Decision-Theoretic Monte Carlo Smoothing for Scaling Probabilistic Tracking in Hybrid Dynamic Systems

Vandi Verma and Reid Simmons
Robotics Institute
Carnegie Mellon University
5000 Forbes Ave.
Pittsburgh, PA 15213, USA
412-268-1858
{vandi, reids}@cs.cmu.edu

Abstract— Detecting faults on-board planetary rovers is important since human intervention may not be possible due to communication delays. In this paper we propose a scalable method for on-board fault detection and identification that may be applied to general fault models with limited computation. Although our application focus is on diagnosing rover faults, this method is applicable in general for tracking any general non-linear, non-Gaussian hybrid (discrete-continuous) dynamic system online.

Our formulation of the fault detection problem requires estimating robot and environmental state, as it changes over time, from a sequence of noisy sensor measurements. We propose a Monte Carlo algorithm that will generate new trajectories if the probability of the current set of fault hypothesis being tracked is low. This approach maintains a fixed lag history of measurements, controls and samples.

Experimental results of a dynamic simulation of a six-wheel rocker-bogie rover show a significant improvement in performance over the classical approach.

Keywords— Fault diagnosis, particle filters, tracking, Bayesian filtering, Monte Carlo smoothing, autonomous rovers

TABLE OF CONTENTS

0-7803-7231-X/01/$10.00/© 2002 IEEE
IEEEAC paper # 1228(version 2)

1. INTRODUCTION

Planetary rovers operate in environments where human intervention is expensive, slow, unreliable, or impossible. It is therefore essential to monitor the behavior of these robots so that contingencies may be addressed before they result in catastrophic failures. This monitoring needs to be efficient since there is limited computational power available on rovers.

The algorithm presented in this paper provides a probability distribution over the various fault and operational states of the rover. Detecting and classifying faults in terms of a probability distribution over fault states captures the uncertainty in the estimate that results from noisy and insufficient data. In addition, it allows a lot of flexibility in the types of planners/controllers that may be used for controlling the robot and for recovering from faults. For example, such distributions are compatible with classical conditional planners or a Markov Decision Processes (MDPs), which use the most likely state to determine which action to take, and also Partially Observable Markov Decision Processes (POMDPs), which use the distribution over the entire state space.

Our formulation of the fault detection problem requires estimating robot and environmental state, as it changes over time, from a sequence of sensor measurements that provide noisy, partial information about the state. Particle filters address this problem and have been extensively used for Bayesian state estimation in nonlinear systems with noisy measurements. They approximate the probability distribution with a set of samples or particles.

Particle filters have a number of characteristics that make them attractive for fault detection on robots: they can represent arbitrary distributions, handle hybrid state spaces and noise, and can easily be extended to an anytime approach where the estimation accuracy can be adjusted to match available computation. The main problem is that a large number of particles are often needed to maintain a reasonable approximation of the state probability distribution and to enable detection of rare events. This is because particle filters, are predictive, i.e. they generate a hypothesis about a fault at the time a measurement is taken. Regardless of whether they take the next measurement into account to improve the prediction, it's still a prediction. This results in a high variance in the possible fault hypothesis given a finite number of samples.

We propose an algorithm that will re-evaluate past assignments to generate new trajectories if the current hypothesis is unlikely. This approach maintains a fixed lag history of measurements, controls and particles. The process of estimating state based on future measurements is termed smoothing in the literature. Our approach is different from existing Monte Carlo smoothing methods since it is not restricted to re-weighting existing hypothesis, but instead generates new hypothesis with higher posterior likelihood.

Our experimental results of a dynamic simulation of a six-wheel rocker-bogie rover show a significant improvement in performance over the classical approach.

2. Filtering and Smoothing

State estimation is the process of determining the sate of the system from a sequence of data. There are two main classes of state estimation methods. Batch estimation methods and recursive estimation methods.

Bath methods treat all the data with equal importance and find an optimal estimate of the state given the entire sequence. However, full batch estimation is computationally expensive and gets exponentially slower as the rover accumulates monotonically increasing volumes of data. It is therefore not suitable for FDI.

Recursive state estimation methods incorporate the data as it becomes available and replace the data with a statistic. Estimates at subsequent timesteps use this statistic instead of the history of data for state estimation. A majority of the popular recursive state estimation methods make a Markov assumption, i.e., the past and future are conditionally independent given the current state. If the statistic is a sufficient statistic, the performance of recursive state estimation is comparable to batch estimation.

Bayesian Filtering for Recursive State Estimation— Filtering is defined as the process of estimating the state, s_t, at time t, given the sequence of data upto time t, $z_0 \dots z_t$. Hence, it is a forward process. Bayes filters are based on Bayes rule from probability theory.

We denote the multivariate state at time t as s_t and measurements or observations as z_t. Since we concentrate on the discrete time, first order Markov formulation of the dynamic state estimation problem. Hence, the state at time t is a sufficient statistic of the history of measurements, i.e. $p(s_t \mid s_{0:t-1}) = p(s_t \mid s_{t-1})$ and the observations depend only on the current state, i.e. $p(z_t \mid s_{0:t}) = p(z_t \mid s_t)$. The posterior distribution at time t, $p(s_{0:t} \mid z_{1:t})$, includes all the available information up to time t and provides the optimal solution to the state estimation problem.

The filtering distribution, $p(s_t \mid z_{1:t})$, is a marginal of the posterior distribution, $p(s_{0:t} \mid z_{1:t})$. The recursive filter defined as follows, using Bayes rule and the Markov assumption:

$$p(s_t \mid z_{1:t}) = \eta_t \, p(z_t \mid s_t) \int p(s_t \mid s_{t-1}) \, p(s_{t-1} \mid z_{1:t-1}) \, ds_{t-1} \tag{1}$$

This process is known as Bayesian filtering, optimal filtering or stochastic filtering and may be characterized by three distributions: (1) a state transition model $p(s_t \mid s_{t-1})$, (2) a measurement model $p(z_t \mid s_t)$, and, (3) an initial prior distribution, $p(s_0)$. Bayesian filtering provides an optimal estimate of the posterior, at time t, given the sequence of data ,$[0 \dots t]$, up to time t. It is however not the optimal estimate, given future data. A batch estimate of a sequence of data from time $[0 \dots T]$, where $T > t$, would however provide a more accurate estimate.

Smoothing is a backward process of updating past estimates based on future data, i.e. $p(s_t \mid s_{0:t+l})$, $l > 0$. It achieves a balance between the efficiency of filtering and the optimality of batch methods. Bayesian smoothing is defined as follows:

$$p(s_t \mid z_{1:t+l}) = \gamma_t \int p(s_{t+1} \mid s_t) \, p(s_t \mid z_{1:t}) \, p(s_{t+1} \mid z_{1:t+l}) \, ds_{t+1} \tag{2}$$

Although the mathematical structure of the optimal Bayesian solution is well understood [2], [?], [?], for general problems the computational complexity of the optimal filtering algorithm limits it's practical applicability using even the fastest computers available in the foreseeable future. In general robots (specially planetary rovers) tend to be even more power and computation constrained than the state of the art. Particle filters are a popular method for computing tractable approximations to this posterior.

3. Background on Particle Filters

Bayesian Model for Tracking Faults

Fault detection and identification (FDI) has a natural representation as a state estimation problem. We represent the possible fault and operational modes of the systems as explicit states. The sequence of measurements are then used to determine the state of the system.

Let D represent the finite set of discrete fault and operational modes of the rover, $d_t \in D$ the state of the rover at time t and $\{d_t\}$ the discrete, first order Markov chain representing the evolution of the state over time. The problem of monitoring the state of the rover consists of providing a belief (a distribution over the state set D) at each time step as it evolves based on the following transition model:

$$p(d_t = j \mid d_{t-1} = i), \quad (i, j \in D) \tag{3}$$

Each of the discrete fault and operational modes changes the dynamics of the rover. Let $x_t \in \mathbb{R}^{\aleph_n}$ denote the multivariate continuous state of the rover at time t. The non-linear conditional state transition models are denoted by $p(x_t \mid x_{t-1}, d_t)$. The state of the rover is observed through a sequence of measurements, $\{z_t\}$, based on the measurement model $p(z_t \mid x_t, d_t), z_t \in \mathbb{R}^{\aleph_F}$.

Classical Particle Filter

Filtering for FDI consists of estimating the marginal $p(d_t \mid z_{1..t})$ of the posterior distribution $p(x_t, d_t \mid z_{1..t})$. A recursive estimate of this posterior distribution may be obtained with a Bayes filter using the following factorization:

$$p(x_t, d_t \mid z_{1..t}) = \eta_t \, p(z_t \mid x_t, d_t) \times \int \sum_{d_{t-1}} p(x_t, d_t \mid x_{t-1}, d_{t-1}) dx_{t-1} \quad (4)$$

There is no closed form solution to this recursion, hence we use a particle filter approximation. A particle filter (PF) [7], [12] is a Monte Carlo approximation of the posterior in a Bayes filter. PFs approximate the posterior with a set of N fully instantiated state samples or particles $\{(d_t^{[1]}, x_t^{[1]}) \ldots (d_t^{[N]}, x_t^{[N]})\}$ and importance weights $\{w_t^{[i]}\}$:

$$\hat{p}_N(x_t, d_t \mid z_{1..t}) = \sum_{i=1}^{N} w_t^{[i]} \, \delta_{x_t^{[i]}, d_t^{[i]}}(x_t, d_t) \quad (5)$$

where $\delta(\cdot)$ denotes the Dirac delta function. It can be shown that as $N \to \infty$ the approximation in (5) approaches the true posterior density [23]. Because it is difficult to draw samples from the true posterior, we instead draw them from a more tractable distribution $q(\cdot)$, called the proposal (or importance) distribution. There are a large number of possible choices for the proposal distribution, the only condition being that its support must include that of the posterior. The importance weights are used [20], [21] to account for the discrepancy between the proposal distribution $q(\cdot)$ and the true distribution $p(x_t, d_t \mid z_{1:t})$, and for a given sample $x_t^{[i]}, d_t^{[i]}$ the importance weight is

$$w_t^{[i]} = p(x_t^{[i]}, d_t^{[i]} \mid x_{t-1}^{[i]}, d_{t-1}^{[i]}, z_t) \;/\; q(x_t^{[i]}, d_t^{[i]}) \quad (6)$$

The simplest choice for the proposal distribution is the transition probability

$$\begin{aligned} q(x_t, d_t) &= p(x_t, d_t \mid x_{t-1}^{[i]}, d_{t-1}^{[i]}) \\ &= p(x_t \mid x_{t-1}^{[i]}, d_t) p(d_t \mid d_{t-1}^{[i]}) \end{aligned} \quad (7)$$

in which case the importance weight is equal to the likelihood

$$p(z_t \mid x_{t-1}^{[i]}, d_{t-1}^{[i]}, z_{1...t-1})$$

This is the most widely used proposal distribution [1], [8], [13] and is simple to compute, but it can be inefficient since it ignores the most recent measurement z_t. Particularly in the fault diagnosis domain, where there are a large number of possible faults that may occur at any instant in time, the most recent measurement can be very informative.

A well-known problem with particle filters is that a large number of particles are often needed to obtain a reasonable approximation of the posterior distribution. For real-time fault detection and identification, maintaining such a large number of particles is typically not practical. However, the variance of the particle-based estimate can be high with a limited number of samples, since a large number of faults may potentially occur at any instant.

Rao-Blackwellized Particle Filter

RBPFs factor the state space, so that a subset of the variables may be computed analytically given the values of the rest of variables, which are obtained from particle filtering [17], [6]. Since our representation of the fault diagnosis problem has a natural factorization and the discrete state transitions $p(d_t \mid d_{t-1}$ are conditionally independent, Rao-Blackwellized particle filters may be used to track the state space. Discrete states are sampled as in a classical particle filter, and conditioned on these discrete samples, $d_t^{[i]}$, the continuous state $x_t^{[i]}$ is computed analytically.

The most common representation for the conditional probability density $p(x_t \mid x_{t-1}^{[i]}, d_t^{[i]})$, is a Gaussian, $(\mu_t^{[i]}, \sigma_t^{[i]})$, where $\mu_t^{[i]}$ and $\sigma_t^{[i]}$ represent the mean and covariance of the Gaussian conditioned on the discrete sample $d_t^{[i]}$. In [4] this representation was used for diagnosing faults in conditional linear-Gaussian systems. When the faults being tracked have linear-Gaussian models, a Kalman filter may be used to efficiently estimate the posterior mixture of Gaussians. If the models are nonlinear numerical integration methods may be used for computing a posterior. An Unscented Kalman filter (UKF) [10] is used in [?]. Although the conditional density is no longer exact convergence guarantees can no longer be made, it is found to work well in practice.

The UKF (and variants known as derivative-free filter, linear-update filter, linear-regression filter or statistical linearization filter), is a recursive minimum mean square error estimator that often provides an improvement over the Extended Kalman filter (EKF) for nonlinear models. The EKF linearizes nonlinear functions using only the first order terms of the Taylor series expansion, which can be a poor approximation. Unlike the EKF, the UKF does not approximate the nonlinear process and measurement models. It uses the actual nonlinear models and instead approximates the distribution of the state random variable as a Gaussian. It is called a linear-update filter because the update to the state is a linear function of the measurement. The Gaussian approximation is specified using a minimal set of deterministically chosen sample points called sigma points. These sigma points are a sufficient statistic of the mean and covariance of the Gaussian approximation. Each sigma point is independently propagated through the actual nonlinear model to provide a posterior Gaussian approximation.

In this paper we use a Rao-Blackwellized particle filter to track a hybrid (discrete-continuous) state. A UKF is used for computing the conditional posterior of the continuous state. This is also called the Gaussian particle filter (GPF) [9].

4. Decision-Theoretic Sampling

Faults are low-probability, high-cost events. The classical particle filter generates particles proportional only to the posterior probability of an event. Monitoring a system to detect and identify faults based on a CPF therefore requires a very large number of particles and is computationally expensive.

Decision-theoretic sampling [27], [24] incorporates a model of cost when generating particles. This approach is motivated by the observation that the cost of not tracking hypotheses is related to risk. Not tracking a rare but risky state may have a high cost, whereas not tracking a rare but benign state may be irrelevant. Incorporating a cost model into particle filtering improves the tracking of states that are most critical to the performance of the robot. Decision-theoretic sampling generates particles by factoring in the cost. Since faults have a high cost, even though they have a low probability, a smaller number of particles than the CPF may be used to monitor these events because it ensures particles will be generated to represent them.

The cost function assigns a real-valued cost to states and control. The control selected, given the exact state, results in the minimum cost. The approximate nature of the particle representation may result in sub-optimal control and hence increased cost. The goal of decision-theoretic sampling is to generate particles that minimize the cumulative increase in cost due to the approximate particle representation. This is done by modifying the classical particle filter to generate particles in a risk-sensitive manner, where risk is defined as a function of the cost and is positive and finite. Given a suitable risk function $r(d)$, a decision-theoretic particle filter generates particles that are distributed according to the invariant distribution,

$$\gamma_t \; r(d_t) \; p(d_t, x_t \mid z^t) \tag{8}$$

where, γ_t is a normalization constant that ensures that equation (8) is a probability distribution. Instead of using just the posterior distribution to generate the particles, a product of the risk times the posterior is used. The choice of risk function is important.

Thrun et. al. in [24] present a method for obtaining this risk function via a Markov decision process (MDP) [19] that calculates the approximate future risk of decisions made in a particular state. The risk function is computed as the difference between the value function (computed using value-iteration) obtained from accurate tracking and the value-function obtained from inaccurate tracking. The value function for accurate tracking is computed using optimal control and the value function for inaccurate tracking is obtained using random control. The idea is that if the state-estimate is good, control is optimal and if it is poor control is sub-optimal. The risk-functions used in this paper are computed using this method.

The decision-theoretic filter is shown to improve state-estimation of low-probability high-risk states, but given numerous low probability high-risk states the performance of the decision-theoretic filter degrades. This is because not all the fault states may be sampled.

5. Limitations of Monte Carlo Smoothing and the Auxiliary Particle Filter

The problem with the decision-theoretic filter is that it is purely predictive. There are a large number of possible faults at a given instance, but only a small number of these have a high probability given the next measurement. In this section we discuss methods that generate samples by taking into account future measurements.

The estimation of the state of the system using future measurements is termed smoothing. It introduces a slight delay in providing state estimates (proportional to the number of timesteps over which smoothing is performed), but results in improved state estimates. Kalman smoothing is the best known, but may only be used in a filter that linearizes the state equations (such as the Kalman filter, and Extended Kalman Filter (EKF)). An improvement, Expectation Propagation [18], may be used with methods that linearize the measurement equations (such as EKF, UKF). Expectation-propagation re-linearizes the measurement equations until a globally-stable solution is reached. linear-Gaussian models, such as the Variable State (VSDF). Two-filter smoothing approaches [14] combines the results from running filtering forward and independently in reverse.

There are also methods in the literature for Monte Carlo smoothing. Forward particle filtering is performed up to time $t+1$. Particles at time t are then re-weighted and resampled, based on a sampled particle from time $t+1$ to obtain an updated estimate $s_{t|t+1}$ [5], [?]. There are also methods that re-weight particles at time t, using a density fitted to the particles at time $t+1$ [15]. Forward sampling is then performed using an updated proposal distribution $q(s_{t|t+1}) = p(s_{t+1} \mid s_{t|t+1})$, based on the smoothed state estimate.

Particles in a particle filter represent trajectories through state space. Resampling (although essential for reducing the variance of importance weights) reduces the number of independent trajectories, (or the effective sample size [3]). The aim of Monte Carlo smoothing is to ensure long independent trajectories, else the particle filter can often degenerate to tracking a single hypothesis. It does so by re-weighting particles to assign higher weight to particles that are more likely to survive, given future data. It is important to note that the backward pass in smoothing does not generate any new hypotheses. In addition, forward sampling from smoothed estimates is not guided by the smoothing process. Smoothing only improves the marginal distribution at time $t - L$, where L is the length of the fixed-lag over which smoothing is performed. In some applications such a hybrid-state estimation for fault diagnosis, where there are a large number of possible faults that may occur at any instance, states that resulted in long trajectories in the forward pass, may not do so subsequent forward passes using smoothed estimates.

The Auxiliary particle filter [?] is a method that uses future data to bias sampling to regions with high posterior likelihood. Subsequent variants, such as the Unscented particle filter [25], present alternate methods for estimating the posterior likelihood. The bias is compensated for by introducing an additional term in the importance weight.

As discussed in [26], the problem with using Auxiliary particle filter variants for hybrid state estimation is that it is computationally very expensive. A estimate of the posterior likelihood (such as with a UKF [26]), needs to be computed for every possible next discrete state transition. Measurement noise makes it difficult to find a reasonable estimate of a thereshold that may be used for selecting posterior states with high likelihood without an exhaustive search.

6. Decision-Theoretic Monte Carlo Smoothing

The idea behind decision-theoretic Monte Carlo smoothing (DTMS) is to maintain a time varying risk-function. Future measurements are used to smooth the risk-function to discourage sampling of states that are unlikely given future measurements. This approach is a compromise between the purely predictive decision-theoretic theoretic approach in section(RSPF)), which is efficient but inaccurate (for a large number of faults), and the auxiliary particle filter from the previous section, which is more accurate, but computationally expensive.

In DTMS, each particle in addition to the discrete-continuous state, $\{(d_t^{[1]}, x_t^{[1]}) \ldots (d_t^{[N]}, x_t^{[N]})\}$, and importance weight, $\{w_t^{[i]}\}$, also maintains an estimate of the risk-function, $r_t^[i]$, at time t. As discussed in section(), the initial risk function is computed for each discrete state $d_t, t = 0$, using a MDP, which assumes that complete information about the current state is available This is an efficient approximation. The process is actually a Partially Observable Markov Decision Process (POMDP) [22], [11], but computing a value function for large state spaces in a POMDP is computationally expensive. A POMDP is a model from operations research (OR) literature for decision making using information states (distributions over states) when complete information about the current state is not available. Smoothing the risk-function improves the approximation of the risk-function at time t, given a more accurate estimate of the probability distribution over states at time $t + 1$.

The aim of smoothing the risk function at time t is not merely to reinforce the discrete states that are already represented in the particle set at time $t+1$ since, but to encourage exploration of alternate high-risk states. Hence, instead of increasing the risk of transitions that have a high posterior likelihood we decrease the risk of transitions that have low posterior likelihood. A fixed lag history of particles is maintained. If a trajectory terminates at time $t + 1$, i.e. the particle is eliminated during re-sampling, the risk function at time t for that particle is updated to reduce the risk of state $t + 1$. Subsequent forward sampling therefore encourages sampling alternate high-risk states, which now have a relatively higher risk. This process is iterated until the trajectory survives re-sampling at time $t + 1$ or all possible transitions have been explored. The advantage of this approach is that extra computation is required only when the particle estimate starts to degenerate and not otherwise.

7. Experimental Results

Preliminary experimental results show an improvement in performance for small particle sets when using DTMS over decision-theoretic sampling without smoothing. For our experiments we used data from simulating stuck wheel faults on a six-wheel rocker-bogie rover in the Darwin2k [16] simulator. There are 7 discrete states in this simple example – a normal operation state and a state representing each of the stuck wheel faults. The continuous state tracks the $[x, y, \theta]$ position of the rover in global co-ordinates and measurements are noisy estimates of the position. Each experiment described below was run with the true state being each of the 7 discrete states. Except when specified otherwise, each experiment was repeated 1000 times to compute the standard error bars shown. The experiments measure error in terms of Kulback-Leibler divergence from the true distribution (computed using 10^6 particles with a classical filter. In the first experiment, shown in figure(1) as a dashed black line, a Rao-Blackwellized particle filter was used. The discrete states were sampled and the conditional continuous states were obtained using a UKF since the process is nonlinear (GPF). As expected the filter tacked the normal state well, but tracked fault states poorly since the probability of faults is low and no particles jumped to fault states. This experiment was only repeated 50 times since additional runs would not have made a visible difference to the graph. The reason the KL-divergence appears to increase slightly with more particles is because we use a Dirichlet 1 prior and the effect of the prior decreses with larger sample sizes.

In the second experiment, the Rao-Blackwellized filter above was enhanced to use decision-theoretic sampling. The results were good when the experiment was limited to one or two highly distinct fault states. When the full experiment was performed, performance degraded as shown in figure(1)as the dash-dot blue line. This is all faults have high risk and an incorrect fault (that explained the measurement relatively better than the normal state) was often confused when the correct fault state was not sampled.

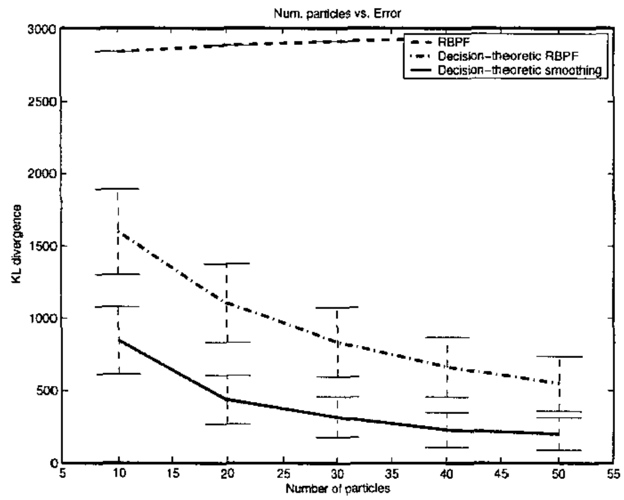

Figure 1. Number of particles vs. error in terms of KL-divergence for Rao-Blackwelized particle filter, Decision-theoretic RBPF, and Decision-theoretic RBPF with smoothed loss function

In the third experiment, shown in figure(1) as a solid red line, decision-theoretic smoothing of the risk function was performed with the GPF. Since the size of the discrete state space in this preliminary experiment is small, small particle sets were used to highlight the relative differences between the approaches.

8. Conclusions

We present a scalable method for monitoring hybrid state spaces and demonstrate the applicability of the approach for fault diagnosis on rovers. This approach is valid even when the process is nonlinear. In addition it is also capable of monitoring an arbitrarily large number of faults. It is based on particle filters and has all the advantages of particle filters which include the ability to represent non-parametric posteriors, nonlinear processes, and easy extension to an anytime approach. The main drawback of particle filters is that a large number of samples may be needed for reasonable approximations. This is because small particle sets often degenerate to tracking a single trajectory. Our approach uses decision-theoretic Monte Carlo smoothing to ensure that multiple independent trajectories are maintained and focuses particles in regions of the state space that are expected to be high-risk at that time. This requires a comparatively smaller number of samples for performance comparable to other state-of-the-art methods. Experimental results demonstrate the improvement in performance obtained with this approach.

References

[1] D. Avitzour. A stochastic simulation bayesian approach to multitarget tracking. In *IEEE Transaction on Aerospace and Electronic systems*, 1995.

[2] Y. Bar-Shalom and T. R. Fortmann. *Tracking and Data Association.* Academic Press, 1988.

[3] J. Carpenter, P. Clifford, and P. Fearnhead. An improved particle filter for non-linear problems. In *EE proceedings - Radar, Sonar and Navigation*, 1999.

[4] N. de Freitas. Rao-blackwellised particle filtering for fault diagnosis. In *IEEE Aerospace*, 2002.

[5] A. Doucet, S. J. Godsill, and M. West. Monte carlo filtering and smoothing with application to time-varying spectral estimation. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2000.

[6] A. Doucet, N. Gordon, and V. Krishnamurthy. Particle filters for state estimation of jump markov linear systems. Technical report, Cambridge University, 1999.

[7] N.J. Gordon, D.J Salmond, and A.F.M Smith. Novel approach to nonliner/non gaussian bayesian state estimation. In *IEE Proceedings-F*, volume 140, pages 107–113, 1993.

[8] N.J. Gordon, D.J Salmond, and A.F.M Smith. Novel approach to nonliner/non-Gaussian Bayesian state estimation. In *IEE Proceedings-F*, volume 140(2), pages 107–113, 1993.

[9] F. Hutter and R. Dearden. The gaussian particle filter for diagnosis of non-linear systems. In *Proceedings of the 5th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, 2003.

[10] S. J. Julier and J. K. Uhlmann. A general method of approximating nonlinear transformations of probability distributions. Technical report, Robotics Research Group, Department of Engineering Science, University of Oxford, 1996.

[11] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra. Planning and acting in partially observable stochastic domains. *Artificial Intelligence*, 101, 1998.

[12] K. Kanazawa, D. Koller, and S. Russell. Stochastic simulation algorithms for dynamic probabilistic networks. pages 346–351, 1995.

[13] G. Kitagawa. A monte carlo filtering and smoothing method for non-gaussian nonlinear state space models. In *Proceedings of the 2nd U.S.-Japan Joint Seminar on Statistical Time Series Analysis*, 1993.

[14] G. Kitagawa. The two-filter formula for smoothing and an implementation of the gaussian-sum smoother. In *Annals Institute of Statistical Mathematics*, 1994.

[15] D. Koller, U. Lerner, and D. Angelov. A general algorithm for approximate inference and its application to hybrid bayes nets. In *imate Inference and its Application to Hybrid Bayes Nets D. Koller, U. Lerner, and D. Angelov, In Proceedings of the 15th Annual Conference on Uncertainty in AI (UAI).*

[16] C. Leger. *Darwin2K: An Evolutionary Approach to Automated Design for Robotics.* Kluwer Academic Publishers, 2000.

[17] R. Chen S. Liu. Mixture kalman filters. In *Journal of Royal Statistical Society*, 2000.

[18] T. P. Minka. *A family of algorithms for approximate Bayesian inference*. PhD thesis, MIT, 2001.

[19] M. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley and Sons Incorporated, 1994.

[20] D.B. Rubin. *Using the SIR algorithm to simulate posterior distributions*. Oxford University Press, 1988.

[21] R. Y. Rubinstein. Simulation and Monte Carlo method. In *John Wiley and Sons*, 1981.

[22] E. J. Sondik. *The optimal control of partially observable markov Decision Processes*. PhD thesis, Stanford, 1971.

[23] M.A. Tanner. *Tools for Statistical Inference*. Springer Velag, 1996.

[24] S. Thrun, J. Langford, and V. Verma. Risk sensitive particle filters. In *Neural Information Processing Systems (NIPS)*, Dec 2001.

[25] R. van der Merwe, N. de Freitas, Arnaud Doucet, and Eric Wan. The unscented particle filter. Technical report, Cambridge University Engineering Department, Cambridge, England, August 2000.

[26] V. Verma, G. Gordon, and R. Simmons. Efficient monitoring for planetary rovers. In *International Symposium on Artificial Intelligence and Robotics in Space*, 2003.

[27] V. Verma, J. Langford, and R. Simmons. Nonparametric fault identification for space rovers. In *International Symposium on Artificial Intelligence and Robotics in Space (iSAIRAS)*, June 2001.