NONLINEAR FUSION OF MULTIPLE SENSORS WITH MISSING DATA

Alon Shalev Housfater, Xiao-Ping Zhang

Department of Electrical & Computer Engineering 350 Victoria Street Ryerson University, Toronto, Canada E-mail: {ashalev,xzhang}@ee.ryerson.ca

ABSTRACT

We introduce a new algorithm, multiple imputation particle filter, to solve the problem of data fusion with missing data in nonlinear state space models. The new algorithm is then applied to the problem of fusing observations by multiple asynchronous radars. Simulated data is used demonstrate the effectiveness and performance of the fusing algorithm.

1. INTRODUCTION

Handling missing data has always been a part of statistical analysis, which frequently occurs in various surveys and experiments [1], [2]. Missing data issues also occur in many signal processing applications [3], one important example is data fusion; most realizations of sensor networks operate in an asynchronous manner due to the difficulty in synchronizing observations across multiple independent sensors. For example, in a radar network the radars might be many miles apart, thus making it difficult to synchronize their observations [4]. There are several standard approaches in dealing with missing data; one is to employ linear prediction techniques to align the data to the instance of missing data, others include formulating a Kalman filter with a time-varying transition matrix that accounts for the missing observations [4]. Also, one can apply the EM algorithm to resolve the missing data [2]. These algorithms either assume a linear system or ignore the state dynamics while the underlying observed system is often nonlinear; thus, available methods are not applicable for nonlinear systems or fail to utilize the system's nonlinearity. Conversely, a well known estimation method in the context of nonlinear systems is the particle filtering algorithm [5]; however, particle filtering does not incorporate the missing data behavior into the estimation. Therefore, we need to confront the problem of performing fusion on a nonlinear system in the presence of missing data.

In this paper we present a new method of fusing multiple observations in a nonlinear system while accounting for missing data. This is done by combining particle filtering with a multiple imputations technique. We replace the missing data with imputations, i.e. randomly drawn values, to form multiYifeng Zhou

Defence R&D Canada - Ottawa 3701 Carling Avenue Ottawa, Ontario K1A 0Z4 E-mail: yifeng.zhou@drdc-rddc.gc.ca

ple complete data sets. Each data set is then particle filtered, the results of those multiple particle filtering operations are then combined as a weighted sum. We apply this new approach to data fusion in an asynchronous radar network by regarding the asynchronous behavior as missing data at each time instance.

2. PROBLEM FORMULATION

Consider a time-varying stochastic system with X_n denoting the state at time instance n, It is assumed that X_n behaves according to a non-homogenous Markov chain with transition probabilities as described by the recurrence equation

$$X_n = \varphi_n(X_{n-1}, W_n), \tag{1}$$

where W_n is random evolution noise, assumed to be independent identically distributed stochastic process and φ_n is the non-homogenous evolution transformation. Also, let the system be observed by K sensors where the measurement is modeled

$$U_n = \begin{pmatrix} \psi_1(X_n, V_{n,1}) \\ \vdots \\ \psi_K(X_n, V_{n,K}) \end{pmatrix}, \qquad (2)$$

where U_n denotes the noisy observation of the state X_n such that $V_{n,k}$ is an *i.i.d* noise process and ψ_k is the measurement transformation for sensor k, respectively. We denote the k-entry of the K-dimensional vector U_n as U_n^k .

In order to handle the missing data, we define a random indicator variable $R_{n,k}$ for each sensor k at time instance n, this variable indicates if an observation is available or not

$$R_{n,k} = \begin{cases} 0 & \text{observation is available from sensor } k \text{ at time } n \\ 1 & \text{otherwise} \end{cases}$$

Next, we define the missing information set Z_n as the collection of U_n^k for all k = 1, ..., K such that $R_{n,k} = 1$. Similarly, the available information set Y_n is the collection of U_n^k for all k = 1, ..., K such that $R_{n,k} = 0$.

It is assumed that the missing data mechanism is independent of the missing observations given the available observations, this can be written as

$$P(R_{n,k}|Z_n, Y_n) = P(R_{n,k}|Y_n) \text{ for all } k, n.$$
(3)

This standard statistical assumption is known as Missing at Random (MAR) [1]. Our objective is to obtain the posteriori probability density function of the state given all past and present observations. We write this density function as $p(X_n|Y_{0:n})$ where $Y_{0:n}$ denotes all observations from the initial time instance to time instance n.

3. PARTICLE FILTERS AND MULTIPLE IMPUTATIONS

3.1. Particle Filters for Data Fusion with No Missing Data

As stated in the problem formulation, we wish to fuse K observations in a Bayesian sense, i.e., obtain the probability density function $p(X_n|Y_{0:n})$. We use the particle filtering approximation since a closed-form solution does not exist due to the required integration [5]. The particle filter uses a set of N simulated states, or particles, X_n^i and their corresponding weights w_n^i for each time instance n where i indexes the particles. These particles, $\{w_n^i, X_n^i\}_{i=1}^N$, discretize the density $p(X_n|Y_{0:n})$

$$p(X_n|Y_{0:n}) \approx \sum_{i=1}^N w_n^i \delta(X_n - X_n^i),$$
 (4)

where δ denotes the dirac delta function. The particles are obtained in a recursive manner, at each time instance n the particles of time instance n-1 are used in sampling from an importance function $\pi(X_n|X_{0:n}^i, Y_{0:n})$

$$X_n^i \sim \pi(X_n | X_{0:n}^i, Y_{0:n})$$
 for $1 \le i \le N$,

the particle weighting coefficient w_n^i is obtained by the calculation

$$w_n^i = w_{n-1}^i \frac{p(Y_n | X_n^i) p(X_n^i | X_{n-1}^i)}{\pi(X_n^i | X_{0:n}^i, Y_{0:n})}.$$

Note that in an application with missing data, the standard particle filtering algorithm does not incorporate the response information $R_{n,k}$ when performing the procedure nor is it guaranteed to be stable. Thus, in the presence of significant missing data mechanisms, the particle filter could diverge or its performance be severely degraded.

3.2. Handling Missing Data with Multiple Imputations

Let $R_n = [R_{n,1}, \ldots, R_{n,K}]^T$ be the K-dimensional indicator vector for the response of the sensors at time instance n. We

compute the probability density $p(X_n|Y_n, R_n)$ which can be written

$$p(X_n|Y_n, R_n) = \int p(X_n|Y_n, Z_n, R_n) \ p(Z_n|Y_n, R_n) dZ_n,$$
(5)

using the condition of MAR as defined in equation (3), it can be shown [1] that equation (5) reduces to

$$p(X_n|Y_n, R_n) = \int p(X_n|Y_n, Z_n) \ p(Z_n|Y_n) dZ_n.$$

Thus, we can approximately compute the density $p(X_n|Y_n)$ by the Monte Carlo approximation

$$p(X_n|Y_n) = \lim_{M \to \infty} \frac{1}{M} \sum_{j=1}^M p(X_n|Y_n, Z_n^j),$$

where $Z_n^j \sim p(Z_n|Y_n)$ are the multiple imputations.

Notice that multiple imputations does not use the past observations and the state transition equation in estimating the density $p(X_n|Y_n)$. This is significant since many real world problems are well modeled by a Markov structure, which does use past values to determine the present ones. Thus, in such application, we expect that the performance of the multiple imputations be degraded.

4. NEW MULTIPLE IMPUTATION PARTICLE FILTERS

We present a new algorithm that resolves the mentioned deficiencies in the particle filtering and multiple imputations algorithms. This algorithm performs the fusion using both the state and observation dynamics while accounting for the missing data.

4.1. Algorithm for Multiple Imputation Particle Filter

First, we draw the random observations, i.e. imputations, from a proposal function ϕ

$$Z_n^j \sim \phi(Z_n | Y_{0:n})$$
 for $j = 1, ..., M$,

where each imputation Z_n^j has an associated weight p_n^j such that $\sum_j p_n^j = 1$. Note that we can write the filtering probability density $p(X_n|Y_{0:n})$ [3]

$$p(X_n|Y_{0:n}) = \int p(X_n|U_{0:n-1}, Y_n) p(Z_n|Y_{0:n}) dZ_n.$$
 (6)

By forming the imputed data sets $U_n^j = \{Z_n^j, Y_n\}$ and taking a Monte Carlo approximation, we can write equation (6) as

$$p(X_n|Y_{0:n}) \simeq \sum_{j=1}^{M} p_n^j p(X_n|U_{0:n-1}, U_n^j).$$
(7)

Next, the algorithm performs particle filtering on each data set U_n^j to obtain an approximation as in equation (4)

$$p(X_n|U_{0:n-1}, U_n^j) \approx \sum_{i=1}^N w_n^{j,i} \delta(X_n - X_n^{j,i}),$$
 (8)

where $X_n^{j,i}$ is the *i* particle for the *j* imputation at time instance *n* and $w_n^{j,i}$ is its weight. Finally, the algorithm combines the multiple particle filtering results by substituting equation (8) into equation (7) to obtain an approximation of the desired density

$$p(X_n|Y_{0:n}) \simeq \sum_{j=1}^M \sum_{i=1}^N p_n^j w_n^{j,i} \delta(X_n - X_n^{j,i}).$$

4.2. New Particle Based Imputation Procedure

In the discussion above we used the proposal function ϕ to draw imputations but did not specify the function or how to go about drawing the imputations. Usually, the imputation process is implemented using Markov Chain Monte Carlo (MCMC) methods [2]. However, in many applications, MCMC techniques are not attractive since they are too computationally intensive to be performed at each time instance. Therefore, we suggest a new approach to performing the imputation process. The imputing probability density $p(Z_n|Y_{0:n})$ can be written

$$p(Z_n|Y_{0:n}) = \int p(Z_n|X_n)p(X_n|Y_{0:n})dX_n.$$
 (9)

Note that the filtering density $p(X_n|Y_{0:n})$ appears inside the integral, since we do not know this density, we are unable to sample directly from $p(Z_n|Y_{0:n})$. However, equation (9) suggests the following approximation. First, find a discrete density $\tilde{p}(X_n|Y_{0:n})$ that approximate the true filtering density well. Then using relationship (9), one can obtain the discrete density $\tilde{p}(Z_n|Y_{0:n})$, which will approximate the desired density $p(Z_n|Y_{0:n})$. We write the approximate filtering density $\tilde{p}(X_n|Y_{0:n})$ using a particle approximation

$$\widetilde{p}(X_n|Y_{0:n}) = \sum_{i=1}^{\widetilde{N}} \widetilde{w}_n^i \delta(X_n - \widetilde{X}_{n,i}),$$

where $p(X_n|Y_{0:n})$ is approximated by the set $\{\widetilde{w}_n^i, \widetilde{X}_n^i\}_{i=1}^N$ as in equation (4). Substituting this approximation into equation (9) we have

$$p(Z_n|Y_{0:n}) \approx \sum_{i=1}^{\widetilde{N}} \widetilde{w}_n^i p(Z_n|\widetilde{X}_{n,i}).$$

We assume $p(Z_n | \tilde{X}_{n,i}) = p(Y_n | \tilde{X}_{n,i})$ then this mixture of densities is known and can be sampled from in a straightforward procedure. Note that it is natural to extend this procedure to an iterative procedure where $\tilde{p}(X_n | Y_{0:n})$ is the output of the previous iteration's multiple imputations particle filter.

Briefly, an outstanding issue in this discussion is that of convergence. It turns out that the algorithm does converge in a certain sense; using ideas from [6], we have obtained almost sure weak convergence results under some regularity conditions.

5. APPLICATION - ASYNCHRONOUS RADAR NETWORK

Many radar systems are implemented using a radar network in order to cover a large area and increase reliability, these radars transmit observations asynchronously due to the large distances between the radars. We begin by formulating the radar network's state space model and then proceed to show simulations of the performance of the multiple imputation particle filter compared against the EM algorithm.

5.1. Multiple Radars Model

Consider K radars measuring a common target. The target motion is described by a simple near constant velocity state model [4]

$$X_{n+1} = \Phi_{n+1}X_n + \Gamma_{n+1}V_n,$$

$$\Phi_{n+1} = \begin{pmatrix} 1 & \Delta t_{n+1} & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & \Delta t_{n+1}\\ 0 & 0 & 0 & 1 \end{pmatrix}$$

, and

$$\Gamma_{n+1} = \begin{pmatrix} \Delta t_{n+1}^2/2 & 0\\ \Delta t_{n+1} & 0\\ 0 & \Delta t_{n+1}^2/2\\ 0 & \Delta t_{n+1} \end{pmatrix}.$$

The state vector X_n is defined as $X_n = [x(n), v_x(n), y(n), v_y(n)]^T$, and the system noise $V_n \sim \mathcal{N}(0, \Sigma_v)$ is assumed to be a Gaussian *i.i.d.* process. Let φ denote the cartesian to polar transformation

$$\psi(X_n) = \begin{pmatrix} \sqrt{x(n)^2 + y(n)^2} \\ \tan^{-1}\left(\frac{y}{x}\right) \end{pmatrix}.$$

Also, The radar observation model is given by

$$Y_n^j = \varphi(X_n) + W_n^j,$$

where $W_n \sim \mathcal{N}(0, \Sigma_w^j)$ denotes the system noise, which is assumed to be a Gaussian *i.i.d.* process. It is also assumed that the system and observation noises are mutually independent.

5.2. Simulations

We simulate two radar network configurations: two asynchronous radars and three asynchronous radars. We compare the performance of the standard EM fusion [2] and the multiple imputation particle filter. The two radar network is simulated with a 20 time units track where at each time instance, the observations are repeated 30 times to obtain average performance results. The three radar network is simulated with a 12 time units track where at each time instance, the observations are repeated 10 times. Both simulations are implemented with a particle filter that uses 250 particles for 40 imputations. The performance results for the two radar network are shown in Figure 1, we can see significant performance improvement by the multiple imputation particle filter as compared to the EM algorithm. We can also see some improvement by the multiple imputation particle filter for the three radar network, shown in Figure 2. However, the improvement is not as significant as for the two radar network. This discrepancy can be explained as follows, we expect that the more significant the missing data behavior of the system, the more performance gains we will realize from the multiple imputation particle filter, which is designed to deal with missing data, as compared to the EM algorithm. Due to the nature of the simulation, the ratio of missing to available information for the simulated two radar network is %50 while the missing information ratio for the simulated three radar network is almost always %25. Then, as expected, the multiple imputation particle filter shows significant performance gain for the more aggressive missing data pattern of the two radar network.



Fig. 1. Multiple Imputation Particle Filter Performance - Two Radars

6. CONCLUSION

In this paper, a novel multiple imputation particle filter approach was presented for fusing multiple sensor measurements in the presence of missing data. The algorithm is suitable for nonlinear systems in real time applications such as asynchronous sensor networks. The algorithm draws imputations using a new particle based method, it then combines the multi-



Fig. 2. Multiple Imputation Particle Filter Performance - Three Radars

ple imputations with a particle filtering technique. Simulated data is used to demonstrate the effectiveness and performance of the proposed algorithm. The results showed that the proposed algorithm outperformed the widely used standard EM algorithm.

7. REFERENCES

- D. B. Rubin, Multiple Imputation for Nonresponse in Surveys, Wiley, 1987
- [2] J. L. Schafer and J. Schafer, Analysis of Incomplete Multivariate Data, Chapman & Hall CRC, 1997.
- [3] J. S. Liu, A. Kong and W. H. Wong, "Sequential Imputations and Bayesian Missing Data Problems," *Journal* of the American Statistical Association, vol. 89, no. 425, 1994.
- [4] Y. Zhou, "A Kalman Filter Based Registration Approach for Multiple Asynchronous Sensors," Tech. Rep., Defense R&D Canada - Ottawa, 11 2003, http://cradpdf.drdc.gc.ca/PDFS/unc17/p520870.pdf
- [5] A. Doucet, S. Godsill and C. Andrieu, "On Sequential Monte Carlo Sampling Methods for Bayesian Filtering," *Statistics and Computing*, vol. 10, no. 2, pp. 197-208, 2000.
- [6] D. Crisan and A. Doucet, "A Survey of Convergence Results on Particle Filtering Methods for Practitioners," *IEEE. Trans. Signal Proc.*, vol. 50, no. 3, pp. 736-746, 2002.