Particle Filtering Approach for Data Association

Zoran Sjanic[*,†]
[*]Saab AB
[†]Division of Automatic Control
[†]Department of Electrical Engineering
[†]Linköping University
Linköping, Sweden
Email: zoran.sjanic@saabgroup.com

***Abstract*—An initial work has been performed to implement a sequential Monte Carlo method to solve the data association problem. The main motivation is to overcome the incorrect association when the state estimates are inaccurate. The solution is based on modeling the data association as a stochastic variable and estimated with a bootstrap particle filter. Two variants of the proposal function are evaluated, one with the uniform distribution over possible associations, and the other one with the distribution depending on the measurements and state estimates. The performance of both proposals is evaluated on the small simulation example, and compared to a purely deterministic approach, Nearest-Neighbour, as well. The obtained initial results are quite promising, and more evaluation and expansion to more examples and real data sets is suggested for the future work.**

***Index Terms*—Data Association, Particle Filter, Sequential Monte Carlo**

I. Introduction

In this work a look at a method to solve the data association problem based on particle filter has been carried out. Here data association is used as a term to define the assignment of sensor measurements of a set of objects to the actual objects, [1]–[4]. Examples include measurements of a static environment obtained from a moving platform and target tracking where the targets are moving and the measuring platform can be moving or static. It is also assumed that the environment or targets can be represented as a set of points (called landmarks or point targets depending on the application). This kind of environment or target representation is common in many applications, for example Simultaneous Localization and Mapping (SLAM) using camera as a sensor [5]–[7], structure from motion [8]–[10], bundle adjustment [11], [12], visual odometry [13], [14] or target tracking [4]. Normally, the assignment is performed by some method, usually deterministic, for example nearest-neighbour algorithm [15], auction algorithm [16], Hungarian algorithm [17] or Munkres algorithm [18]. However, in many cases the correct data association can be hard to obtain and/or computationally demanding, for example if the number of association variables or landmarks or targets is large or if there are errors in the involved variables. These errors can be platform's position and orientation estimation errors, as well as measurement noise, which is always the case in practice. In order to try to solve the association problem with the difficulties described above a sequential Monte Carlo filtering approach is proposed.

This work has been supported by the Industry Excellence Center LINK-SIC founded by The Swedish Governmental Agency for Innovation Systems (VINNOVA) and Saab AB.

The main idea here is to view data association as a stochastic variable that can be estimated by a Bayesian filter. The assumption is that the resulting filtered posterior will contain the correct association despite the errors in the involved variables. In the case of data association, however, the exact filtering is not possible due to the intractability of solving the needed calculations with the involved probability functions. This is due to the size of the problem, i.e., exponential growth of the possible assignment combinations. To overcome this, a sequential Monte Carlo approximation to the optimal Bayesian filtering is implemented in form of a bootstrap particle filter [19]. Although this idea of using particle filters in conjunction with data association is not new, see e.g., [20]–[25], there are some significant differences. In the previous work, the whole estimation problem, together with the data association, is solved with the particle filter, making the data association implicit and incorporated in the particle filter algorithm. In this work, a slightly different method is used. The platform's estimated states, like position and orientation, as well as landmarks are assumed to be given and the filter is only applied to the data association. This kind of approach is applicable for the estimation methods that are not based on particle filters, like EKF e.g., [14] or optimization, e.g., [26], [27], to mention some examples. The proposed data association filter will be called Data Association Particle Filter or DAPF and its performance will be examined with a small example to get some initial results and see the feasibility of the approach. It shall also be noted that, the approach is applicable to both batch and sequential estimation cases since the filter is run forward for each time step and it is only data from that time step that is needed for the estimation of the data association.

The paper is organized as follows; Section I, this section, introduces the problem and some background. Section II describes the data association problem and setup, while Section III explains the basics of the particle filter in a brief way. Section IV handles the implementation details of the DAPF and Section V illustrates some results obtained with simulated data in a simple scenario. Finally, in Section VI conclusions and some future work prospects are given.

II. Data Association

A data association problem, in this context, is a problem of assigning a set of measurements of landmarks at a certain time,

$Y_t = \{y_t^i\}_{i=1}^{m_t}$ and $y_t^i \in \mathbb{R}^{d_y}$, to the correct landmarks. The set of landmarks is defined as $L = \{l^j\}_{j=1}^{N_l}$ and $l^j \in \mathbb{R}^{d_l}$. Here, d_y and d_l are dimensions of the measurements and the landmarks, respectively. The association can then be defined as finding a correspondence variables at time t, $C_t = \{c_t^j\}_{j=1}^{N_l}$, such that $c_t^j = i$ if measurement y_t^i is originating from landmark l^j, otherwise $c_t^j = 0$. For example, if there are 5 landmarks in total and 2 measurements available at time $t = 6$, and it is landmarks 3 and 5 that are measured by measurements 2 and 1, the correspondence variables will be $c_6^1 = 0,\ c_6^2 = 0,\ c_6^3 = 2,\ c_6^4 = 0,\ c_6^5 = 1$, or $C_6 = \{0,\ 0,\ 2,\ 0,\ 1\}$.

This can be done in many ways, depending on the nature of the measurements and the other information that is available. Normally, if the measurements are coming from the camera images, they will be regarded as direction (or bearing) only and that will be the only available information. In some cases other type of attributes can also be extracted from the images and associated with the landmarks and image measurements. An example of this kind of attributes is the SIFT descriptor [28]. No matter which kind of information from the measurements and the landmarks is used, the goal of the data association is to find measurements and landmarks that belong to each other according to some fitness measure, $f(C_t, Y_t, L, \theta_t)$, under the certain constraints on the possible associations. Here θ_t denotes any other variable that is needed to evaluate the fitness function, for example platform's position. In general, the best association, $\widehat{C}_t$, should be the one giving the best fitness measure value over all possible associations under the constraints, i.e.,

$$\widehat{C}_t = \arg\max_{C_t} f(C_t, Y_t, L, \theta_t) \quad \textbf{s.t. } C_t \in \mathsf{C} \tag{1}$$

where C defines a set of possible associations, for example a constraint that only one measurement can be assigned to at most one landmark and vice versa. This problem can be solved in many different ways, as already mentioned in Section I.

III. Particle Filter

In this chapter a brief description of the particle filter will be done, and for more details see e.g., [19]. A particle filter is a Monte Carlo based approach for approximating optimal filtering densities when these are intractable to calculate. This is very common in many cases, for example when the underlying models are not linear. For this reason, particle filters has been used as an important enabler to solve non-linear estimation problems. Basically, the aim of the estimation is to find a posterior distribution of a stochastic variable $\mathbb{X}$ (normally denoted state) given a set of observations (or measurements) $\mathbb{Y}$, denoted $p(\mathbb{X}|\mathbb{Y})$. Usually, the sought variable cannot be observed directly and some estimation method must be applied. In a filtering context, the stochastic variable and the observations are normally time series, i.e., $\mathbb{X} = \{x_t\}_{t=0}^T = x_{0:T}$ and $\mathbb{Y} = \{Y_t\}_{t=1}^T = Y_{1:T}$. The posterior distribution becomes then $p(x_{0:T}|Y_{1:T})$. Here it is assumed that $x_t \in \mathbb{R}^{n_t}$ and $Y_t \in \mathbb{R}^{m_t \cdot d_y}$, i.e., the dimension of the state and the observation can change in different time steps, and m_t and d_y are defined before. In order to estimate the posterior, there are two conditional distributions that will be used, state transition distribution, $p(x_t|x_{t-1})$, that assumes Markov properties of the state, as well as likelihood, $p(Y_t|x_t)$, relating the observations with the state. Given these distributions, the optimal filtering can be formulated as follows

$$p(x_t|Y_{1:t-1}) = \int p(x_t|x_{t-1})p(x_{t-1}|Y_{1:t-1})\ dx_{t-1} \tag{2a}$$

$$p(x_t|Y_{1:t}) = \frac{p(Y_t|x_t)p(x_t|Y_{1:t-1})}{\int p(Y_t|x_t)p(x_t|Y_{1:t-1})\ dx_t} \tag{2b}$$

The first equation is called the prediction step and is used to predict the state at time t given all the observations up to time $t-1$. The second equation is called the measurement update and is used to incorporate the observation at time t in the state at the same time. Notice that the denominator in the update equation is a normalization factor, such that the posterior distribution integrates to 1. These two steps are computed sequentially for each time resulting in a filtered posterior distribution of the state $p(x_t|Y_{1:t})$ for each time step. Notice also that the distributions above might be conditioned not only on a state, but on some other variables too, like e.g., L or θ_t in (1). This conditioning will be implicitly assumed in the rest of the paper but will be included in cases where clarification is needed.

The problem with the approach above is that very few distributions allow for an exact solution of the expressions above and closed form expressions of the resulting distributions. In order to solve that problem, an approximate solution can be done, where the distributions above are represented with N particles, $\{x_t^i\}_{i=1}^N$ with associated importance weights $\{w_t^i\}_{i=1}^N$. With these, and the proposal distribution used to draw particle samples from, the particle filter can be implemented according to Algorithm 1. Note that in the case of data association, the state variable x_t will be the association variable C_t, and the distributions will be seen as probability mass functions (pmf) defined on a discrete space, i.e., $C_t \in \mathbb{N}^{N_l}$ for all t. In this case the integral becomes summation in (2a).

IV. Data Association Particle Filter

In order to implement the Data Association Particle Filter (DAPF) we need to define the pmfs in (2) and Algorithm 1. For the comparison sake two proposal functions will be defined, one that is uniform and other one that is using information from the measurements in order to sample the most probable association. Also, the information from the measurements will be used in both cases in order to define how many associations to assign. In all cases the exact probability densities and likelihoods cannot be expressed exactly and some approximations must be done. These will be explained further below.

The sampling from a proposal distribution in step 1. in Algorithm 1 will be performed in two ways, as mentioned above, and these will be compared with respect to execution time and association performance. The common thing applicable to both proposals is decision on how many landmarks were measured. This is done by assuming a transition probability between observed and not observed landmarks at time $t-1$

Algorithm 1 Particle Filter

Require: $p(x_0)$, $p(x_t|x_{t-1})$, $p(Y_t|x_t)$, $\pi(x_t|x_{0:t-1}, Y_{1:t})$, $Y_{1:T}$
Ensure: $\{\hat{p}(x_t|Y_{1:t})\}_{t=0}^T$

Initialize:
Draw N initial particles from the prior distribution and set their importance weights to equal weight:
$x_0^i \sim p(x_0)$, $w_0^i = \frac{1}{N}$, $i = 1 : N$

for $t = 1$ **to** T **do**
1. Draw N samples from the proposal distribution:
   $x_t^i \sim \pi(x_t|x_{0:t-1}^i, Y_{1:t})$, $i = 1 : N$
2. Update the importance weights:
   $\tilde{w}_t^i = w_{t-1}^i \frac{p(Y_t|x_t^i)p(x_t^i|x_{t-1}^i)}{\pi(x_t^i|x_{0:t-1}^i, Y_{1:t})}$
3. Normalize the weights:
   $w_t^i = \frac{\tilde{w}_t^i}{\sum_{j=1}^N \tilde{w}_t^j}$
4. Approximate the filtering posterior distribution:
   $\hat{p}(x_t|Y_{1:t}) = \sum_{i=1}^N w_t^i \delta(x_t^i - x_t)$
5. Resample the particles:
   Draw N particles from $\{x_t^i\}_{i=1}^N$ with the probability proportional to their respective weight and set $w_t^i = \frac{1}{N}$, $i = 1 : N$ thereafter

end for

and time t. Two different transition probabilities will be used; one that is describing the probability that a landmark observed at time $t-1$ is observed at time t as well, called P_O, and used for the observed landmarks, and the other one describing the probability that a landmark that is not observed at time $t-1$ is observed at time t, called P_N, and used on the not observed ones. These probabilities are user specified and are used in conditional Bernoulli distributions as

$$p_O(\mathbb{I}(c_t^j)|\mathbb{I}(c_{t-1}^j)) = P_O^{\mathbb{I}(c_{t-1}^j)}(1-P_O)^{1-\mathbb{I}(c_{t-1}^j)} \tag{3a}$$

$$p_N(\mathbb{I}(c_t^j)|\mathbb{I}(c_{t-1}^j)) = P_N^{1-\mathbb{I}(c_{t-1}^j)}(1-P_N)^{\mathbb{I}(c_{t-1}^j)} \tag{3b}$$

where $\mathbb{I}(x) = 0$ if $x = 0$ and $\mathbb{I}(x) = 1$ if $x > 0$. The number of possible landmarks that is used is not the set of all the landmarks, but a subset chosen based on the current estimate of the platform's state, sensor characteristics, for example field-of-view, and landmark positions, giving $j \in N_t^l \subset \{1, ..., N_l\}$. This is done in order to speed up the computations.

When the possible observed landmarks has been decided, giving the set $\tilde{L}_t \subset N_t^l$ with cardinality n_t, an assignment of these to the measurements is performed. Depending on the number of available measurements and observed landmarks two cases are distinguished: if the number of observed landmarks is greater or equal to the number of measurements, $n_t >= m_t$, all the measurements are assigned, but not all the landmarks are used (except in the equality case) and if the number of observed landmarks is less than the number of available measurements, $n_t < m_t$, all the observed landmarks will be assigned, but not all the measurements.

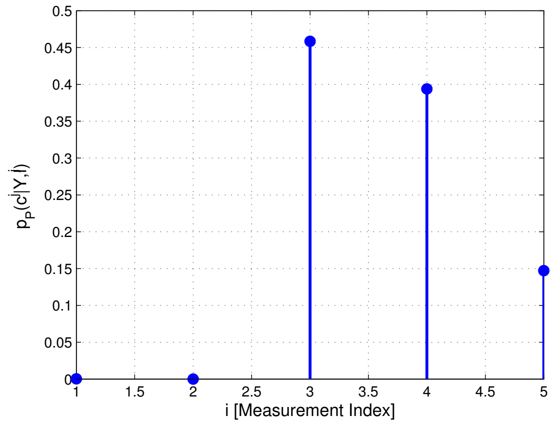


Figure 1: An example of a proposal distribution p_P for a single landmark and 5 measurements. The landmarks is measured by the measurement number 3.

In case the uniform proposal is used the assignments will be drawn from a uniformly distributed integers between 1 and m_t, i.e, $c_t^j \sim \mathsf{U}(1, m_t)$ and $j \in \tilde{L}_t$.

In case the non-uniform proposal is used, the assignments will be drawn from a approximate distribution based on measurements, platform's states and landmarks' positions. This approximate distribution (for a single landmark) can be described as

$$p_P(c_t^j|Y_t, \tilde{L}_t, \theta_t) = e^{-\mu(y_t^{c_t^j}-h(\theta_t,l^j))^T(y_t^{c_t^j}-h(\theta_t,l^j))} \tag{4}$$

Here, μ is seen as a tuning variable and is user defined. The measurement function, $h(\theta, l)$, is used to create a estimated measurement given the landmarks and the platform's position and orientation. Exactly how the measurement function is defined is dependent on the setup. An example plot of the distribution above for a single landmark and 5 measurements is given in Figure 1. Sampling from this distribution is simply performed by rejection sampling, see e.g., [15]. During sampling, in both uniform and non-uniform case, the care is taken to handle one-to-one assignment, i.e., only one measurement can be assigned to only one landmark. All the steps above are summarized in Algorithm 2. In this way a set of particles with proposed associations will be created, $\{C_t^i\}_{i=1}^N$ which is the result of step 1. in Algorithm 1.

The exact likelihood function that should be used in step 2. in Algorithm 1 is impossible to find exactly, and in this work a viable approximation is used instead. It is defined as

$$p(Y_t|C_t, L, \theta_t) = \prod_{j \in L} \left[e^{-\nu|y_t^{c_t^j}-h(\theta_t,l^j)|\mathbb{I}(c_t^j)} \prod_{k=1}^{m_t} \left(1 - e^{-\nu|y_t^k - h(\theta_t,l^j)|}\right)^{1-\mathbb{I}(c_t^j)} \right] \tag{5}$$

where $\mathbb{I}(x)$ is defined as before. The intuition behind this approximation is based on making the total likelihood as high as possible for both the landmarks that are correctly measured and associated (the first term of the outer product)

Algorithm 2 Proposal Distribution $\pi(C_t|C_{t-1}^i, Y_t)$

Require: $\{C_{t-1}^i\}_{i=1}^N$, Y_t, θ_t
Ensure: $\{C_t^i\}_{i=1}^N$

for $i = 1$ **to** N **do**
  1. Decide possible observed landmarks, $\tilde{L}_t$, by using (3)
  for $j \in \tilde{L}_t$ **do**
    2. Draw assignments with care taken to one-to-one condition:
    if The uniform proposal is used **then**
      Draw assignments from a discrete uniform distribution: $c_t^{j,i} \sim \mathsf{U}(1, m_t)$
    else
      Draw assignments from pmf in (4) with rejection sampling: $c_t^{j,i} \sim p_P(c_t^j|Y_t, \tilde{L}_t, \theta_t)$
    end if
  end for
end for

and the landmarks that are correctly not measured and not associated (the second term, i.e., the inner product). This approximation can be seen as similar to a (non-normalized) multinomial distribution pmf for each association combination, where the probabilities are represented by the exponential terms depending on the actual association and the number of outcomes is 2, i.e., associated or non-associated landmarks. Note also that it is absolute value of the difference between the measurements and the measurement function that is used here. This is done to have a more heavy-tail likelihood and not punish large errors too much. Just as before, ν is seen here as a tuning parameter and is user defined. Exactly how to tune the parameters μ and ν is not easy to suggest, but in general they are related to the measurement noise and accuracy of the parameter θ. The tuning parameters should be chosen larger if the measurement noise is big or the expected θ accuracy is low, and smaller otherwise. An illustrative example of the likelihood function is shown in Figure 2. Here, three examples are used, all with two landmarks in order to be able to illustrate the likelihood in a two-dimensional graph, with different number of measurements. In all three examples, it can be seen that the likelihood is the highest for a correct association.

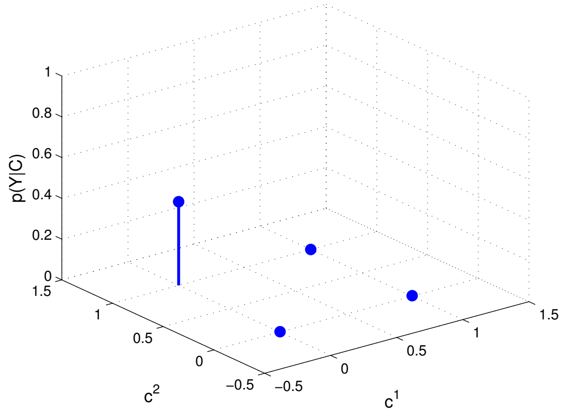
With these pmfs defined, Algorithm 1 can be run and the approximation of the posterior distribution of the data association can be obtained. Although the full approximation of the filtering posterior is available after each time step, for evaluation purposes only the maximum a posteriori estimate is used, i.e.,

$$\widehat{C}_t = \arg\max_{C_t} \left(\hat{p}(C_t|Y_{1:t})\right) \tag{6}$$

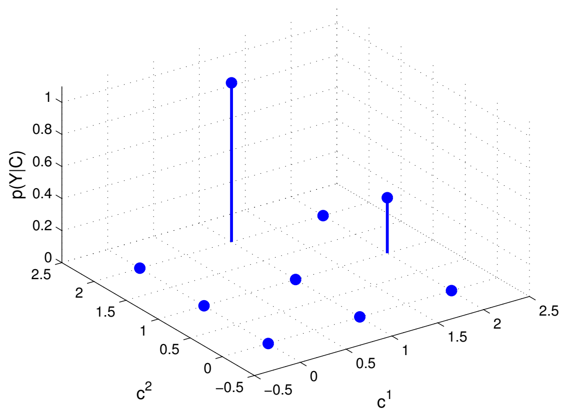
where $\hat{p}(C_t|Y_{1:t})$ is defined in step 4. in Algorithm 1.

V. Experimental Results

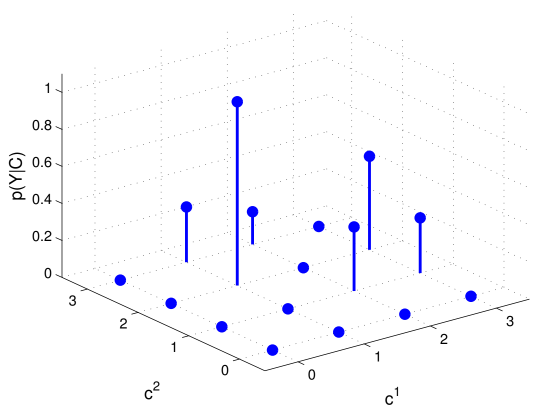
In order to illustrate the results of the approach, a simple two-dimensional case is used. The platform state is defined as



(a) Likelihood function for two landmarks and only one measurement. Only landmark 2 is measured, i.e., $C = \{0,\ 1\}$.



(b) Likelihood function for two landmarks and two measurements. Both landmarks are measured, landmark 1 is measured with measurement 1 and landmark 2 with measurement 2, i.e., $C = \{1,\ 2\}$.



(c) Likelihood function for two landmarks and three measurements. Both landmarks are measured, landmark 1 with measurement 1 and landmark 2 with measurement 2 and the third measurement is seen as clutter, i.e., $C = \{1,\ 2\}$.

Figure 2: Illustration of the likelihood function in (5) for three the case with two landmarks and different number of measurements.

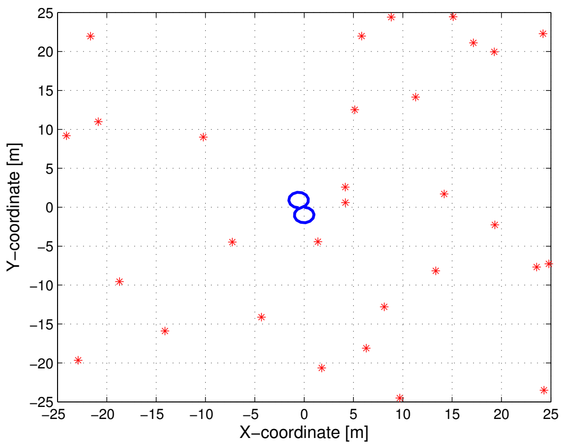


Figure 3: The trajectory (in blue) and one of the landmark configurations (red *) used for the performance evaluation of DAPF.

$\theta_t = [p_t^x \; p_t^y \; \psi_t]^T$, i.e., the position in X- and Y-dimensions as well as orientation. Also, the landmark coordinates are two-dimensional, $l = [l^x, l^y]^T$, and the measurements are one-dimensional directions from the platform to the landmarks measured in the platform's reference frame. This gives that the measurement function, $h(\theta, l)$, is defined as

$$\begin{bmatrix} g_1 \\ g_2 \end{bmatrix} = \begin{bmatrix} \cos(\psi) & \sin(\psi) \\ -\sin(\psi) & \cos(\psi) \end{bmatrix} \begin{bmatrix} l^x - p^x \\ l^y - p^y \end{bmatrix} \tag{7a}$$

$$h(\theta, l) = \arctan\left(\frac{g_2}{g_1}\right) \tag{7b}$$

It is assumed that the sensor has a limited field-of-view, which in this case was 35 degrees. For the simulation performed here, a trajectory consisting of 25 time steps and with sampling time of 1 second according to Figure 3 is used. The trajectory is plotted in blue.

The evaluation is performed for two cases; the first one where the landmark positions and the platform state are assumed to be correct and used as such in the DAPF, and the other one where the platform state is randomly perturbed, but the correct landmark positions are used. In both cases a small noise is added to the measurements. 100 Monte Carlo simulations are performed for each case. In the first case the same trajectory is used, but different landmark configurations are randomized for each run. In the second case the same landmark configuration is used, but, as mentioned, the platform state is varied for each run. In the second case a comparison to the deterministic association, Nearest-Neighbor, is also done. Also, the performance is evaluated for the different number of particles, N, and $N \in \{500,\ 1000,\ 2000,\ 4000,\ 8000\}$. The utilized performance measure is the average number of wrong associations over all times and MC-simulations for each number of used particles, i.e.,

$$\bar{C}(N) = \frac{1}{25 \cdot 100} \sum_{t=1}^{25} \sum_{k=1}^{100} \tilde{C}(N)_t^k \tag{8}$$

where $\tilde{C}(N)_t^k$ is defined as number of wrong associations at time step t and MC run k. The wrong association in this case is considered both missed association, i.e., no assignment to a landmark when there is one, as well as wrong assignment, i.e., wrong measurement assigned to a landmark.

The results can be seen in Figure 4. In 4a the first case from above is depicted, and, as expected, the uniform proposal distribution performs worse then the one using measurement information. In 4b a comparison of the both proposal variants and the deterministic approach is shown. It is interesting to see that DAPF with the non-uniform proposal actually outperforms the deterministic approach in average, while the one with the uniform proposal does not. However, it is approaching the deterministic one's performance for the relatively big number of particles (8000 in this case). It can also be seen that the non-uniform proposal case does not improve much for the increased number of particles beyond 2000. The main drawback of the non-uniform proposal can be seen in 4c, where that approach is about three times slower than the uniform proposal, in average. Depending on application, this could be an acceptable trade-off.

VI. Conclusions and Future Work

In this work a sequential Monte Carlo method, the particle filter, to solve the data association problem is proposed, DAPF, and the performance is investigated on a small simulated example. The performance is evaluated for two different proposal distributions and different number of used particles, as well as compared to a deterministic approach, Nearest-Neighbor in this case. The obtained results are quite promising and show that the suggested method with non-uniform proposal performs better than the deterministic approach, with the price of higher execution time.

Since this is just an initial work, as a next step a more thorough investigation of the performance is suggested, as well as more extensive testing on real data sets.

References

[1] W. Wijesoma, L. Perera, and M. Adams, "Toward multidimensional assignment data association in robot localization and mapping," *Robotics, IEEE Transactions on*, vol. 22, no. 2, pp. 350 – 365, april 2006.

[2] D. Hähnel, S. Thrun, B. Wegbreit, and W. Burgard, "Towards Lazy Data Association in SLAM," in *ISRR*, ser. Springer Tracts in Advanced Robotics, P. Dario and R. Chatila, Eds., vol. 15. Springer, 2003, pp. 421–431.

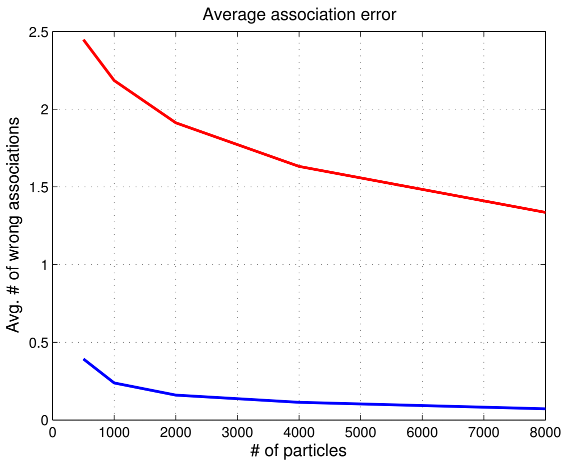
[3] F. Dellaert, S. Seitz, C. Thorpe, and S. Thrun, "EM, MCMC, and Chain Flipping for Structure from Motion with Unknown Correspondence," *Machine Learning*, vol. 50, no. 1-2, pp. 45–71, 2003.

[4] Y. Bar-Shalom, T. Kirubarajan, and X.-R. Li, *Estimation with Applications to Tracking and Navigation*. New York, NY, USA: John Wiley & Sons, Inc., 2002.

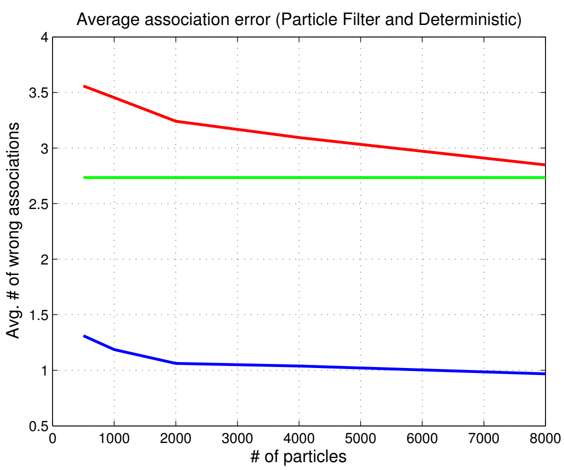
[5] T. Bailey and H. Durrant-Whyte, "Simultaneous Localization and Mapping (SLAM): Part II," *IEEE Robotics & Automation Magazine*, vol. 13, no. 3, pp. 108–117, Sep. 2006.

[6] H. Durrant-Whyte and T. Bailey, "Simultaneous Localization and Mapping: Part I," *IEEE Robotics & Automation Magazine*, vol. 13, no. 12, pp. 99–110, June 2006.

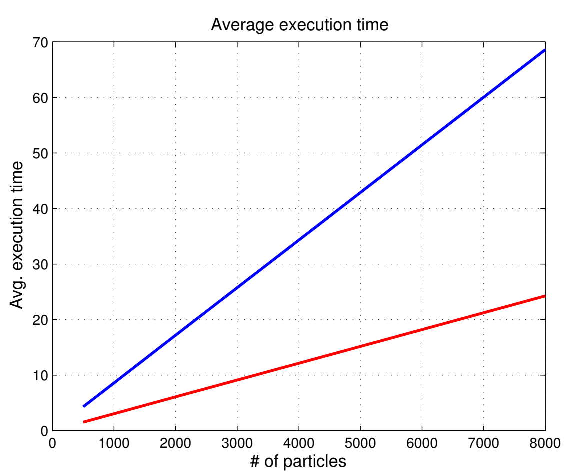
[7] T. Lemaire, C. Berger, I.-K. Jung, and S. Lacroix, "Vision-Based SLAM: Stereo and Monocular Approaches," *Int. J. Comput. Vision*, vol. 74, no. 3, pp. 343–364, Sep. 2007. [Online]. Available: http://dx.doi.org/10.1007/s11263-007-0042-3



(a) Average number of wrong associations for uniform (red) and non-uniform (blue) proposal distribution as a function of number of particles. The platform states and landmark positions are correct.



(b) Average number of wrong associations for uniform (red) and non-uniform (blue) proposal compared to deterministic association (green) as a function of number of particles. The platform positions and orientation are perturbed in each time.



(c) Average execution time for uniform (red) and non-uniform (blue) proposal distribution as a function of number of particles.

Figure 4: Average performance of the DAPF for uniform and non-uniform proposal distribution.

[8] M. Pollefeys, D. Nistér, J. M. Frahm, A. Akbarzadeh, P. Mordohai, B. Clipp, C. Engels, D. Gallup, S. J. Kim, P. Merrell, C. Salmi, S. Sinha, B. Talton, L. Wang, Q. Yang, H. Stewénius, R. Yang, G. Welch, and H. Towles, "Detailed Real-Time Urban 3D Reconstruction from Video," *Int. J. Comput. Vision*, vol. 78, no. 2-3, pp. 143–167, Jul. 2008. [Online]. Available: http://dx.doi.org/10.1007/s11263-007-0086-4

[9] C. Taylor, D. Kriegman, and P. Anandan, "Structure and Motion in Two Dimensions from Multiple Images: A Least Squares Approach," in *Proceedings of the IEEE Workshop on Visual Motion*, Princeton, NJ, USA, October 1991, pp. 242 –248.

[10] Z. Zhang, "A New Multistage Approach to Motion and Structure Estimation: From Essential Parameters to Euclidean Motion Via Fundamental Matrix," INRIA, France, Tech. Rep., 1996.

[11] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, 2004.

[12] B. Triggs, P. Mclauchlan, R. Hartley, and A. Fitzgibbon, "Bundle adjustment - a modern synthesis," in *Vision Algorithms: Theory and Practice*, ser. Lecture Notes in Computer Science, B. Triggs, A. Zisserman, and R. Szeliski, Eds., vol. 1883. Springer-Verlag, 2000, pp. 298–372.

[13] T. Lupton and S. Sukkarieh, "Visual-Inertial-Aided Navigation for High-Dynamic Motion in Built Environments Without Initial Conditions," *Robotics, IEEE Transactions on*, vol. 28, no. 1, pp. 61 –76, feb. 2012.

[14] M. Li and A. I. Mourikis, "High-precision, consistent ekf-based visual-inertial odometry," *International Journal of Robotics Research*, vol. 32, no. 6, pp. 690–711, May 2013. [Online]. Available: http://dx.doi.org/10.1177/0278364913481251

[15] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Berlin, Heidelberg: Springer-Verlag, 2006.

[16] D. P. Bertsekas, "An Auction Algorithm for Shortest Paths," *SIAM Journal on Optimization*, vol. 1, no. 4, pp. 425–447, 1991. [Online]. Available: http://dx.doi.org/doi/10.1137/0801026

[17] H. W. Kuhn, "The Hungarian Method for the Assignment Problem," *Naval Research Logistics Quarterly*, vol. 2, no. 1-2, pp. 83–97, 1955. [Online]. Available: http://dx.doi.org/10.1002/nav.3800020109

[18] J. Munkres, "Algorithms for the Assignment and Transportation Problems," *Journal of the Society for Industrial and Applied Mathematics*, vol. 5, no. 1, pp. pp. 32–38, 1957. [Online]. Available: http://www.jstor.org/stable/2098689

[19] N. Gordon, D. Salmond, and A. Smith, "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," *IEEE Proceedings F, Radar and Signal Processing*, vol. 140, no. 2, pp. 107–113, 1993.

[20] N. Bergman and A. Doucet, "Markov chain Monte Carlo data association for target tracking," in *2000 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings*, vol. 2, June 2000, pp. II705–II708 vol.2.

[21] R. Karlsson and F. Gustafsson, "Monte carlo data association for multiple target tracking," in *IEE International Seminar Target Tracking: Algorithms and Applications*, 2001.

[22] D. Schulz, W. Burgard, D. Fox, and A. B. Cremers, "Tracking multiple moving targets with a mobile robot using particle filters and statistical data association," in *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation*, vol. 2, May 2001, pp. 1665–1670 vol.2.

[23] 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.

[24] O. Frank, J. Nieto, J. Guivant, and S. Scheding, "Multiple target tracking using Sequential Monte Carlo Methods and statistical data association," in *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003)*, vol. 3, Oct 2003, pp. 2718–2723 vol.3.

[25] S. Oh, S. Russell, and S. Sastry, "Markov Chain Monte Carlo Data Association for Multi-Target Tracking," *IEEE Transactions on Automatic Control*, vol. 54, no. 3, pp. 481–497, March 2009.

[26] Dellaert, F. and Kaess, M., "Square Root SAM: Simultaneous Localization and Mapping via Square Root Information Smoothing," *International Journal of Robotics Research*, vol. 25, no. 12, pp. 1181–1203, 2006. [Online]. Available: http://dx.doi.org/10.1177/0278364906072768

[27] Z. Sjanic and M. A. Skoglund, "Prediction error method estimation for Simultaneous Localisation and Mapping," in *2016 19th International Conference on Information Fusion (FUSION)*, July 2016, pp. 927–934.

[28] D. Lowe, "Object Recognition from Local Scale-Invariant Features," in *Proceedings of the Seventh International Conference on Computer Vision (ICCV'99)*, Corfu, Greece, 1999, pp. 1150–1157.