

Heterogeneous Feature State Estimation with Rao-Blackwellized Particle Filters

Gian Diego Tipaldi, Alessandro Farinelli, Luca Iocchi, Daniele Nardi
Dipartimento di Informatica e Sistemistica
University of Rome "La Sapienza"
Via Salaria 113, I-00198 Rome, Italy
E-mail {lastname}@dis.uniroma1.it

Abstract—In this paper we present a novel technique to estimate the state of heterogeneous features from inaccurate sensors. The proposed approach exploits the reliability of the feature extraction process in the sensor model and uses a Rao-Blackwellized particle filter to address the data association problem. Experimental results show that the use of reliability improves performance by allowing the approach to perform better data association among detected features.

Moreover, the method has been tested on a real robot during an exploration task in a non-planar environment. This last experiment shows an improvement in correctly detecting and classifying interesting features for navigation purpose.

I. INTRODUCTION

Mobile robots have been applied in domains such as space exploration [1], surveillance [2], [3] and search and rescue [4]. A primary requisite for such robot is to provide an estimate of its state (e.g., position, velocity, etc.) along with a representation of its surrounding environment (e.g., a map). Localization and Mapping are two very well known areas in the field of mobile robots, and several successful approaches have been recently proposed (e.g., [5], [6]).

Most of these approaches are very well suited for structured, planar environments; however, increasing interest is posed towards environments where a plain planar representation is not sufficient: such as outdoor navigation or search and rescue. In such domains, robotic systems deal with complex 3D objects (such as slopes, ramps or stairs) that can be detected and identified only through sophisticated techniques and algorithms. Detection of such complex features is relevant both to low level primary tasks, such as navigation, and to higher level complex tasks, such as surveillance, search and rescue, etc.

When such complex features are involved in the estimation process, the reliability of the feature extraction process cannot be taken for granted. A wrong feature classification, might lead the robotic platform to a completely wrong and possibly dangerous maneuver. Consequently, a correct estimation of detected features from the observations acquired over time is of crucial importance. For example, consider a robot which is performing an exploration task in a complex unstructured environment.

Previous works in the field of localization and mapping do not usually explicitly consider the possibility of false measurements due to sensor inaccuracy. Those works rather process all the measures that the robot receives from its sensors using a probabilistic error model, and filter out false readings through iterative filtering techniques (Kalman Filters, Particle Filters, etc.). This kind of approach is mainly

justified by the high accuracy of currently used sensing devices which are deployed for this applications (e.g., laser range finders).

In this work we propose a method to estimate the state of a set of complex heterogeneous features considering the reliability of the feature extraction process involved. The features we are considering may belong to different classes and can be represented in different ways. Features can be extracted by different processing routines, possibly returning only partial observations about the feature state. Our aim is not to provide an approach for Localization and Mapping, but rather to locate the heterogeneous features extracted by different dedicated routines inside the world representation built by the robot.

We suppose that the routine that detects features is able to provide a reliability measure of the classification performed. Such reliability measure is used as an approximate prior over the current association. In this way we are able to explicitly consider possible failures of the low level detection routine such as false positives and false negatives. Moreover, we deal with both the two cases arising from knowing or not in advance the number of features that are in the environment.

The main ideas presented in this paper are: i) the use of reliability of the extraction process in the sensor model; ii) a formal framework for data association using sensor reliability, based on Rao-Blackwellized particle filters.

The paper is organized as follow. Related work is presented in Section II. Section III describes the formalization of the problem, Section IV the Rao-Blackwellized particle filter approach, while Section V presents our approach both when the number of features is known a priori and when such number has to be estimated on-line. Section VI shows the experimental results obtained. Finally, section VII concludes the paper.

II. RELATED WORK

The problem of state estimation is related to the problem of Localization, Mapping and Simultaneous Localization and Mapping (SLAM) [7]. Also several works in the SLAM area uses Rao-Blackwellized particle filters [8], [9]. However, in this paper we present an approach to address a different problem, we want to estimate the state of interesting features in the environment on top of two other modules present in the system: a SLAM algorithm for mapping and localization [10] and a feature detection process (implemented as a neural network classifier).

Works in the area of Multi-Object Tracking (MOT) are also related to our work [11]. In particular Rao-Blackwellized particle filters have been used in [12] to track multiple objects. The use of particle filtering technique for MOT is justified by the asymptotic convergence of particle filtering to the optimal Bayesian estimator. Moreover, the use of Rao-Blackwellized particle filters provides an efficient approach. In addition to this, in our work we use in the proposal distribution the reliability of the feature detection process, and model the association evolution using a Dirichlet model. In this way we are able to have a more efficient estimation process, as shown by the experimental results.

In the work [13] the authors present an approach based on Markov Chain Monte Carlo Data Association to track multiple objects. The authors prove that when the number of features in the environment is known a priori their approach ensure convergence in polynomial time and with a single scan processing. However, when the number of features is unknown their approach cannot be considered an incremental approach, and this is a fundamental requirement for our application.

In [14] a similar approach is presented. The authors use a Rao-Blackwellized Particle Filter to estimate people location in the environment. Each particle represents the association history between Kalman filtered object tracks and observations. While the authors use a very similar technique to our work, there are two important differences. First of all, their approach is designed for tracking purposes, while our presents a general framework for data association. Secondly, we also use the sensor reliability within the framework, while they just rely on the observation likelihood.

III. PROBLEM FORMULATION

Let $F = \{f^1, \dots, f^K\}$ be a set of K features which are present in the environment, where K may be either known or not (see section V for further details on this). Each feature has associated a state x_t^k , which can evolve during time. The nature of the state is strictly connected with the class of the corresponding feature. Therefore, such states neither belongs to the same space nor have the same dimension. The state evolution of each feature is modeled by a stochastic process ruled by the distribution:

$$p(x_{t+1}^k | x_t^k) \quad (1)$$

The features persist over time. The robot is able to observe a subset of the features at time. Those features are detected by a complex algorithm, and a reliability measure about the detection is provided¹. The number of features is modelled as a Poisson distribution, with parameter $\lambda_n V$, where V is the dimension of the explored space and λ_n is the scattered rate of the features. There are also false alarms and the number of false alarms also follows a Poisson distribution with parameter λ_f , where λ_f is the false alarm rate.

Without loss of generality, we restrict ourselves to consider just one observation for time step. Let $z_t = \langle y_t, Rel(y_t) \rangle$ the reading at time t . This reading is composed by two parts: y_t is an observation of the feature²; $Rel(y_t)$ is a vector of

¹The reliability is intended as a measure on how likely the feature is detected and associated to an already existing one

²More precisely, y_t is a raw data, which is transformed into a feature observation according to that feature's class

numerical values that encodes the reliability of the detection process for each feature. For the sake of clarity, we use the notation $Rel^k(y_t)$, $k = 1, \dots, K$ to denote the probability of the readings being generated by the k -th feature, $Rel^0(y_t^j)$ to denote the probability of the reading being a false alarm and $Rel^{K+1}(y_t^j)$ to denote the probability of the readings being generated by a new feature. It is worth noting, that $Rel^0(y_t^j)$ and $Rel^{K+1}(y_t^j)$ do not depend on the detection process, but are instead modelled by the Poisson distributions defined above.

The observations y_t are generated from

$$\begin{cases} p(\Psi^k(y_t) | x_t^k) & \text{if the observation is from } x_t^k \\ u_t & \text{otherwise} \end{cases} \quad (2)$$

where $p(\Psi^k(y_t) | x_t^k)$ is the likelihood of the state given the observation, Ψ^k is a function that maps the raw data y_t into the measurement space of the k -th feature and $u_t \sim \text{Unif}(\mathcal{R})$ is a random process for false alarm, \mathcal{R} being the domain of y_t . This likelihood depends on the class associated to the feature and in general is different among the features. In the rest of the paper, we will omit writing the function Ψ for notational simplicity.

Association among observations and features is obtained through a joint association event [13] $\alpha_t = \{T_t^0, T_t^1, \dots, T_t^K\}$, where T_t^k is the set of observations associated to the feature k . At every time step, the system associates an observation z_t with a feature: for instance, if z_t is associated with the k -th feature, then the correspondent track $T_t^k = T_{t-1}^k \cup \{z_t\}$ will be updated. Therefore the goal of the system is to provide $\mathbb{E}(x_t^k | z_{1:t})$, which is dependent on $P(\alpha_t | z_{1:t})$.

In the following, for notational simplicity, we will make use of the term label to address the association between observations and features. We map an association event into a sequence of labels $L_{1:t} = \{L_1, \dots, L_t\}$ where $L_i = k$ if the observation i is associated to the feature k . Using this notation, we can derive the incremental update equation of the filter.

IV. RAO-BLACKWELLIZED PARTICLE FILTERS

The previous problem can be well described within the Bayesian framework, resulting in estimating the joint posterior over the associations and features' state, $p(X_{1:t}, L_{1:t} | z_{1:t})$, given the history of observations. Once obtained this distribution, it is straightforward to obtain $p(X_t | z_{1:t})$. One way is to simply marginalize out the association. This is similar to a Joint Probabilistic Data Association Filter, which simulates an exact Bayes filter and compute the expected posterior by just marginalizing out the last association made [11]. A second way is to obtain $p(X_t | z_{1:t}, \hat{L}_{1:t})$, being $\hat{L}_{1:t}$ a Maximum A Posteriori estimator for $p(L_{1:t} | z_{1:t})$. This approach is in a way similar to the Multiple Hypothesis Tracker; the main difference lies in the fact that RBPF is a fully probabilistic approach, while MHT uses some deterministic rules to add and delete association tracks.

Exact inference in such a model is not possible, since the number of possible associations grows hyper-exponentially with the number of measurements. As a matter of fact, this

number is equal to the number of set partitions of the set of measurements, which is known as the *Bell number* [15].

However, only a small fraction of the set partitions is feasible with respect to the real associations. Based on this observation, it is reasonable, both from a theoretical and practical perspective, to approximate the distribution of the associations with a sum of different samples. The reason behind this is that the mass of the distribution is concentrated only in a small portion of the state space. In other words, we want to focus our attention on the most probable associations, which are represented by the samples drawn from the association distribution, by using Monte Carlo methods [16].

The key idea of the Rao-Blackwellized particle filter for this problem is to estimate a posterior $p(L_{1:t}|z_{1:t})$ about potential labels $L_{1:t}$ of the measurements given the observations $z_{1:t}$, and to use this posterior to compute a posterior over features' state and feature association.

$$p(X_{1:t}, L_{1:t}|z_{1:t}) = p(X_{1:t} | L_{1:t}, z_{1:t})p(L_{1:t} | z_{1:t}) \quad (3)$$

This can be done efficiently, when the posterior over features' state $p(X_{1:t} | L_{1:t}, z_{1:t})$ can be computed analytically given the knowledge of $L_{1:t}$ and $z_{1:t}$. Luckily, there are several situations where this assumption holds (like for Linear Gaussian or Hidden Markov models). This technique is known as Rao-Blackwellized Particle Filter and is proved to reduce the variance of the estimate, according to the Rao-Blackwell Theorem [17].

In the next section we will show how to use the Rao Blackwellized Particle Filters in this context, by providing the proposal distribution and the weight computation, first when the number of features is fixed and known, then when their number is not known in advance and can increase over time.

V. STATE ESTIMATION WITH RBPF

In the previous section, we described the general framework of Rao Blackwellized Particle Filter. Here we show how to instantiate the framework in the context of heterogeneous features' state estimation. To this end, we need to define what information the samples (particles) represent and the nature of the distributions involved.

Each sample is characterized by the following $N+2$ -uple:

$$s_t^{(i)} = \left\langle w_t^{(i)}, L_{1:t}^{(i)}, \theta_t^{1,(i)}, \theta_t^{2,(i)}, \dots, \theta_t^{N,(i)} \right\rangle \quad (4)$$

where $w_t^{(i)}$ represents the importance weight, $L_{1:t}^{(i)}$ the association history and each $\theta_t^{n,(i)}$ the sufficient statistic of the corresponding feature, e.g. mean and variance if they are represented as Gaussians.

Recalling that the association is independent from the past reliability given the last one³, we can write the distribution in the following recursive way

$$p(L_{1:t}|z_{1:t}) \propto p(y_t|L_{1:t}, y_{1:t-1}, Rel(y_{1:t})) \cdot p(L_{1:t}|Rel(y_{1:t}), y_{1:t-1}) \quad (5)$$

$$= p(y_t|L_{1:t}, y_{1:t-1})p(L_t|Rel(y_t), L_{1:t-1}) \cdot p(L_{1:t-1}|z_{1:t-1}) \quad (6)$$

³The past reliabilities are embedded in the previous labels, thus resulting in this independence property

We now have a recursive distribution, which is well suited to be implemented in the Sequential Monte Carlo framework. We choose as proposal the distribution:

$$p(L_t|Rel(y_t), L_{1:t-1}) \quad (7)$$

While this is not the optimal proposal, as we do not use the measurements y_t , the reliability allows us to highly reduce the association space. This is because the reliability can be seen as a direct observation of the label, as it reflects the information provided by the detection algorithm. Experimental results show the differences between using or not such information (see section VI)

After the sampling stage, we apply the importance sampling principle, obtaining the following weight distribution:

$$w_t = w_{t-1}p(y_t|L_{1:t}, y_{1:t-1}) \quad (8)$$

which is easy to evaluate given the fact that we have already computed the predicted state distribution $p(X_t|L_{1:t}, y_{1:t})$ using the sufficient statistics embedded within the particles.

In the end, we adopt an adaptive resampling schema, by considering the effective number of particles [18]. This number,

$$N_{eff}(t) = \frac{1}{\sum_{i=1}^N (w_t^{(i)})^2} \quad (9)$$

is an approximative measure which tells us how well the set of samples approximates the goal distribution and is strongly related to the variance in the particle weights. We perform a resampling step when this quantity falls below a certain threshold, $N_{eff} < N_{tsh}$.

A. Fixed and known number of features

When the number of features to estimate is fixed and known, the problem is slightly simpler, as the association is constrained to one of the existing classes.

With respect to the proposal distribution, we have two important distributions to take into account. The first is given by the frequency of the different features in the environment. The second is given by the reliability and reflects our degree of belief about the association provided. We can use the fact that in a fixed and known number of features, the new label L_t is given by the product $p(Rel(y_t)|L_t)p(L_t|L_{1:t-1})$ as the reliability does not depend on the previous labels. We can obtain $p(L_t|L_{1:t-1})$ by marginalizing the multinomial distribution over the past associations. The parameters of this distribution have to be estimated, in order to reflect the real labels' frequency. To do so, we use a MAP estimate, using the Dirichlet distribution as a conjugate prior.

As for the weight, we need to define the conditional likelihood distribution, which strongly depends on the label associated to the reading. We can define the weight, by marginalizing with respect to the estimated state of the feature, obtaining:

$$p(y_t|L_{1:t}, y_{1:t-1}) = \int p(y_t|x_t^{L_t})p(x_t^{L_t}|T_t^{L_t})dx_t^{L_t} \quad (10)$$

This integral can be computed analytically for some distribution, such as Gaussians and discrete distributions. Otherwise, numerical or stochastic methods can be used to obtain a close approximation.

B. Variable and unknown number of features

The previous framework can be extended to handle an unknown number of features. First of all, we notice that the main difference in dealing with a known or unknown number of features lies in the sampling procedure, namely in the distribution $p(L_t|Rel(y_t), L_{1:t-1})$. In this scenario, we need to focus also on the fact that observations can come from an unknown feature, and model this fact appropriately. We use the Dirichlet Process, in order to estimate the probability of having a new class and assigning the observation to already existing classes. This distribution is often used in the infinite mixture model, and is considered a natural extension of the standard Dirichlet distribution when the number of classes is not known in advance[19], [20].

Using the Dirichlet Process as our prior over the parameters of $p(L_t|L_{1:t-1})$, and the independence of the reliability of time t from previous labeling, we can still reduce ourselves to:

$$p(L_t|Rel(y_t), L_{1:t-1}) = p(Rel(y_t)|L_t)p(L_t|L_{1:t-1}) \quad (11)$$

when:

$$p(L_t|L_{1:t-1}) = \begin{cases} \frac{|T_{t-1}^i|}{|L_{1:t-1}|+c} & i = 1, \dots, K \\ \frac{c}{|L_{1:t-1}|+c} & i = K + 1 \end{cases} \quad (12)$$

with c encoding our belief on the number of features in the environment.

The importance weight defined for the fixed case, (10), is still valid in this situations We just want to notice that in the case of the new feature, the integral reduces to the computation of the normalizing factor of the distribution (as the state is equal to the observation due to the initialization).

VI. EXPERIMENTS

We performed extensive experimental analysis of the approach in an abstract simulation and validation on data extracted from a mobile platform. The experiments are described in details in the following two sections. We first analyze some numerical results of the algorithm by implementing an abstract simulator. Then, we use our algorithm with data obtained from a real robot, in order to show applicability of this technique in real scenarios.

A. Abstract Simulator

We use an abstract simulator to perform an extensive quantitative analysis of correctness and completeness of the association algorithm. We use a Markov Chain to simulate the observations arrival. This chain is tuned in order to simulate a robot path inside an environment. This is achieved by giving higher probability to persistent move in the chain and it results in a burst of observations of one feature, followed by a burst of another feature and so on. The state of the features is given by a simple one dimensional Gaussian, as we decided to focus our attention on the association process, rather than on the estimation one. The observation of the state are sampled from a Gaussian distribution with different values for the variance. The reliability values for the feature association are sampled from a Dirichlet distribution, whose parameters are tuned in order to obtain slightly higher values of reliability for the correct association, simulating a real detection algorithm.

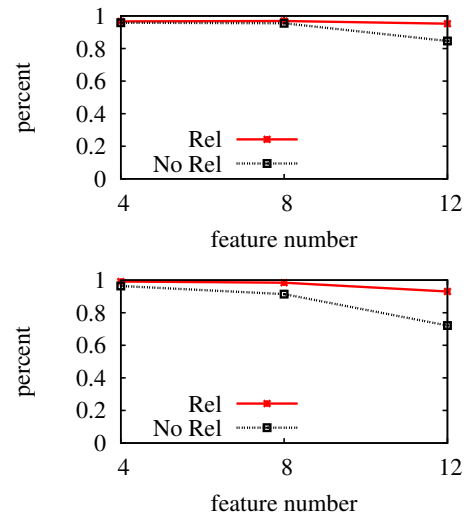


Fig. 1. Number of features: The graphs show the variation of the two metrics with respect to the number of features and the use of reliability. We show the results regarding correctness (top) and completeness (bottom)

We performed several experiments, varying the number of features to be tracked, the peakness of the Dirichlet distribution and the variance of the observations. In all those cases, we compute two measures, in order to evaluate the performances.

The first metric is related to the correctness of the association, and is given by:

$$\frac{TruePositive}{TruePositive + FalsePositive} \quad (13)$$

This value measures the percentage of correct associations made by the algorithm, with respect to the complete set of associations provided.

The second metric is related to the completeness of the association, and is given by:

$$\frac{TruePositive}{TruePositive + FalseNegative} \quad (14)$$

This value measures the percentage of correct associations made by the algorithm, with respect to the complete set of correct associations.

In order to prove the efficacy of using the reliability, we also performed the same experiments without using the reliability in the data association process, notice that, such an approach is similar to the one presented in [12]. We use the same parameters and the input data for both the approaches. The data set was composed of 700 simulated observations, with 5 runs with different random seeds. The number of particles used in the experiments was 1000, which allows for real time execution. The features are scattered within the environment at about 10m of distance.

In the first experiment, we analyze the variation of the performances when varying the number of features. When the number of features increases the performance of both the algorithms decreases, as one would expected. However, this decrease in performance is much more evident when the reliability is not taken into account, as can be seen in

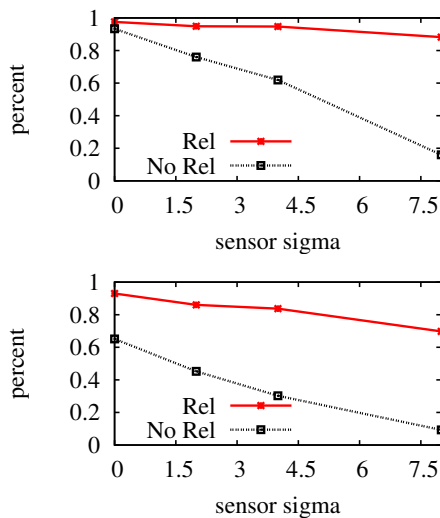


Fig. 2. Precision of the sensor: The graphs show the variation of the two metrics with respect to the precision of the sensor and the use of reliability. We show the results regarding correctness (top) and completeness (bottom)

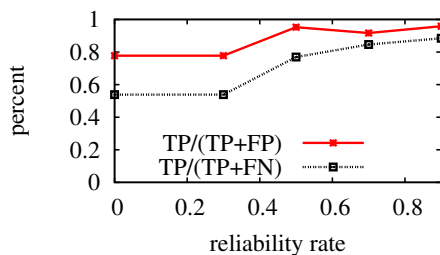


Fig. 3. Reliability: The graphs show the variation of the two metrics with respect to the peakiness of the reliability. Notice that a reliability of 0 means not using the reliability.

Figure 1. This is due to the increased combinatorial space when more features are presented. The reliability provides us some kind of "hint" about the right associations, which reduces the search space.

In the second experiment, we vary the precision of the sensor. The sensor is related to the analytical part of the features' state and its variance influences the importance weight computation. Figure 2 shows the result. As we can see, less precision in the sensor has some impact on the precision of the algorithm. This is due to the fact that when features are too close in the state space, there is not so much difference in the importance weight of a wrong association. As before the results show that even if there is a decrease in performance, such decrease is much more evident when the reliability is not taken into account.

Finally, we performed some experiments to show how the performance varies according to different levels of reliability. In Figure 3 the results are plotted. As one would expected, the performance increases when the reliability grows. In the experiments we let this value to vary from 0.4 percentage of getting the real association up to a 0.9 one. When the reliability value is 0 reliability is not used in the algorithm.

Further experiments with high dimensional features results

in similar classification rate, showing good robustness of the association with respect to the feature structure.

B. Real Robot

We performed a set of experiments on data extracted by a mobile platform performing an exploration task in a non-planar environment.

We focus on features that might be of interest for the navigation process. In particular, in the experiments we detect and estimate ramps and slopes present in the environment. A feature's state is composed by the feature class and the feature absolute position inside the environment.

A stereo system mounted on the robot is used to collect 3D information about these features. In particular, we compute a disparity map between the left and right image of the camera⁴ and use such a map for classifying three kinds of features: slopes, planes and ramps. To do so, we pre-process the disparity map extracting a vector of disparity values along the main direction of the view and then trained a Neural Network classifier with these data. The Neural Network was composed of one hidden layer with ten neurons, one input layer of ten neurons and one output layer of three and was trained with a standard backpropagation algorithm.

After training, the Neural Network was used to classify the three classes of features (slopes, ramps and planes) that are of interest for our application. Notice that, although for navigation purposes only ramps and slopes are considered, planes are still classified as features by the Neural Network and sometimes misclassifications of such a feature give rise to false positives.

We estimate the position of each feature using the position of the robot provided by a SLAM algorithm [10], applied during robot navigation in the environment, adding the distance given by the disparity map.

During the experiments, the robot performed a predefined path several times acquiring data for the interesting features along the way. We compared three approaches: i) our approach using reliability; ii) our approach without using reliability; iii) a nearest neighbor approach. The last method is a well known, simple method for data association that gives us a measure of the difficulty of the problem we are trying to solve. The same acquired data are given as input to all the approaches. In the environment there were four interesting features, highlighted by boxes in the map (see Figure 4). The rightmost feature is a *ramp* heading towards the wall, the leftmost feature is a *slope*, while the features in the middle are a *ramp* and a *slope* (going from bottom-left to top-right). Notice that, these last features will be interpreted as slopes or ramps depending on the heading of the robot. In this environment, a desirable behavior for an estimation algorithm would be to report one feature for each of the elements present in the environment (i.e., 3 ramps and 3 slopes in this case).

In Figure 4 we report the features detected by our algorithm (boxes) and the features detected by a simple nearest neighbour approach (crosses). As it is possible to see from this figure, the nearest neighbour approach reports several

⁴To compute the disparity image we used the Small Vision System software <http://www.ai.sri.com/software/SVS>

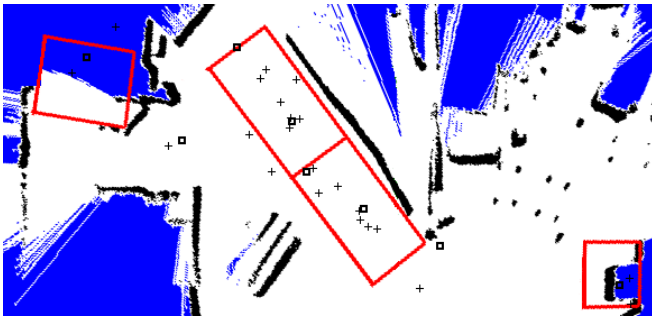


Fig. 4. Real robot experiments: results obtained using a simple nearest neighbor approach (crosses) and our current work (boxes).

incorrect associations, showing that the problem addressed is not a trivial one.

Compared to the nearest neighbour, our approach reports fewer misclassification and obtains a better accuracy of the feature state estimation. Similar results are obtained comparing to an approach similar to [12] (e.g., without the use of reliability) confirming the results obtained in simulation.

VII. CONCLUSIONS AND FUTURE WORK

In this paper we have presented an approach to estimate the state of heterogeneous features from inaccurate sensors. Two main novel aspects are present in the approach: the use of a Rao-Blackwellized particle filter to address the data association problem and the integration of the reliability of the feature extraction process in the sensor model. The use of a Rao-Blackwellized particle filters provides an efficient way to approximate the optimal Bayesian estimation of the features' states. Moreover, by integrating the reliability information in the estimation process our approach is able to consistently improve the performance of the system, providing a better estimate of the features' states.

We have performed several experiments on an abstract simulator as well as on a mobile robot. The abstract simulator allowed us to perform a quantitative evaluation over broad set of parameters, confirming that the use of reliability can improve the performance of the estimation process.

To validate the results obtained in the simulated environment, the method has been applied to a mobile robot exploration task, showing improvements in correctly detecting and classifying 3D features that are of interest for navigating in a non-planar environment.

In the future, we want to consider highly heterogeneous features, where the state space is not just a position and a class, but, for example, the slope and the length of a ramp, or the number of steps of a staircase, etc.. Moreover, it is interesting to validate our approach in situations where simple strategies, like the nearest neighbor, would not be applicable as it is not possible to define a distance among different features.

Another important aspect we want to further investigate is the use of different detection algorithms, for instance from different sensors, and analyze how different detectors influence the total reliability and how it is possible to use one detector to improve the performance of the others in terms of detection accuracy.

VIII. ACKNOWLEDGMENTS

Alessandro Farinelli is supported by the European Office of Aerospace Research and Development under grant number 053015. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the European Office of Aerospace Research and Development.

REFERENCES

- [1] D. Goldberg, V. Csicirello, M. B. Dias, R. Simmons, S. Smith, T. Smith, and A. Stentz, "A distributed layered architecture for mobile robot coordination: Application to space exploration," in *Proceedings of the 3rd Int. NASA Workshop on Planning and Scheduling for Space*, 2002.
- [2] L. E. Parker, "The effect of heterogeneity in teams of 100+ mobile robots," in *Proc. of the NRL Workshop on Multi-Robot Systems*, A. C. Shultz and L. E. Parker, Eds., vol. II. Washington, DC: Kluwer Academic Publishers, 2003, pp. 205–215.
- [3] K. e. a. Konolige, "Centibots: Large scale robot teams," in *Proc. of the NRL Workshop on Multi-Robot Systems*, A. C. Shultz and L. E. Parker, Eds., vol. II. Washington, DC: Kluwer Academic Publishers, 2003, pp. 193–204.
- [4] R. Murphy, J. G. Blitch, and J. L. Casper, "RoboCup/AAAI urban search and rescue events: Reality and competition," *AI Magazine*, vol. 1, no. 23, pp. 37–42, 2002.
- [5] W. Burgard, D. Fox, and S. Thrun, "Robust monte carlo localization for mobile robots," *Journal of Artificial Intelligence*, 2001.
- [6] J. Leonard and H. Durrant-White, "Mobile robot localization by tracking geometric beacons," *IEEE Transactions on Robotics and Automation*, vol. 7, pp. 376–382, 1991.
- [7] W. Burgard, D. Fox, and T. S., "A probabilistic approach to concurrent mapping and localization," *Machine Learning*, 1998.
- [8] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit, "FastSLAM: A factored solution to the simultaneous localization and mapping problem," in *Proceedings of the AAAI National Conference on Artificial Intelligence*. Edmonton, Canada: AAAI, 2002.
- [9] G. Grisetti, C. Stachniss, and W. Burgard, "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)*, 2005.
- [10] G. Grisetti, G. D. Tipaldi, C. Stachniss, W. Burgard, and D. Nardi, "Speeding up rao blackwellized slam," in *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, Orlando, FL, USA, 2006, pp. 442–447.
- [11] Y. B. Shalom and T. E. Fortmann, *Tracking and Data Association*. Boston: Academic-Press, 1988.
- [12] S. Sarkka, A. Vehtari, and J. Lampinen, "Rao-blackwellized monte carlo data association for multiple target tracking," in *Proc. of 7th Int. Conf. on Information Fusion*, vol. I, 2004, pp. 583–590.
- [13] S. Oh, S. Russell, and S. Sastry, "Markov chain monte carlo data association for general multiple target tracking problems," in *Proc. of 43rd IEEE Conference on Decision and Control (CDC)*, 2004.
- [14] D. Schulz, D. Fox, and J. Hightower, "People tracking with anonymous and id-sensors using rao-blackwellized particle filters," in *Proc. of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2003.
- [15] A. Nijenhuis and H. Wilf, *Combinatorial Algorithms*. Academic Press, 1978.
- [16] A. Doucet, N. De Freitas, and N. Gordon, *Sequential Monte Carlo Methods in Practice*. Springer Verlag, 2001.
- [17] A. Doucet, N. Freitas, K. Murphy, and S. Russel, "Rao-blackwellized particle filtering for dynamic bayesian networks," in *Proceedings of the 16th Annual Conference on Uncertainty in Artificial Intelligence (UAI 2000)*, 2000.
- [18] J. Liu, "Metropolized independent sampling with comparisons to rejection sampling and importance sampling," *Statist. Comput.*, vol. 6, pp. 113–119, 1996.
- [19] A. Ranganathan and F. Dellaert, "A rao-blackwellized particle filter for topological mapping," in *Int. Conf. on Robotics and Automation*, 2006.
- [20] D. Blackwell and J. B. MacQueen, "Ferguson distributions via poly urn schemes," *Annals of Statistics*, vol. 1, pp. 353–355, 1973.