Batch Maximum Likelihood (ML) and Maximum A Posteriori (MAP) Estimation With Process Noise for Tracking Applications

A. B. Poore, B. J. Slocumb, B. J. Suchomel, F. H. Obermeyer, S. M. Herman, S. M. Gadaleta

Numerica Corporation, PO Box 271246, Fort Collins, CO 80527-1246

ABSTRACT

Batch *maximum likelihood* (ML) and *maximum a posteriori* (MAP) estimation with process noise is now more than thirty-five years old, and its use in multiple target tracking has long been considered to be too computationally intensive for real-time applications. While this may still be true for general usage, it is ideally suited for special needs such as bias estimation, track initiation and spawning, long-term prediction of track states, and state estimation during periods of rapidly changing target dynamics. In this paper, we examine the batch estimator formulation for several cases: nonlinear and linear models, with and without a prior state estimate (MAP vs. ML), and with and without process noise. For the nonlinear case, we show that a single pass of an extended Kalman smoother-filter over the data corresponds to a Gauss-Newton step of the corresponding nonlinear least-squares problem. Even the iterated extended Kalman filter can be viewed within this framework. For the linear case, we develop a compact least squares solution that can incorporate process noise and the prior state when available. With these new views on the batch approach, one may reconsider its usage in tracking because it provides a robust framework for the solution of the aforementioned problems. Finally, we provide some examples comparing linear batch initiation with and without process noise to show the value of the new approach.

Keywords: Batch ML Estimation, Batch MAP Estimation, Nonlinear Least Squares, Track Initiation and Spawning, Extrapolation

1. INTRODUCTION

Batch *maximum likelihood* (ML) and *maximum a posteriori* (MAP) estimation with process noise is now more than thirty-five years old,[1,2] but its use in multiple target tracking has long been considered to be too computationally intensive for real-time applications. While this may still be true, significant advances over the last twenty years in nonlinear least-squares, nonlinear optimization, corresponding linear algebra techniques, and computing power leads one to re-examine its use for *special problems* in multiple target tracking.

While "batch ML/MAP estimation with process noise" is not intended to replace the extended Kalman filter (EKF), the unscented Kalman filter (UKF), the interacting multiple model (IMM) algorithm, or variable-structure interacting multiple model (VSIMM) filtering techniques in general, it may improve these techniques for estimation problems that are highly nonlinear relative to a given measurement update rate or signal-to-noise ratio. (The performance of the extended or iterated extended Kalman filter (or the unscented Kalman filter) may degrade significantly in such cases.) Some potential areas in which better optimization and estimation may improve performance significantly are (1) estimation of sensor and navigation biases, (2) long-term state prediction, (3) track initiation and spawning, and (4) estimation during periods of rapidly changing dynamics (i.e., maneuvers). Also, in multiple frame assignment (MFA) or multiple hypothesis tracking (MHT) approaches, one employs a moving window over which it is natural to use batch ML and MAP estimation. Thus, even though this approach is not intended to replace extended Kalman filtering or interacting multiple models (IMM), it may be an efficient method for both monitoring the success of these methods and providing a robust numerical technique to correct some of their difficulties.

This paper is organized as follows. In Section 2, the "batch ML/MAP estimator with process noise" is reviewed for the nonlinear problem. First, the MAP and the ML formulations are given, then the relationship to the iterated Kalman filter and to the iterated Kalman smoother-filter is explained. Section 3 addresses the "batch ML/MAP estimator with process noise" for the linear problem. A new compact formulation is given for the batch estimator that handles both cases with and without a prior state estimate.

Further author information: Send correspondence to Aubrey B Poore via email (abpoore@numerica.us) or telephone (970-419-8343).

Report Documentation Page

Form Approved
OMB No. 0704-0188

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE **2003**	2. REPORT TYPE	3. DATES COVERED **00-00-2003 to 00-00-2003**
4. TITLE AND SUBTITLE **Batch Maximum Likelihood (ML) and Maximum A posteriori (MAP) Estimation with Process Noise for Tracking Applications**		5a. CONTRACT NUMBER 5b. GRANT NUMBER 5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)		5d. PROJECT NUMBER 5e. TASK NUMBER 5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Numerica Corporation,PO Box 241246,Fort Collins,CO,80527-1246**		8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)		10. SPONSOR/MONITOR'S ACRONYM(S) 11. SPONSOR/MONITOR'S REPORT NUMBER(S)
12. DISTRIBUTION/AVAILABILITY STATEMENT **Approved for public release; distribution unlimited**		
13. SUPPLEMENTARY NOTES **The original document contains color images.**		
14. ABSTRACT **see report**		
15. SUBJECT TERMS		

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**		**12**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

2. ML AND MAP: THE DISCRETE NONLINEAR PROBLEM

It is known that, with Gaussian a priori statistics, the maximum *a posteriori* estimate is equivalent to an appropriate least-squares curve fit, using the inverses of the plant- and measurement-noise covariances as weighting matrices. This section shows a development of the appropriate cost function and the least squares problem. Such a formulation can also be posed as a discrete or continuous nonlinear two-point boundary-value problem. The development in this section follows closely that in the book by Jazwinski[1] and the book by Sage and Melsa.[2]

2.1. The model and statistical formulation

The discrete time state and observation models are given by

$$x_{k+1} = f_k(x_k) + \Gamma_k w_k, \tag{1}$$

$$z_k = h_k(x_k) + v_k, \tag{2}$$

where

$x_k = n$-dimensional state vector,

$f_k(x_k) = n$-dimensional vector-valued function,

$\Gamma_k = n \times r$ matrix,

$w_k = r$-dimensional plant-noise vector.

$z_k = m$-dimensional observation vector,

$h_k(x_k) = m$-dimensional vector-valued function,

$v_k = m$-dimensional observation-noise vector,

For the discrete-estimation model, w_k and v_k are assumed to be independent zero-mean Gauss-Markov white sequences such that

$$E\{w_k w_j^T\} = Q_k \delta_{k-j},\ E\{v_k v_j^T\} = R_k \delta_{k-j},\ E\{x_0 w_k^T\} = 0,\ E\{x_0 v_k^T\} = 0,\ x_0 \sim \mathcal{N}(\overline{x}_0, P_0),$$

where δ_{k-j} is the Kronecker delta function, and Q_k and R_k are nonnegative definite $r \times r$ and $m \times m$ covariance matrices, respectively. The sequences $x_0, x_1, ..., x_N$ and $z_1, z_2, ..., z_N$ are denoted by X_N and Z_N, respectively. We further assume that $p(x_0)$ is known and is normal with mean $\overline{x}_0$ and covariance P_0. We shall assume that all functions are sufficiently smooth so that up to two derivatives exist and are continuous. Also, it is typically assumed that the time of the prior is such that $t_0 < t_1$.

The best estimate of x throughout an interval will, in general, depend on the criteria used to determine the best estimate. Here the term "best estimate" denotes that estimate derived from maximizing the conditional probability function $p(X \mid Z)$ as a function of x throughout that interval, and is known as the maximum a posteriori estimator. The following derivation is adapted from that found in the book by Sage and Melsa[3] and is repeated here for completeness.

Bayes' rule applied to $p(X_N \mid Z_N)$ yields

$$p(X_N \mid Z_N) = \frac{p(Z_N \mid X_N)\, p(X_N)}{p(Z_N)}. \tag{3}$$

If v_k is Gaussian and x_k is known, Eq. (2) implies $p(z_k \mid x_k)$ is Gaussian; so that, if X_N is given,

$$p(Z_N \mid X_N) = \prod_{k=1}^{N} \frac{\exp\left\{-\frac{1}{2}(z_k - h_k(x_k))^T R_k^{-1}(z_k - h_k(x_k))\right\}}{(2\pi)^{m/2} \det(R_k)^{\frac{1}{2}}}. \tag{4}$$

Using the chain rule for probabilities, $p(\alpha, \beta) = p(\alpha \mid \beta)\, p(\beta)$, results in

$$p(X_N) = p(x_N \mid X_{N-1})\, p(x_{N-1} \mid X_{N-2}) \cdots p(x_1 \mid x_0)\, p(x_0). \tag{5}$$

Since w_k is a white Gauss-Markov sequence, x_k is Markov and $p(x_k \mid X_{k-1}) = p(x_k \mid x_{k-1})$. Thus $p(X_N)$ is composed of Gaussian terms, and

$$p(X_N) = p(x_0) \prod_{k=1}^{N} p(x_k \mid x_{k-1}), \tag{6}$$

where $p(x_k \mid x_{k-1})$ is Gaussian and, from equation (1), has mean $f_{k-1}(x_{k-1})$ and covariance

$$\Omega_{k-1} = \Gamma_{k-1}\, Q_{k-1}\, \Gamma_{k-1}^T.$$

$p(Z_N)$ contains no terms in x_k since Z_N is the known conditioning variable. Thus $p(Z_N)$ can be considered a normalizing constant with respect to the intended maximization. Analytically,

$$p(x_k \mid x_{k-1}) = \frac{\exp\left\{-\frac{1}{2}(x_k - f_{k-1}(x_{k-1}))^T \Omega_{k-1}^{-1}(x_k - f_{k-1}(x_{k-1}))\right\}}{(2\pi)^{n/2}\det(\Omega_{k-1})^{\frac{1}{2}}}. \tag{7}$$

Summarizing, we have

$$\begin{aligned}
p(X_N \mid Z_N) &= \frac{p(Z_N \mid X_N)\, p(X_N)}{p(Z_N)} = \frac{p(x_0)}{p(Z_N)} \prod_{k=1}^{N} p(z_k \mid x_k) p(x_k \mid x_{k-1}) \\
&= \frac{p(x_0)}{p(Z_N)} \prod_{k=1}^{N} \frac{\exp\left\{-\frac{1}{2}(z_k - h_k(x_k))^T R_k^{-1}(z_k - h_k(x_k))\right\}}{(2\pi)^{m/2}\det(R_k)^{\frac{1}{2}}} \times \\
&\quad \frac{\exp\left\{-\frac{1}{2}(x_k - f_{k-1}(x_{k-1}))^T \Omega_{k-1}^{-1}(x_k - f_{k-1}(x_{k-1}))\right\}}{(2\pi)^{n/2}\det(\Omega_{k-1})^{\frac{1}{2}}} \\
&= A \exp\left(-\frac{1}{2}\|x_0 - \overline{x}_0\|^2_{P_0^{-1}} - \frac{1}{2}\sum_{k=1}^{N}\|z_k - h_k(x_k)\|^2_{R_k^{-1}} - \frac{1}{2}\sum_{k=1}^{N}\|x_k - f_{k-1}(x_{k-1})\|^2_{\Omega_{k-1}^{-1}}\right)
\end{aligned}$$

where $A^{-1} = p(Z_N)\prod_{k=1}^{N}(2\pi)^{m/2}\det(R_k)^{\frac{1}{2}}(2\pi)^{n/2}\det(\Omega_{k-1})^{\frac{1}{2}}$. Assuming that Γ_{k-1} (and thus Ω_{k-1}) does not depend on the state, the problem of maximizing $p(X_N \mid Z_N)$ is equivalent to

$$\text{Minimize } J = \frac{1}{2}\|x_0 - \overline{x}_0\|^2_{P_0^{-1}} + \frac{1}{2}\sum_{k=1}^{N}\|z_k - h_k(x_k)\|^2_{R_k^{-1}} + \frac{1}{2}\sum_{k=1}^{N}\|x_k - f_{k-1}(x_{k-1})\|^2_{\Omega_{k-1}^{-1}}. \tag{8}$$

We further examine this expression for the two cases of (with and without) the prior in the following subsections.

2.1.1. A second formulation of the MAP estimation problem

The nonlinear least squares formulation for the MAP estimate is that shown in (8). An equivalent formulation, established using the Moore-Penrose inverse of Ω_{k-1}, is

$$\text{Minimize } J = \frac{1}{2}\mu_0 P_0^{-1}\mu_0 + \frac{1}{2}\sum_{k=1}^{N} v_k^T R_k^{-1} v_k + \frac{1}{2}\sum_{k=1}^{N} w_{k-1}^T Q_{k-1}^{-1} w_{k-1} \tag{9}$$

$$\begin{aligned}
\text{Subject to } \; x_k &= f_{k-1}(x_{k-1}) + \Gamma_{k-1} w_{k-1} \quad k = 1, \ldots, N \\
z_k &= h_k(x_k) + v_k \quad k = 1, \ldots, N \\
x_0 &= \overline{x}_0 + \mu_0.
\end{aligned}$$

REMARK 1. *The formulation in (9) is more robust than the one in (8) because when measurements are closely spaced in time, $\Omega_k \to 0$, which leads to an ill-conditioned optimization problem.*

As formulated, the prior is assumed to be at a time prior to the first measurement. The formulation is easily modified to account for a prior at time $t_0 = t_1$. In this case, $x_1 = f_0(x_0) = x_0$, and the formulation is

$$\text{Minimize } J = \frac{1}{2}\|x_1 - \overline{x}_1\|^2_{P_1^{-1}} + \frac{1}{2}\sum_{k=1}^{N}\|z_k - h_k(x_k)\|^2_{R_k^{-1}} + \frac{1}{2}\sum_{k=2}^{N}\|x_k - f_{k-1}(x_{k-1})\|^2_{\Omega_{k-1}^{-1}}. \tag{10}$$

2.1.2. A second formulation of the ML estimation problem

A second formulation in which there is no prior can be extracted from (10) by assuming that the density $p(x_0)$ on x_0 is a uniform distribution, i.e., a noninformative prior. This leads to the problem

$$\text{Minimize } J = \frac{1}{2}\sum_{k=1}^{N} \|z_k - h_k(x_k)\|^2_{R_k^{-1}} + \frac{1}{2}\sum_{k=2}^{N} \|x_k - f_{k-1}(x_{k-1})\|^2_{\Omega_{k-1}^{-1}}, \tag{11}$$

which has the equivalent formulation,

$$\begin{aligned} \text{Minimize } J &= \frac{1}{2}\sum_{k=1}^{N} v_k^T R_k^{-1} v_k + \frac{1}{2}\sum_{k=2}^{N} w_{k-1}^T Q_{k-1}^{-1} w_{k-1} \\ \text{Subject to } x_k &= f_{k-1}(x_{k-1}) + \Gamma_{k-1} w_{k-1} \quad k = 1, \ldots, N \\ z_k &= h_k(x_k) + v_k \quad k = 1, \ldots, N \\ x_0 &= \overline{x}_0 + \mu_0. \end{aligned}$$

2.2. Relation to the iterated extended Kalman filter and filter-smoother

The above formulation uses Gaussian noise assumptions and a ML/MAP estimation formulation to compute the "modal trajectory." In this section, an explanation of the relationship between this formulation and the iterated extended Kalman filter (IEKF) and iterated extended Kalman filter-smoother is provided.

2.2.1. The iterated extended Kalman filter

In 1993, Bell and Cathey[4] showed that the iterated extended Kalman filter (IEKF) can be viewed as a sequence of Gauss-Newton steps for a nonlinear least squares problem. Here is the explanation. Given $x_k \sim \mathcal{N}(x_{k|k-1}, P_{k|k-1})$ and $z_k = h_k(x_k) + v_k$, then each iteration of the estimate of x_k via an IEKF is obtained as a Gauss-Newton step of the nonlinear least squares problem

$$\text{Minimize } J = \frac{1}{2}\|x_k - x_{k|k-1}\|^2_{P_{k|k-1}^{-1}} + \frac{1}{2}\|z_k - h_k(x_k)\|^2_{R_k^{-1}} \tag{12}$$

or equivalently

$$\text{Minimize } J(x_k) = \frac{1}{2}\left\| \begin{pmatrix} x_k - x_{k|k-1} \\ h_k(x_k) - z_k \end{pmatrix} \right\|^2_{W_k^{-1}},$$

where $W_k = \text{diag}(P_{k|k-1}, R_k)$, i.e., the IEKF, when convergent, computes a minimum of this nonlinear least squares problem.

2.2.2. The iterated extended Kalman filter-smoother

We now extend the comparison to the case of the filter-smoother. Given $x_{k-1} \sim \mathcal{N}(x_{k-1|k-1}, P_{k-1|k-1})$, $x_k = f_{k-1}(x_{k-1}) + \Gamma_{k-1} w_{k-1}$, $z_k = h_k(x_k) + v_k$, the estimation-smoothing problem for $\{x_{k-1}, x_k\}$ is posed as

$$\text{Minimize } J = \frac{1}{2}\|x_{k-1} - x_{k-1|k-1}\|^2_{P_{k-1|k-1}^{-1}} + \frac{1}{2}\|z_k - h_k(x_k)\|^2_{R_k^{-1}} + \frac{1}{2}\|x_k - f_{k-1}(x_{k-1})\|^2_{\Omega_{k-1}^{-1}} \tag{13}$$

or, equivalently,

$$\text{Minimize } J(x_{k-1}, x_k) = \frac{1}{2}\left\| \begin{pmatrix} x_{k-1} - x_{k-1|k-1} \\ x_k - f_{k-1}(x_{k-1}) \\ h_k(x_k) - z_k \end{pmatrix} \right\|^2_{W_{k-1}^{-1}}$$

where $W_{k-1} = \text{diag}(P_{k-1|k-1}, \Omega_{k-1}, R_k)$. This corresponds to the case of $N = 1$ in equation (8).

Given an initial approximation (x_{k-1}^0, x_k^0), the sequence $\{(x_{k-1}^i, x_k^i)\}_{i \geq 0}$ is generated via

$$(x_{k-1}^{i+1}, x_k^{i+1}) = (x_{k-1}^i, x_k^i) + (\Delta x_{k-1}^i, \Delta x_k^i)$$

The correction terms $(\Delta x^i_{k-1}, \Delta x^i_k)$ are the solution of the linear least squares problem

$$\text{Minimize}\left\| \begin{pmatrix} I & 0 \\ -D_{x^i_{k-1}} f_{k-1}(x^i_{k-1}) & I \\ 0 & D_{x_k} h_k(x^i_k) \end{pmatrix} \begin{pmatrix} \Delta x^i_{k-1} \\ \Delta x^i_k \end{pmatrix} + \begin{pmatrix} x^i_{k-1} - x_{k-1|k-1} \\ x^i_k - f_{k-1}(x^i_{k-1}) \\ h_k(x^i_k) - z_k \end{pmatrix} \right\|^2_{W^{-1}_{k-1}} \tag{14}$$

The normal equations for this least squares problem (14) are

$$\begin{aligned} &\begin{pmatrix} P^{-1}_{k-1|k-1} + F^T_{k-1}(x^i_{k-1})\Omega^{-1}_{k-1}F_{k-1}(x^i_{k-1}) & -F^T_{k-1}(x^i_{k-1})\Omega^{-1}_{k-1} \\ -\Omega^{-1}_{k-1}F_{k-1}(x^i_{k-1}) & \Omega^{-1}_{k-1} + H^T_k(x^i_k)R^{-1}_k H_k(x^i_k) \end{pmatrix} \begin{pmatrix} \Delta x^i_{k-1} \\ \Delta x^i_k \end{pmatrix} \\ &= \begin{pmatrix} P^{-1}_{k-1|k-1} & -F^T_{k-1}(x^i_{k-1})\Omega^{-1}_{k-1} & 0 \\ 0 & \Omega^{-1}_{k-1} & H^T_k(x^i_k)R^{-1}_k \end{pmatrix} \begin{pmatrix} x_{k-1|k-1} - x^i_{k-1} \\ f_{k-1}(x^i_{k-1}) - x^i_k \\ z_k - h_k(x^i_k) \end{pmatrix} \end{aligned}$$

Numerically, one can solve these normal equations using three different factorization schemes: a) Cholesky Factorization, b) QR factorization, c) singular value decomposition (SVD). We, instead, solve them directly using the matrix inversion lemma (and considerable linear algebra). The closed-form solution to the the correction terms is given by

$$\begin{aligned} \Delta x^i_{k-1} &= (x_{k-1|k-1} - x^i_{k-1}) \\ &\quad + S_k(x^i_{k-1})K_k(x^i_{k-1}, x^i_k)\Big(z_k - h_k(x^i_k) - H_k(x^i_k)\big[F_{k-1}(x^i_{k-1})(x_{k-1|k-1} - x^i_{k-1}) + (f_{k-1}(x^i_{k-1}) - x^i_k)\big]\Big) \\ \Delta x^i_k &= F_{k-1}(x^i_{k-1})(x_{k-1|k-1} - x^i_{k-1}) + (f_{k-1}(x^i_{k-1}) - x^i_k) \\ &\quad + K_k(x^i_{k-1}, x^i_k)\Big(z_k - h_k(x^i_k) - H_k(x^i_k)\big[F_{k-1}(x^i_{k-1})(x_{k-1|k-1} - x^i_{k-1}) + (f_{k-1}(x^i_{k-1}) - x^i_k)\big]\Big) \end{aligned}$$

where

$$\begin{aligned} H_k(x^i_k) &= D_{x_k} h_k(x^i_k), \\ F_{k-1}(x^i_{k-1}) &= D_{x_{k-1}} f_{k-1}(x^i_{k-1}), \\ P_{k|k-1}(x^i_{k-1}) &= \Omega_{k-1} + F_{k-1}(x^i_{k-1})P_{k-1|k-1}F^T_{k-1}(x^i_{k-1}), \\ K_k(x^i_{k-1}, x^i_k) &= P_{k|k-1}(x^i_{k-1})H^T_k(x^i_k)\Big(H_k(x^i_k)P_{k+1|k}(x^i_{k-1})H_k(x^i_k)^T + R_k\Big)^{-1}, \\ S_k(x^i_{k-1}) &= P_{k-1|k-1}F^T_{k-1}(x^i_{k-1})P_{k|k-1}(x^i_{k-1})^{-1}, \\ S_k(x^i_{k-1})K_k(x^i_{k-1}, x^i_k) &= P_{k-1|k-1}F^T_{k-1}(x^i_{k-1})H^T_k(x^i_k)\Big(H_k(x^i_k)P_{k+1|k}(x^i_{k-1})H^T_k(x^i_k) + R_k\Big)^{-1}. \end{aligned}$$

The covariance of the states x^i_{k-1} and x^i_k, denoted by $P_{k-1,k|k-1}$ is given by

$$(P_{k-1,k|k-1}(x^i_{k-1}, x^i_k))^{-1} = \begin{pmatrix} P^{-1}_{k-1|k-1} + F^T_{k-1}(x^i_{k-1})\Omega^{-1}_{k-1}F_{k-1}(x^i_{k-1}) & -F^T_{k-1}(x^i_{k-1})\Omega^{-1}_{k-1} \\ -\Omega^{-1}_{k-1}F_{k-1}(x^i_{k-1}) & \Omega^{-1}_{k-1} + H^T_k(x^i_k)R^{-1}_k H_k(x^i_k) \end{pmatrix}.$$

Note that these equations represent precisely the iterations in the extended Kalman filter-smoother.[1]

In the case $N > 1$, we conjecture that an iteration of the extended Kalman filter-smoother over several states corresponds to Gauss-Newton steps in the solution of the nonlinear least squares.

2.2.3. Remarks on The Nonlinear Least Squares Approach

Given the above observations, one would in general add a globalization for the nonlinear least squares problem. This could be achieved with a line search or a trust region method (i.e., a modern form of the Levenberg-Marquardt update). In addition, one one can combine a full Newton step with one of these globalization techniques. We state without proof that the full Newton step corresponds to a quadratic Kalman smoother-filter pass over the data.

3. ML AND MAP: THE DISCRETE LINEAR PROBLEM

We now examine the batch ML/MAP estimation formulations for the case of linear problems. Relative to the nonlinear problem, two aspects are important in the linear case: (i) the least squares solution is computed in precisely one Gauss-Newton step, and (ii) an efficient compact formulation (with or without a prior) is possible and we present a new approach here. The linear problem is important in tracking because some nonlinear problems can be transformed to linear ones via the converted measurement[5] approach.

3.1. The mathematical model

The discrete process and observation models are given by

$$x_{k+1} = F_k x_k + g_k + \Gamma_k w_k, \quad z_k = H_k x_k + u_k + v_k,\ x_0 \sim \mathcal{N}(\overline{x}_0, P_0)$$

where

$x_k = n$-dimensional state vector,	$z_k = m$-dimensional observation vector,
$F_k = n \times n$ matrix,	$H_k = m \times n$ matrix,
$g_k = n$-dimensional known input vector,	$u_k = m$-dimensional known input vector,
$\Gamma_k = n \times r$ matrix,	$v_k = m$-dimensional observation-noise vector.
$w_k = r$-dimensional plant-noise vector.	

For the discrete estimation model, w_k and v_k are assumed to be independent zero-mean Gauss-Markov white sequences such that $E\{w_k w_j^T\} = Q_k \delta_{k-j}$, $E\{v_k v_j^T\} = R_k \delta_{k-j}$, $E\{x_0 w_k^T\} = 0$, $E\{x_0 v_k^T\} = 0$, $x_0 \sim \mathcal{N}(\overline{x}_0, P_0)$ where δ_{k-j} is the Kronecker delta function, and Q_k and R_k are positive definite $r \times r$ and $m \times m$ covariance matrices, respectively.

REMARK 2. *The form of the linear estimation problem (incorporation of g_k and u_k) is intended to cover the Gauss-Newton step in the nonlinear estimation problem.*

3.2. Batch ML estimation with process noise

Given the sequence of measurements $\{z_0, z_1, \ldots, z_N\}$, the problem is to estimate $\{x_0, x_1, \ldots, x_N\}$ assuming that there is no prior. (Note that the notation has changed now from the previous section in which the measurements were numbered beginning at one (1)). The determination of $\{x_0, x_1, \ldots, x_N\}$ via a maximum likelihood estimation is the solution of the linear least squares problem

$$\text{Minimize } J = \frac{1}{2}\|z_0 - H_0 x_0 - u_0\|^2_{R_0^{-1}} + \frac{1}{2}\sum_{k=1}^{N}\|z_k - H_k x_k - u_k\|^2_{R_k^{-1}} + \frac{1}{2}\sum_{k=1}^{N}\|x_k - F_{k-1}x_{k-1} - g_{k-1}\|^2_{\Omega_{k-1}^{-1}} \quad (15)$$

3.3. Batch MAP estimation with process noise

Given the sequence of measurements denoted by $\{z_1, \ldots, z_N\}$ at the times $\{t_1, t_2, \ldots, t_N\}$, respectively, and assuming that there is a complete or partial prior at $t_0 \le t_1$, the estimation of $\{x_0, x_1, \ldots, x_N\}$ can be posed as a linear least squares problem (15) in which the equation

$$z_0 = H_0 x_0 + u_0 + v_0, \quad (16)$$

where $v_0 \sim \mathcal{N}(0, R_0)$, can be used to cover the case of both the partial and full priors. Here are the two cases.

Case 1: Given a prior $x_0 \sim \mathcal{N}(\overline{x}_0, P_0)$ at a time t_0 before the first measurement at time t_1 ($t_0 < t_1$) so that the measurements and states are $\{z_1, \ldots, z_N\}$ and $\{x_0, \ldots, x_N\}$, respectively, then $z_0 = H_0 x_0 + u_0 + v_0$ is interpreted as

$$\overline{x}_0 = I x_0 + 0 + (-\mu_0) \quad (17)$$

i.e., $z_0 = \overline{x}_0$, $H_0 = I$, $u_0 = 0$, $v_0 = -\mu_0$.

Case 2: If the prior at a time before the first measurement is just part of the state, then one can proceed as in the following example. Suppose the state x is partitioned as $x = [p\ v]^T$, then $\overline{x}_0 = x_0 - \mu_0$ is replaced by

$$\begin{pmatrix} 0 & I \end{pmatrix} \overline{x}_0 = \begin{pmatrix} 0 & I \end{pmatrix} x_0 - \begin{pmatrix} 0 & I \end{pmatrix} \mu_0$$

3.3.1. A compact least squares formulation

The first step in the derivation of the compact form is to note

$$x_p = \begin{cases} \Phi_{pk} x_k - \sum_{i=p+1}^{k} \Phi_{pi}\big(g_{i-1} + \Gamma_{i-1} w_{i-1}\big) & \text{for } p = 0, 1, \ldots, k-1 \text{ if } p < k \\ \Phi_{pk} x_k + \sum_{i=k+1}^{p} \Phi_{pi}\big(g_{i-1} + \Gamma_{i-1} w_{i-1}\big) & \text{for } p = k+1, \ldots, N \text{ if } p > k, \end{cases}$$

where

$$\Phi_{pi} = \begin{cases} \big(F_{i-1} F_{i-2} \cdots F_p\big)^{-1} & \text{for } p < i \\ I & \text{for } p = i, \\ F_{p-1} F_{p-2} \cdots F_i & \text{for } p > i. \end{cases}$$

Define

$$\epsilon_{pk} = \begin{cases} v_p - H_p\big(\sum_{i=p+1}^{k} \Phi_{pi} \Gamma_{i-1} w_{i-1}\big) & \text{for } p < k \\ v_k & \text{for } p = k, \\ v_p + H_p\big(\sum_{i=k+1}^{p} \Phi_{pi} \Gamma_{i-1} w_{i-1}\big) & \text{for } p > k, \end{cases} \tag{18}$$

and

$$z_{pk} = \begin{cases} z_p - u_p + H_p\big(\sum_{i=p+1}^{k} \Phi_{pi} g_{i-1}\big) & \text{for } p < k \\ z_k - u_k & \text{for } p = k, \\ z_p - u_p - H_p\big(\sum_{i=k+1}^{p} \Phi_{pi} g_{i-1}\big) & \text{for } p > k. \end{cases} \tag{19}$$

Then, one can write

$$z_{pk} = H_p \Phi_{pk} x_k + \epsilon_{pk} \text{ for } p = 0, 1, \ldots, N. \tag{20}$$

In vector notation

$$\mathcal{Z}_{Nk} = \mathcal{H}_{Nk} x_k + \epsilon_{Nk} \tag{21}$$

where

$$\mathcal{H}_{Nk} = \begin{pmatrix} H_0 \Phi_{0k} \\ H_1 \Phi_{1k} \\ \vdots \\ H_{k-1} \Phi_{k-1k} \\ H_k \Phi_{kk} \\ H_{k+1} \Phi_{k+1k} \\ \vdots \\ H_{N-1} \Phi_{N-1k} \\ H_N \Phi_{Nk} \end{pmatrix}, \quad \mathcal{Z}_{Nk} = \begin{pmatrix} z_{0k} \\ z_{1k} \\ \vdots \\ z_{k-1k} \\ z_{kk} \\ z_{k+1k} \\ \vdots \\ z_{N-1k} \\ z_{Nk} \end{pmatrix}, \text{ and } \mathcal{E}_{Nk} = \begin{pmatrix} \epsilon_{0k} \\ \epsilon_{1k} \\ \vdots \\ \epsilon_{k-1k} \\ \epsilon_{kk} \\ \epsilon_{k+1k} \\ \vdots \\ \epsilon_{N-1k} \\ \epsilon_{Nk} \end{pmatrix} \tag{22}$$

3.3.2. Solution methods for the linear least squares problem

In this subsection, we assume that the given data is $\{z_0, z_1, \ldots, z_N\}$ at times $\{t_0, t_1, \ldots, t_N\}$ so that we are considering the problem

$$\text{Minimize } J = \frac{1}{2}\|z_0 - H_0 x_0 - u_0\|^2_{R_0^{-1}} + \frac{1}{2}\sum_{k=1}^{N} \|z_k - H_k x_k - u_k\|^2_{R_k^{-1}} + \frac{1}{2}\sum_{k=1}^{N} \|x_k - F_{k-1} x_{k-1} - g_{k-1}\|^2_{\Omega_{k-1}^{-1}}$$

The two prior cases discussed above can be handled within the current formulation by appropriately interpreting the equation $z_0 = H_0 x_0 + u_0 + v_0$. The solution x_k can be posed as that of a linear least squares problem,

$$\text{Minimize } \|\mathcal{H}_{Nk} x_k - \mathcal{Z}_{Nk}\|^2_{W_{Nk}^{-1}} \tag{23}$$

where $W_{Nk} = E\{\mathcal{E}_{Nk}\mathcal{E}_{Nk}^T\}$ is the weighting or covariance matrix, $Q = \text{diag}\ (Q_0, \ldots, Q_{N-1})$ and $R = \text{diag}\ (R_0, \ldots, R_N)$. The normal equation for this problem is

$$\left(\mathcal{H}_{Nk}^T W_{Nk}^{-1} \mathcal{H}_{Nk}\right) x_k = \mathcal{H}_{Nk}^T W_{Nk}^{-1} \mathcal{Z}_{Nk}. \tag{24}$$

Assuming $\left(\mathcal{H}_{Nk}^T W_{Nk}^{-1} \mathcal{H}_{Nk}\right)$ is nonsingular, the solution to the normal equations is

$$x_k = \left(\mathcal{H}_{Nk}^T W_{Nk}^{-1} \mathcal{H}_{Nk}\right)^{-1} \mathcal{H}_{Nk}^T W_{Nk}^{-1} \mathcal{Z}_{Nk}, \tag{25}$$

where

$$P_k = \left(\mathcal{H}_{Nk}^T W_{Nk}^{-1} \mathcal{H}_{Nk}\right)^{-1} \tag{26}$$

is identified as the covariance matrix of x_k.

The weighting matrix $W_{Nk} = [w_{pq}]_{(p,q)\in[0,1,\ldots,N]\times[0,1,\ldots,N]}$ has the following form

$$W_{Nk} = \begin{pmatrix} w_{00} & w_{01} & \cdots & w_{0k-1} & 0 & 0 & 0 & \cdots & 0 \\ w_{10} & w_{11} & \cdots & w_{1k-1} & 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots \\ w_{k-10} & w_{k-11} & \cdots & w_{k-1k-1} & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & w_{kk} & 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 0 & w_{k+1k+1} & w_{k+1k+2} & \cdots & w_{k+1N} \\ 0 & 0 & \cdots & 0 & 0 & w_{k+2k+1} & w_{k+2k+2} & \cdots & w_{k+1N} \\ \vdots & \vdots & \vdots & \cdots & \vdots & & & & \\ 0 & 0 & \cdots & 0 & 0 & w_{Nk+1} & w_{Nk+2} & \cdots & w_{NN} \end{pmatrix}, \tag{27}$$

where the computation of $w_{pq} = E\{\epsilon_{pk}\epsilon_{qk}^T\} \in \Re^{m\times m}$, which is a block symmetric and positive semi-definite matrix, can be divided into several natural cases:

$$\text{For } q < p < k \quad w_{pq} = H_p\left(\sum_{i=p+1}^{k} \Phi_{pi}(\Gamma_{i-1}Q_{i-1i-1}\Gamma_{i-1}^T)\Phi_{qi}^T\right)H_q^T \tag{28}$$

$$\text{For } q = p < k \quad w_{pp} = R_{pp} + H_p\left(\sum_{i=p+1}^{k} \Phi_{pi}(\Gamma_{i-1}Q_{i-1i-1}\Gamma_{i-1}^T)\Phi_{pi}^T\right)H_p^T \tag{29}$$

$$\text{For } q < p = k \quad w_{kq} = 0 \tag{30}$$

$$\text{For } q = p = k \quad w_{kk} = R_{kk} \tag{31}$$

$$\text{For } p > k \geq q \quad w_{pq} = 0 \tag{32}$$

$$\text{For } p > q > k \quad w_{pq} = H_p\left(\sum_{i=k+1}^{q} \Phi_{pi}(\Gamma_{i-1}Q_{i-1i-1}\Gamma_{i-1}^T)\Phi_{qi}^T\right)H_q^T \tag{33}$$

$$\text{For } q = p > k \quad w_{pp} = R_{pp} + H_p\left(\sum_{i=k+1}^{p} \Phi_{pi}(\Gamma_{i-1}Q_{i-1i-1}\Gamma_{i-1}^T)\Phi_{pi}^T\right)H_p^T. \tag{34}$$

The shape of the matrix W_{Nk} depends on the value of k and generally separates into the following cases: (a) $k = 0$, (b) $1 \leq k \leq N-1$, (c) $k = N$. In each case, the kth row and column are zero with the single exception of the block w_{kk}. The form for $1 \leq k \leq N-1$ is given above, while the form for $k = 0$ and $k = N$ are as follows:

$$W_{N0} = \begin{pmatrix} w_{00} & 0 & 0 & \cdots & 0 \\ 0 & w_{11} & w_{12} & \cdots & w_{1N} \\ 0 & w_{21} & w_{22} & \cdots & w_{2N} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 0 & w_{N1} & w_{N2} & \cdots & w_{NN} \end{pmatrix} \quad (k = 0), \tag{35}$$

$$W_{NN} = \begin{pmatrix} w_{00} & w_{01} & \cdots & w_{0N-1} & 0 \\ w_{10} & w_{11} & \cdots & w_{1N-1} & 0 \\ \vdots & \vdots & \ldots & \vdots & \vdots \\ w_{N-10} & w_{N-11} & \cdots & w_{N-1N-1} & 0 \\ 0 & 0 & \cdots & 0 & w_{NN} \end{pmatrix} \quad (k = N). \tag{36}$$

REMARK 3. *The batch estimation solution for the case of process noise but no prior is handled by deleting the first row and column of (27) and altering (22) accordingly.*

4. SIMULATION RESULTS

To verify the accuracy of the batch estimator and the value of using process noise and the prior state estimate within the batch estimator, a simulation was implemented for the linear problem using the estimator in (25). A constant velocity single target was propagated from an initial state. Cartesian 3-D measurements were generated and random Gaussian noise was added to each measurement (covariance of $R = 100 \times I$). Gaussian process noise was also added to the dynamic propagation of the target (covariance of $Q = 100 \times I$). The batch filter with a window size of $N = 11$ was implemented with the measurement noise covariance matched to the actual Gaussian noise covariance used to generate the data. Different cases were assumed about use of the process noise and the prior state within the batch estimator, as will be explained.

In the first case, shown in Figure 1, the batch estimator only assumed the presence of measurement noise (i.e., no process noise and no prior estimate were used in the estimator). The first figure shows the position RMSE accuracy versus the the index $k \in \{1, \ldots, 11\}$ while the second plot shows the velocity RMSE accuracy. The index k refers to batch index (a discrete time index within the window) where the state estimate is generated. Also plotted is the square-root of the trace of the batch estimator covariance matrix. This represents the "perceived" accuracy of the estimate as produced by the batch estimator. Several important things should be noticed in these plots. First, the shape of the position RMSE implies that the estimate is more accurate at certain time points. The position estimate is most accurate at $k = 3$ and $k = 9$ while the velocity estimate is most accurate at $k = 6$. Meanwhile, the square-root of the trace of the covariance is much smaller than the RMSE, indicating that the estimator thinks it is doing much better than it actually is. This mismatch is attributed to the fact that the assumed model (just measurement noise) is not correct.

The next case is shown in Figures 2. Here, the batch estimator was implemented with both measurement and process noise. Comparing Figure 1 and Figure 2 we see that the latter estimator is much more accurate than the former (i.e., including process noise in the estimator improves the accuracy). In addition, the RMSE and the square-root of the trace of the covariance matrix match, indicating that the uncertainty reported by the batch estimator covariance represents the actual performance of the state estimate (i.e., there is a model match). Also notice that the estimator accuracy is significantly worse at the end points of the batch, $k \in \{1, 11\}$, while it is roughly the same for the internal points, $k \in \{2, 3, \ldots, 10\}$. This implies that if one wants to obtain the best possible estimate of the target state, then one should choose an internal point in the batch at which to request the estimate. Generally speaking, this result makes sense because a "polynomial curve fit" to data is the least accurate at the end-points in a data set.

The third case is shown in Figures 3. Here, the batch estimator is implemented with measurement and process noise and a prior state estimate. The covariance of the prior state was set to $P_0 = 100 \times I$. The prior state estimate used by the batch estimator was provided at the time of the first measurement, i.e., $k = 1$. These plots show that the RMSE accuracy agrees with the square-root of the trace of the covariance, thus the model match is achieved. Also notice that at $k \in \{1, 2, 3\}$ that the estimator accuracy is much better. This is the influence of the prior state (i.e., the availability of prior information improves the estimate here). As k increases, the prior state must be propagated from time t_1 to time t_k. As it is propagated, the uncertainty grows and the value of the prior information diminishes. At $k = 4$ we see that the RMSE is equivalent to that shown in Figure 2, thus there is no benefit provided by the prior here.

The forth case is shown in Figure 4. Here, we have created a constant accelerating target with an an initial state value of $x_0 = [100,\ 100,\ 100,\ 50,\ 50,\ 50,\ 10,\ 10,\ 10]$. Measurement noise was added with $R = 100 \times I$. Meanwhile, in our batch estimator we assume only a nearly constant velocity model (i.e., a six-state model). In Figure 4, the plot on the left shows the RMSE and the covariance trace for the batch estimator that assumes no process noise. The plot on the right

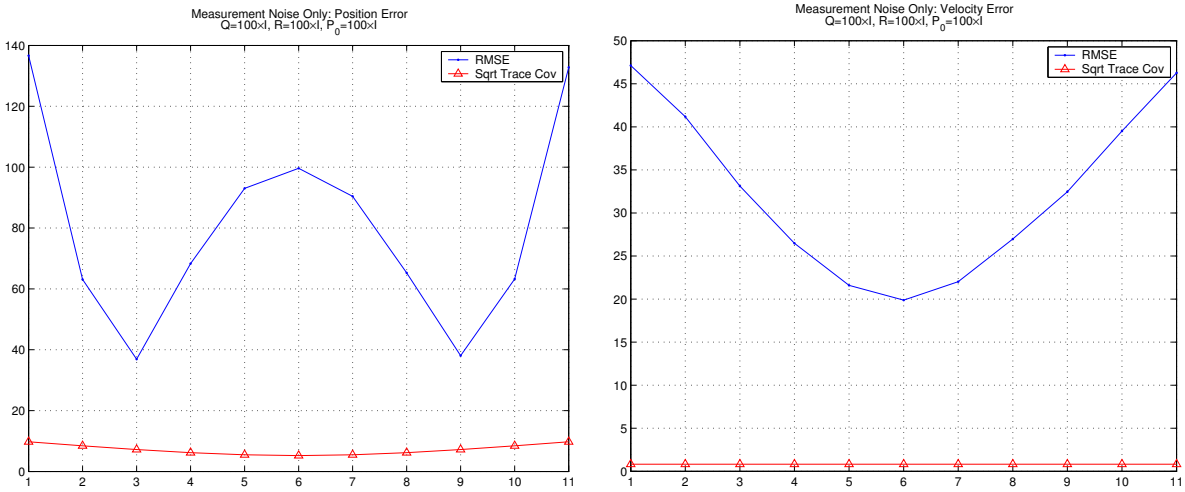

Figure 1: Plot of the position and velocity RMSE for measurement noise only case.

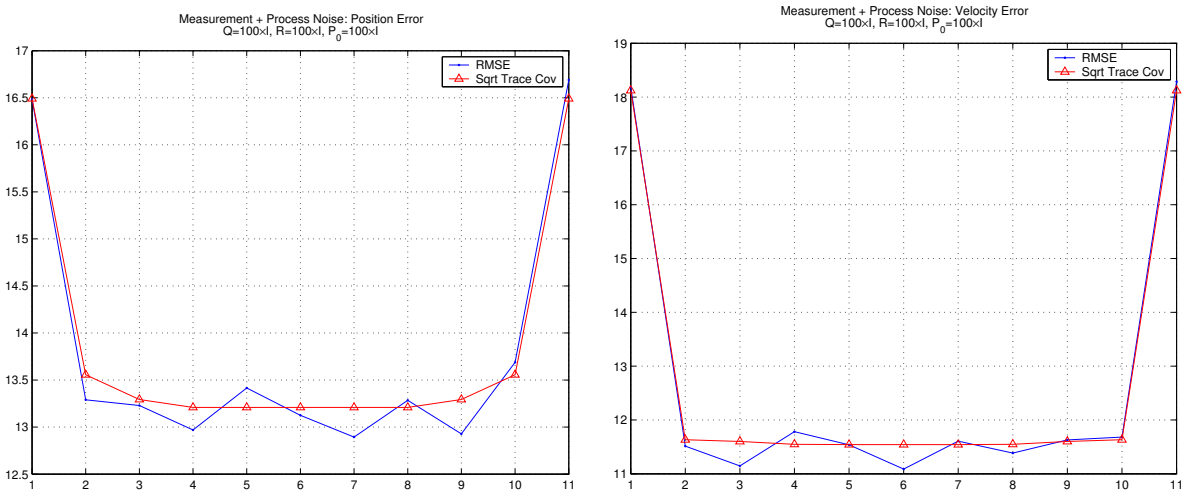

Figure 2: Plot of the position and velocity RMSE for measurement and process noise case.

is for the batch estimator with process noise, where $Q = 100 \times I$. As can be seen, the estimator without process noise performs much worse than the one with process noise. Furthermore, the covariance estimate is highly optimistic (i.e., it thinks the accuracy is much better than it actually is), while the square root of the covariance trace for the batch with process noise is comparable to the RMSE. Hence, the estimation accuracy and the reported covariance are much better when using the batch with process noise.

In summary, single target simulations were performed to verify the correctness of the batch filter implementation. In the case where both measurement noise and process noise were assumed in the filter model, the RMSE accuracy agreed with the square-root of the trace of the covariance. This agreement proved that the estimator was consistent and performs correctly. When we compared the performance to the case where no process noise was assumed in the batch filter, we found that the estimator accuracy was much worse, and the covariance reported was inconsistent with the accuracy of the state estimate. When a prior state is included in the batch estimator, the information provided significantly improved the batch estimate, but only at time points near that of the provided prior state estimate. Finally, when the target dynamics do not match the assumed model (e.g., the target is accelerating when a constant velocity model is used), then the batch estimation with process noise performs significantly better than that without process noise.

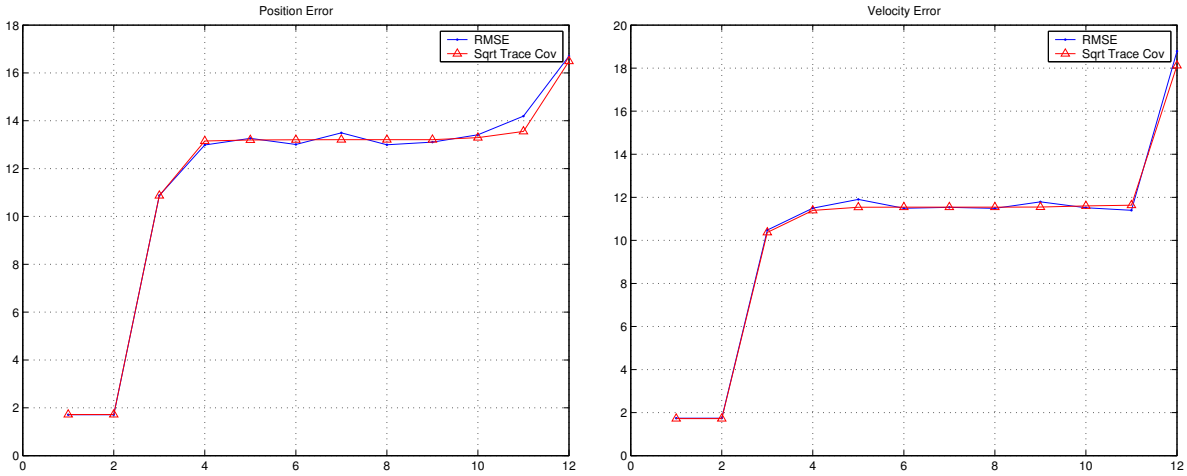

Figure 3: Plot of the position and velocity RMSE for measurement and process noise and prior state estimate case.

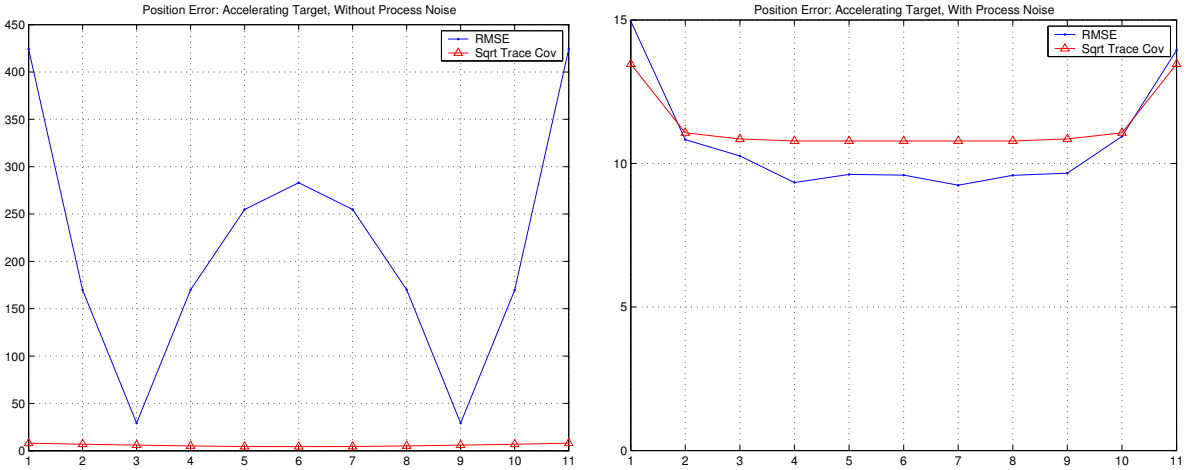

Figure 4. Comparison of position RMSE and covariance trace for an accelerating target for batch estimator without (left) process noise and with (right) process noise.

5. SUMMARY

In summary, this paper has shown the formulation of batch MAP and ML estimation algorithms where process noise has been included in the model. The use of process noise is important in tracking applications where there is uncertainty in the target dynamical model. We first formulated the batch estimator for the discrete nonlinear case. The MAP and ML formulations were given, as well as the relationships to the Iterated Extended Kalman Filter and the Iterated Extended Kalman Filter Smoother. We then formulated the batch estimators for the linear discrete problem, and showed a compact least squares formulation that handles the cases of with/without a prior and with/without process noise. Finally, we presented simulation results that demonstrated the value of the linear batch estimator with process noise and with a prior.

ACKNOWLEDGMENTS

This work was supported in part by the Air Force Office of Scientific Research under grant number F49620-00-1-0108 and the US Air Force under contract F33601-02-F-A581.

REFERENCES

1. A. Jazwinski, *Stochastic processes and filtering theory*, Academic Press, New York, 1970.
2. A. P. Sage and J. L. Melsa, *Estimation Theory with Applications to Communications and Control*, McGraw-Hill, New York, 1971.
3. A. Sage and J. Melsa, *Estimation theory with applications to communications and control*, McGraw-Hill, New York, 1971.
4. B. Bell and F. Cathey, "The iterated kalman filter update as a gauss-newton method," *IEEE Trans. Automatic Control* **38**(2), pp. 294–298, 1993.
5. L. Mo, X. Song, Z. Sun, and Y. Bar-Shalom, "Unbiased converted measurements for tracking," *IEEE Transactions on Aerospace and Electronic Systems* **34**(3), pp. 1023–1026, 1998.