning the speck of dirt. The parameter p is specific to the cleaning unit, type of dirt, and floor to be cleaned and has to be calibrated accordingly. To determine how many such events occur in each cell of the dirt distribution, we apply unweighted area sampling [5] to calculate the proportion $\mathcal{P}_i(\mathbf{x}_{t-1}, \mathbf{x}_t)$ of each cell c_i that the cleaning unit passed over (see Fig. 3 for an illustration). Under the assumption that inside each single cell, the dirt is distributed uniformly, multiplying $\mathcal{P}_i(\mathbf{x}_{t-1}, \mathbf{x}_t)$ with the amount of dirt $\mathbf{d}_{i,t-1}$ that was in the cell at time $t-1$ yields the number of Bernoulli events in this cell. As a series of independent Bernoulli events follows a Binomial distribution, the reduction of dirt $(\mathbf{d}_{i,t-1} - \mathbf{d}_{i,t})$ in cell c_i is distributed according to $\mathrm{Binom}(\mathbf{d}_{i,t-1} - \mathbf{d}_{i,t}; \mathcal{P}_i(\mathbf{x}_{t-1}, \mathbf{x}_t)\mathbf{d}_{i,t-1}, p)$, where $\mathrm{Binom}(k; n, p)$ is the probability of drawing sample k from a Binomial distribution with parameters n and p. Assuming that the amounts of dirt cleaned in different cells are independent, this leads to the dirt actuator model

$$\begin{aligned} & p(\mathbf{d}_t \mid \mathbf{d}_{t-1}, \mathbf{x}_{t-1}, \mathbf{x}_t) \\ &= \prod_{i=1}^{M} \mathrm{Binom}(\mathbf{d}_{i,t-1} - \mathbf{d}_{i,t}; \mathcal{P}_i(\mathbf{x}_{t-1}, \mathbf{x}_t)\mathbf{d}_{i,t-1}, p). \end{aligned} \tag{3}$$

C. Dirt Sensor Model

The dirt sensor model in Eq. (2) is a probabilistic model describing the measurement $\mathbf{z}^{\mathrm{d}}_t$ received if the dirt distribution changes from $\mathbf{d}_{t-1}$ to $\mathbf{d}_t$. The dirt sensor included in our robot, which is the iRobot Roomba, is a small metal plate inside the cleaning unit, which sends a signal if it gets hit by a speck of dirt during cleaning. As dirt measurement $\mathbf{z}^{\mathrm{d}}_t$, we use the number of these signals received between the time steps $t-1$ and t. The rate with which the metal plate gets hit in this time interval depends on the amount of dirt $(\mathbf{d}_{t-1} - \mathbf{d}_t)$ sucked in. Assuming that the individual hitting events occur independently of each other, this process can be described by a Poisson distribution with intensity parameter $\lambda = (\mathbf{d}_{t-1} - \mathbf{d}_t)$. Therefore, we define the dirt sensor model as

$$p(\mathbf{z}^{\mathrm{d}}_t \mid \mathbf{d}_{t-1}, \mathbf{d}_t) \sim \mathrm{Poiss}(\mathbf{d}_{t-1} - \mathbf{d}_t). \tag{4}$$

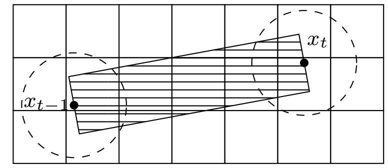

Fig. 3. Visualization of the area covered by the cleaning unit when the robot moves from pose x_{t-1} to pose x_t. The dotted circles mark the diameter of the robot. The width of the covered rectangle depends on the width of the cleaning unit. We use unweighted area sampling to estimate the covered percentage of each grid cell.

IV. A GUARANTEE FOR CLEANING WITH HIGH CONFIDENCE

The goal of our robotic cleaning approach is to guarantee that the maximum dirt value in the environment is reduced below the user-defined threshold τ with confidence $1-\delta$. In this section, we describe our implementation of the recursive update rule in Eq. (2). Furthermore, we show how we use the resulting estimated dirt values to adapt the future cleaning path to achieve the desired cleaning guarantee.

A. Particle Filter Implementation

To implement the recursive estimation scheme, we use a particle filter [17]. The particle filter represents the probability distribution in Eq. (2) as a set of K sample hypotheses, called particles. The particle set has the form

$$Y_t = \{(\mathbf{x}^{[k]}_{0:t}, \mathbf{d}^{[k]}_{1,0:t}, \ldots, \mathbf{d}^{[k]}_{M,0:t}) \mid k \in 1, \ldots, K\}, \tag{5}$$

where $\mathbf{x}^{[k]}_{0:t}$ is the hypothesis of the k-th particle about the history of the robot pose and $\mathbf{d}^{[k]}_{i,0:t}$ is its hypothesis about the history of dirt values of cell c_i. At each time step, we create the updated hypothesis of each particle by drawing a random sample first from the robot motion model, and then, given its outcome, from the dirt actuator model. Then, we assign to all K particles weights $w^{[k]}$ that describe the likelihood of the sensor readings given the sampled states. We compute the weights as the product of the densities of the dirt sensor model and the distance sensor model. Finally, in a *resampling step*, we create a new set of particles from the actual set by drawing K times with replacement from the probability distribution induced by the weights $w^{[k]}$.

Compared to the standard particle filter for robot localization, the extended particle state defined in Eq. (5), of course, has an increased dimensionality and thus an increased computational complexity, as it requires more particles. However, as changes in the dirt distribution happen only in parts

TABLE I

RESULTS OF THE REAL WORLD EXPERIMENTS.

	Successes	**Distance [m]**	**Time [s]**
Our approach	10/10	20.54	321.54
No replanning	4/10	10.98	138.51
Systematic	6/10	81.12	891.85
Roomba	0/10	-	321.54

in which particles are at the moment, we can reduce the computational complexity by sampling dirt values only from those parts. In this efficient approximation, we initialize the dirt distribution of all cells c_i in each particle k with the mean λ_0^i of the initial dirt distribution $\mathbf{d}_{0,i}$. Only if the robot pose hypothesis of a particle enters a cell c_i at some time step s, we sample a concrete instantiation of the dirt value of this cell from the Poisson distribution with parameter λ_{s-1}^i. This value is updated according to the dirt actuator model resulting in the sample $\mathbf{d}_{s,i}^{[k]}$. When, at some time step $t-1$, the pose hypotheses of all particles are outside of c_i again, we estimate the new mean λ_t^i of the Poisson distribution of c_i and set the values $\mathbf{d}_{t,i}^{[k]}$ of all particles to λ_t^i. As estimator, we use the weighted mean of the dirt values in all particles:

$$\lambda_t^i = \sum_{k=1}^{K} w^{[k]}\, \mathbf{d}_{t-1,i}^{[k]}. \tag{6}$$

Although some values $\mathbf{d}_{t-1,i}^{[k]}$ are not samples but contain the previously estimated mean, this is an unbiased estimator. Due to this efficient implementation, the computational complexity of our approach does not depend on the size of the environment but on the resolution of the dirt map. A finer resolution would require a increased number of particles. In the experiments we show that our approach runs online and effectively with a reasonable number of particles.

B. Localization-Based Replanning of the Cleaning Path

To achieve the desired cleaning guarantee, the robot starts cleaning by following the *bounded dirt time minimization* cleaning path introduced by Hess *et al.* [8]. This path is created from the policy $\pi(\mathbf{d}_0) = \{c_{i_1}, \ldots, c_{i_n}\}$ that consists of all cells in which the $(1-\delta)$ quantile of the dirt distribution is above τ. Given $\pi(\mathbf{d}_0)$, we generate the path by solving the traveling salesman problem (TSP) on the fully connected graph with all cells in $\pi(\mathbf{d}_0)$ and the actual robot position as vertices and with the Euclidean distances of the shortest collision-free paths between the cells as edge costs.

The contribution of our approach is that during cleaning, we re-compute the cleaning path based on the current dirt distribution to re-visit previously imperfectly cleaned cells. Concretely, at fixed intervals during operation, we consider the parameters λ_t^i of all cells c_i of the dirt distribution that are currently empty of particles. The currently traversed cells are not considered as their cleaning is in progress and their state might be changing. From the values λ_t^i, we calculate the $(1-\delta)$ quantiles of the dirt distributions in every cell. From the quantiles, we compute the cells that have to be cleaned, i.e., the policy $\pi_t(\hat{\mathbf{d}}_t)$ at time t, and compare it with the policy $\pi_{t-1}(\hat{\mathbf{d}}_{t-1})$ that is currently followed. If the new policy $\pi_t(\hat{\mathbf{d}}_t)$ includes cells that the policy $\pi_{t-1}(\hat{\mathbf{d}}_{t-1})$ would not visit anymore, we re-compute the TSP path using the previous path as an initial guess. The robot ends the cleaning cycle if, after a final re-planning step, the policy contains no cells anymore, i.e., the $(1-\delta)$ quantiles of the dirt distributions in every cell are below τ. Thus, after cleaning, our approach can guarantee that the dirt levels in the whole environment are below τ with confidence $1-\delta$.

V. EXPERIMENTAL EVALUATION

We evaluated the performance of our approach in extensive experiments both in simulation and with a real cleaning robot. In all experiments, we used a confidence level of $1-\delta = 99\%$, and a maximum allowed dirt level of $\tau = 5$. In preliminary experiments, we found that it suffices to set the number of particles to $K = 500$. To compute cleaning paths, we applied a state-of-the-art TSP solver that runs in approximately $\mathcal{O}(n^{2.2})$ [7].

A. Evaluation on a Real Cleaning Robot

To evaluate the effectiveness of our approach in practice, we used an iRobot Roomba 560 vacuuming cleaning robot, which is by default equipped with a dirt sensor. This dirt sensor is a small metal plate inside the suction unit that generates a measurement whenever a speck of dirt hits the plate. In the experiment, we used larger grained flour as dirt. A calibration of the sensor yielded that every measurement (and therefore also every unit of λ_t^i and of τ) corresponds to approximately $0.16\,\mathrm{g}$ of flour. We additionally equipped the robot with an Asus Xtion Pro Live sensor that we used for range measurements, with a gyroscope to improve the odometry readings of the robot, and with a notebook. The robot and the experimentation area are shown in the upper part of Fig. 1. The lower part of the figure shows the occupancy grid map and the initial dirt distribution used in the experiment. The resolution of the cells of the dirt distribution is $0.10\,\mathrm{m}$. For this experiment, we manually specified the initial values of the dirt distribution. Concretely, we selected 25 cells c_i (see Fig. 1), for which we specified a λ_0^i value of the Poisson distribution of 10, 15, 20, or 25. For all other cells of the dirt distribution, we set λ_0^i to zero. For each run of the robot, we sampled a concrete instantiation from the specified initial dirt distribution as ground truth. We performed ten cleaning cycles using our approach. To measure the amount of flour that we put into the environment and the amount that was still there after cleaning, we used a precision balance. The first row of Table I shows the number of successes, i.e., the number of runs after which the 99% quantile of ground truth dirt was not above τ. It also shows the average distance traveled per cleaning cycle as estimated by the particle filter, and the average travel time. As the table shows, our approach successfully met the threshold τ in every run, in fact even in every cell. During one run, it performed on average 22.7 replanning operations that actually changed the path traveled by the robot. The

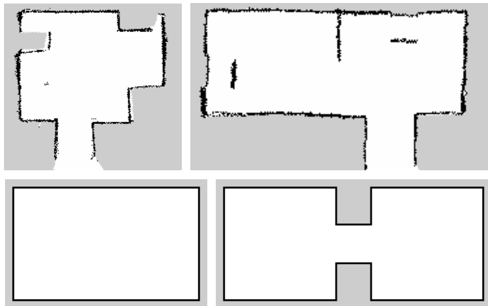

Fig. 4. Environments used for the simulation experiments. The maps in the upper row were recorded using the real robot.

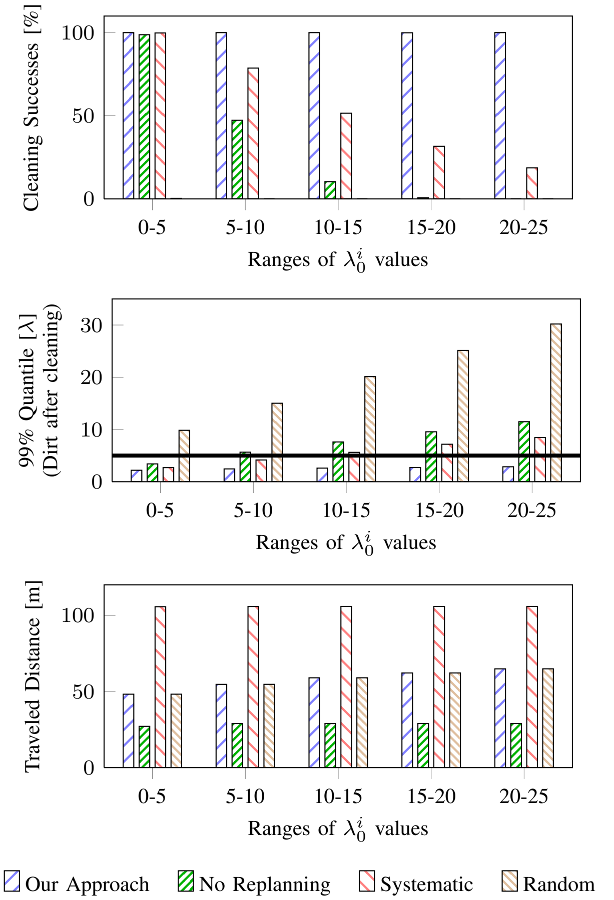

Fig. 5. Results of the simulation experiments averaged over all environments and percentages of dirty cells. The horizontal line in the middle plot marks the maximum allowed dirt level τ.

planning and re-planning times were below 0.3 seconds. For comparison, we also applied three other cleaning strategies to the same scenario. The three other strategies were the bounded-dirt strategy without replanning (*no replanning*) as described in our previous work [8], a *systematic* approach, and the built-in strategy of the *Roomba* robot. For the systematic approach, we let the robot execute a sweeping path that covers the whole environment and overlaps $0.025\,\mathrm{m}$ parallel to the direction of movement of the robot, to account for some of the localization error in the path execution. The Roomba strategy mostly performs randomized movements but also makes use of the dirt sensor. If the robot detects dirt, it performs rotational movements to clean more thoroughly. We started the Roomba strategy at the same pose as our approach and let it run for the same amount of time as our approach. Table I shows the results of the comparison. All other approaches did not meet the desired threshold τ in at least four out of ten runs. The *no replanning* strategy needed shorter times and distances than our approach, as it only visits each selected cell once. The systematic strategy needed more than double the time of our approach and did not succeed in four cases due to slight localization errors. The Roomba strategy always missed at least one dirty cell entirely in the time in which our approach cleaned the whole environment. For the Roomba strategy, no average distance is available, as our distance estimate is based on the particle filter data. As t-tests showed, the 99% quantile of the dirt values after cleaning with our approach is significantly lower than 99% quantiles for all other approaches on a 5% level. Also, our approach needs significantly shorter times and distances than the systematic strategy on a 5% level. Still, the results of this experiment raise additional questions: Does the structure of the environment influence the result? Is our approach still efficient if the dirt is more widely distributed? Does the incorporation of the dirt sensor measurements in the particle filter influence the localization performance? To answer these questions, we performed a number of simulation experiments.

B. Evaluation of the Cleaning Performance in Simulation

In the first set of simulation experiments, we evaluated the influence of the structure of the environment and of the amount of dirt in the environment on the performance of our approach. We applied our approach in four different environments, two maps obtained from real world robotic experiments and two artificial ones (see Fig. 4). In addition to the different environments, we evaluated three different percentages (5%, 10% and 20%) of dirty cells in the environments and five different ranges of λ_0^i values (5-10, 10-15, 15-20, 20-25, and 25-30) of expected dirt in the dirty cells. We sampled the dirty cells in the environment uniformly from the set of all cells of the dirt distribution, and the λ_0^i values of each dirty cell uniformly from the considered range of values. For the sensors and actuators of the simulated cleaning robot, we used the parameters obtained in the real world experiments. For comparison, we also applied the *no replanning* and the *systematic* approach described in the previous section. As the *Roomba* strategy was not available in simulation, we considered another *random* strategy. This strategy drives straight ahead until it hits a wall, randomly rotates on the spot, and drives forward again. In each run, we stopped the random strategy as soon as the robot traveled the same Euclidean distance as with our approach. For each strategy, environment, percentage of dirty cells, and range of dirt levels, we performed 100 trials. We evaluated the distance traveled by the robot and the 99% quantile of the ground truth dirt remaining after each run. We consider a run as successful if the 99% quantile is not above $\tau = 5$.

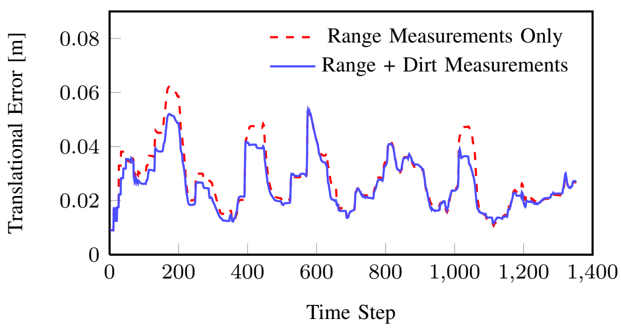

Fig. 6. Translational localization error with and without dirt measurements.

Fig. 5 shows the results of the simulation experiments averaged over all environments and percentages of dirty cells in the environments. The individual results for the different environments and percentages were qualitatively similar. Only the random strategy performed slightly better in the plain rectangular environment than in the others, but in general resulted in a very high value of remaining dirt. As the figure shows, our approach succeeded in every single trial and maintained its success rate for increasing values of dirt per cell, while the success rates of all other approaches decreased. For every range of λ_0^i values, t-tests revealed that the 99% quantiles of the dirt remaining in the environment after cleaning with our approach are significantly lower than the 99% quantiles for all other approaches on a 5% level.

As shown in Fig. 5, the *no replanning* approach yielded the shortest distance. As our approach re-visits previously imperfectly cleaned cells, it needs to travel a longer distance, which increases with the amount of dirt per cell. Still, compared to the systematic approach, t-tests showed that our approach resulted in significantly shorter traveled distances for every evaluated range of λ_0^i values on a 5% level. Note that in real world environments, the dirty cells are typically not uniformly distributed but clustered, e.g., around the dinner table or the entrance area. This would reduce the distance traveled by our approach even more, as the average distance between two dirty cells would be smaller.

C. Evaluation of the Localization Performance

In a second set of simulation experiments, we evaluated the influence of the dirt sensor measurements on the localization performance of the robot. We used the occupancy grid map and the dirt distribution from the real world experiments described in Sec. V-A and simulated 100 runs with the *no replanning* strategy. In each run, we applied a particle filter localization without integrating the dirt sensor measurements as well as the particle filter localization with the dirt sensor integration described in Sec. III-A. For each time step, we recorded the translational error between the ground truth and the most likely particle.

Fig. 6 shows the resulting errors averaged over all runs. Especially in parts of the trajectory with high localization errors, integrating the dirt measurements can substantially reduce the localization error. A t-test revealed that averaged over all time steps, our state estimation approach results in significantly smaller localization errors than the particle filter localization without dirt sensor measurements on a 5% level.

VI. Conclusions

In this paper, we presented a novel approach for jointly estimating the trajectory of the cleaning robot and the distribution of dirt in the environment. For this, we developed probabilistic models of the cleaning unit as well as the dirt sensor of the robot. We showed how to use our approach to adapt the cleaning path during operation, leading to guaranteed low dirt levels in the entire environment with high confidence. Extensive experiments, also with a real cleaning robot, demonstrated that our approach outperforms other approaches in terms of cleanliness after operation and, as a side effect, even improves the localization performance of the robot. One option for future research is to extend our approach to work with a representation of the dirt distribution that considers spatial dependencies.

References

[1] C. Bishop. *Pattern recognition and machine learning*. Springer, 2007.

[2] T. Bretl and S. Hutchinson. Robust coverage by a mobile robot of a planar workspace. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2013.

[3] H. Choset. Coverage for robotics – a survey of recent results. *Annals of Mathematics and Artificial Intelligence*, 31(1):113–126, 2001.

[4] B. Ferris, D. Fox, and Neil D Lawrence. Wifi-slam using gaussian process latent variable models. In *Proc. of the Intl. Conf. on Artificial Intelligence (IJCAI)*, 2007.

[5] J. Foley, A. Van Dam, S. Feiner, J. Hughes, and R. Phillips. *Introduction to computer graphics*. Addison-Wesley Reading, 1994.

[6] J. Gutmann, E. Eade, P. Fong, and M. Munich. A constant-time algorithm for vector field SLAM using an exactly sparse extended information filter. In *Proc. of Robotics: Science and Systems (RSS)*, 2010.

[7] K. Helsgaun. An effective implementation of the Lin-Kernighan traveling salesman heuristic. *European Journal of Operational Research*, 126:106–130, 2000.

[8] J. Hess, M. Beinhofer, D. Kuhner, P. Ruchti, and W. Burgard. Poisson-driven dirt maps for efficient robot cleaning. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2013.

[9] W. Jeong and K. Lee. CV-SLAM: A new ceiling vision-based SLAM technique. In *Proc. of Robotics: Science and Systems (RSS)*, 2005.

[10] C. Jung, W. Chung, J. Ahn, M. Kim, G. Shin, and S. Kwon. Optimal mechanism design of in-pipe cleaning robot. In *Proc. of the IEEE Intl. Conf. on Mechatronics and Automation (ICMA)*, 2011.

[11] Y. Katsuki, T. Ikeda, and M. Yamamoto. Development of a high efficiency and high reliable glass cleaning robot with a dirt detect sensor. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2011.

[12] G. Lawitzky. A navigation system for cleaning robots. *Autonomous Robots*, 9(3):255–260, 2000.

[13] J. Müller, O. Paul, and W. Burgard. Probabilistic velocity estimation for autonomous miniature airships using thermal air flow sensors. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2012.

[14] M. Pearson, C. Fox, J. Sullivan, T. Prescott, T. Pipe, and B. Mitchinson. Simultaneous localisation and mapping on a multi-degree of freedom biomimetic whiskered robot. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2013.

[15] E. Prassler, A. Ritter, C. Schaeffer, and P. Fiorini. A short history of cleaning robots. *Autonomous Robots*, 9(3):211–226, 2000.

[16] C. Stachniss and W. Burgard. Mobile robot mapping and localization in non-static environments. In *Proc. of the AAAI Conf. on Artificial Intelligence (AAAI)*, 2005.

[17] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics*. MIT Press, 2006.

[18] Y. Zhang, J. Liu, G. Hoffmann, M. Quilling, K. Payne, P. Bose, and A. Zimdars. Real-time indoor mapping for mobile robots with limited sensing. In *Proc. of the 3rd Intl. Workshop on Mobile Entity Localization and Tracking*, 2010.