Multi-Stage Iteration Filter-Smoother

Yasuhiro KAWAZOE*

(Received November 30, 1972)

Abstract

The problem of state estimation for noisy discrete-time nonlinear dynamic systems is discussed by proposing the multi-stage iteration filter-smoother in conjunction with the algorithms for prediction, filtering and smoothing which are based upon the viewpoint of marginal maximum likelihood estimation.

Results of the numerical example indicate greatly improved preformance over the filtering only.

1. Introduction

In this paper, the problem of the state estimation for noisy discrete-time nonlinear dynamic systems is developed by the use of the approximated algorithms for sequential prediction, filtering and smoothing. The estimation of states in noisy nonlinear dynamic systems based on noisy nonlinear measurements has been investigated in the past few years and, especially, the effort has been done in the area of nonlinear filtering for the purpose of estimating the current state for the control. On the other hand, the nonlinear smoothing has been little discussed. The problem of the nonlinear smoothing is very important in the viewpoint of the parameter estimation problem, since we may desire to have the better estimate by the use of the smoother, when the observed information is limited.

The nonlinear optimal filter needs in theory the solution of an infinite-dimensional process. However, in the practical side of technology, there is the needness of the approaches to suboptimal filtering for the nonlinear system, since the computational aspects of the truly optimal nonlinear filter are prohitive.

The suboptimal nonlinear filters can be roughly subdivided into first-order filters and higher-order filters. In general, in higher-order filters, the second-order filter is used.

Wishner et, al discussed three different estimation schemes, the first-order filter, the second order filter and the single-stage iteration filter.[1] In the former two filter, if the norminal trajectory and state are not close to the true trajectory and state, the truncated expression in the Taylor series represents poor approximation.

The first-order filter has been pointed out to be a biased estimator.[2]

* Department of Electrical Engineering, Technical Junior College of Yamaguchi University

The bias is due to the multiplicative effect of nonlinearities in the state equation and the observation equation.

Chen and Eulrich used some local iteration algorithm based on the first-order filter in conjunction with the use of one stage optimal smoothing, for the purpose of improving the approximation of the nonlinearities, and it was shown that the bias is reduced by the local iteration.[3]

Meditch provided the approximated algorithm for sequential prediction, filtering and smoothing from the viewpoint of marginal maximum likelihood estimation.[4] The algorithm utilized the approximations which are first-order in the system dynamics and second-order in the measurement function for prediction and filtering, and the approximation which is second-order in the system dynamics.

The author discusses in this paper the multi-stage iteration filter-smoother in conjunction with the algorithms for prediction, filtering and smoothing which are discussed by Meditch.

This paper is organized as follows: The problem formulation is given in Section 2, and the algorithms for prediction, filtering and smoothing are given in Section 3.

The derivation of the multi-stage iteration filtering-smoothing algorithm is given in Section 4. A simple numerical example is presented in Section 5. Finally, a discussion and a conclusion of the results are offered in Section 6 and Section 7.

2. Statement of the Problem

Consider the nonlinear discrete-time system driven by zero mean Gaussian noise $\boldsymbol{w}_k$, and discrete noisy measurements,

$$\boldsymbol{x}_{k+1}=\boldsymbol{f}(\boldsymbol{x}_k)+\boldsymbol{w}_k \quad (1)$$

$$\boldsymbol{y}_k=\boldsymbol{h}(\boldsymbol{x}_k)+\boldsymbol{v}_k \quad (2)$$

where

k is the discrete-time index,

$\boldsymbol{x}$ is an n-vector, the state,

$\boldsymbol{y}$ is an m-vector, the measurements,

$\boldsymbol{w}$ and $\boldsymbol{v}$ are respectively the white Gaussian random sequences with zero mean and

$$E[\boldsymbol{w}_i\boldsymbol{w}_k^T]=Q_k\delta_{ik} \quad (3)$$

$$E[\boldsymbol{v}_i\boldsymbol{v}_k^T]=R_k\delta_{ik} \quad (4)$$

$$E[\boldsymbol{w}_i\boldsymbol{v}_k^T]=0 \quad (5)$$

δ_{ik} is the Kronecker delta,

Q and R are assumed to be positive definite,

$\boldsymbol{f}(\cdot)$ and $\boldsymbol{h}(\cdot)$ are respectively n and m-dimensional, vector valued, twice continuously differentiable functions of the indicated variables.

We obtain an estimate $\boldsymbol{x}_k$, $k \geq 0$, given the sequence of measurements $\{\boldsymbol{y}_0, \boldsymbol{y}_1, \ldots, \boldsymbol{y}_j\}, j \geq 0$. We shall have three cases of (i) prediction, where $j=k-1$ and $k=0, 1, \ldots$: (ii) filtering where $j=k$ and $k=0, 1, \ldots$: and (iii) smoothing where $j=N$ is fixed and $k=N-1, \ldots, 0$. The problem is to obtain algorithms for computing the best estimates for $\boldsymbol{x}_k$, $k=0, 1, \ldots$ using the measured data.

3. Algorithms of Prediction, Filtering and Smoothing

In this section, referring to References 4), we take likelihood functions,

$L(\boldsymbol{x}_k, Y_{k-1}) = ln\, p\, (\boldsymbol{x}_k | Y_{k-1})$, for prediction,

$L(\boldsymbol{x}_k, Y_k) = ln\, p\, (\boldsymbol{x}_k | Y_k)$, for filtering,

and $L(\boldsymbol{x}_k, \boldsymbol{x}_{k+1}, Y_N) = ln\, p\, (\boldsymbol{x}_k, \boldsymbol{x}_{k+1} | Y_N)$, for smoothing. In these relations, $Y_k = \{\boldsymbol{y}_0, \boldsymbol{y}_1, \ldots, \boldsymbol{y}_k\}$, and $p(\cdot|\cdot)$ is conditional probability density function. We consider only first and second-order moments and assume that third and higher-order moments are negligible.

The algorithms of prediction, filtering and smoothing are shown as follows.

Prediction:

$$\hat{\boldsymbol{x}}_{k/k-1} = \boldsymbol{f}(\hat{\boldsymbol{x}}_{k-1/k-1}) \tag{6}$$

$$P_{k/k-1} = f_x(\hat{\boldsymbol{x}}_{k-1/k-1}) P_{k-1/k-1} f_x^T(\hat{\boldsymbol{x}}_{k-1/k-1}) + Q_{k-1} \tag{7}$$

where

$$f_x(\hat{\boldsymbol{x}}_{k-1/k-1}) = \partial \boldsymbol{f}(\boldsymbol{x})/\partial \boldsymbol{x} \,|\, \boldsymbol{x} = \hat{\boldsymbol{x}}_{k-1/k-1} \tag{8}$$

Filtering:

$$\hat{\boldsymbol{x}}_{k/k} = \hat{\boldsymbol{x}}_{k/k-1} + G_k[\boldsymbol{y}_k - \boldsymbol{h}(\hat{\boldsymbol{x}}_{k/k-1})] \tag{9}$$

$$P_{k/k} = [I - P_{k/k-1} g_x(\hat{\boldsymbol{x}}_{k/k-1}, \boldsymbol{y}_k)]^{-1}\, P_{k/k-1} \tag{10}$$

where

$$G_k = P_{k/k} h_x^T(\hat{\boldsymbol{x}}_{k/k-1}) R_k^{-1} \tag{11}$$

$$\boldsymbol{g}(\hat{\boldsymbol{x}}_{k/k-1}, \boldsymbol{y}_k) = h_x^T(\hat{\boldsymbol{x}}_{k/k-1}) R_k^{-1}[\boldsymbol{y}_k - \boldsymbol{h}(\hat{\boldsymbol{x}}_{k/k-1})] \tag{12}$$

$$g_x(\hat{\boldsymbol{x}}_{k/k-1}, \boldsymbol{y}_k) = \partial \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{y}_k)/\partial \boldsymbol{x} \,|\, \boldsymbol{x} = \hat{\boldsymbol{x}}_{k/k-1} \tag{13}$$

Smoothing:

$$\hat{\boldsymbol{x}}_{k/N} = \hat{\boldsymbol{x}}_{k/k} + A_k(\hat{\boldsymbol{x}}_{k+1/N} - \hat{\boldsymbol{x}}_{k+1/k}) \tag{14}$$

$$P_{k/N} = P_{k/k} - A_k(P_{k+1/k} - P_{k+1/N})\, A_k^T \tag{15}$$

where

$$A_k=[I-P_{k/k}\gamma_x(\hat{x}_{k/k},\ \hat{x}_{k+1/N})]^{-1}P_{k/k}f_x^T(\hat{x}_{k/k})Q_k^{-1} \tag{16}$$

$$\gamma(\hat{x}_{k/k},\ \hat{x}_{k+1/N})=f_x^T(\hat{x}_{k/k})Q_k^{-1}[\hat{x}_{k+1/N}-f(\hat{x}_{k/k})] \tag{17}$$

$$\gamma_x(\hat{x}_{k/k},\ \hat{x}_{k+1/N})=\partial\gamma(x,\ \hat{x}_{k+1/N})/\partial x\,|\,x=\hat{x}_{k/k} \tag{18}$$

P is the error covariance matrix

$$P=E\{(x-\hat{x})(x-\hat{x})^T\} \tag{19}$$

where E denotes the expected value.

4. Multi-Stage Iteration Filter-Smoother

We take the approach to use the iteration algorithm based on the filter in conjunction with the use of the smoothing for improving the reference trajectory on the state and the estimate.

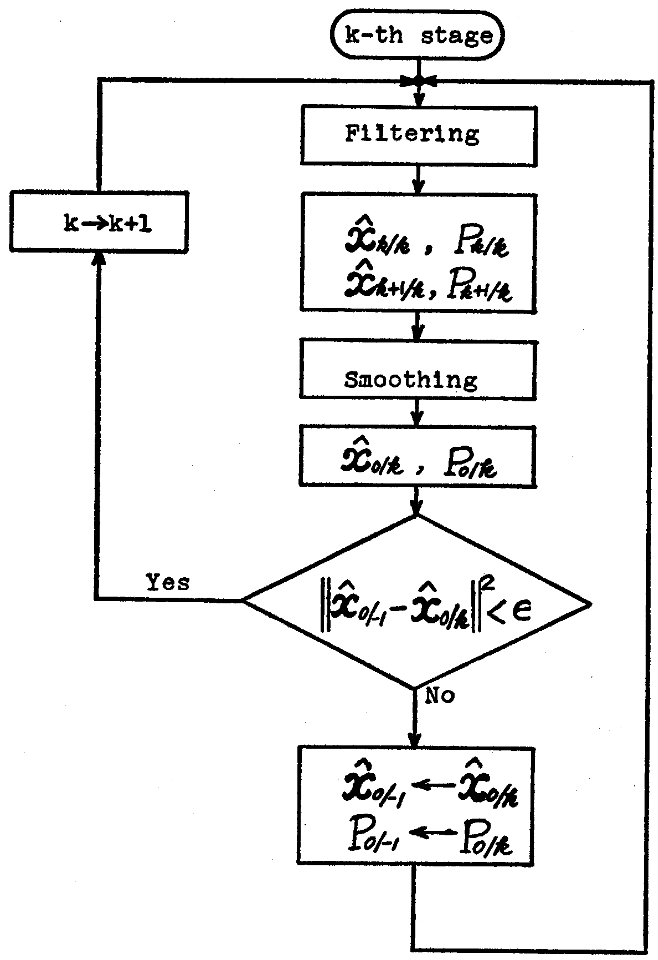

Fig. 1. Flow-chart of multi-stage iteration filter-smoother

The approach is as follows:

At the k-th stage, $\hat{\boldsymbol{x}}_{k/k}$ and $P_{k/k}$ are obtained from the filtering algorithm.

Before processing the data $\boldsymbol{y}_{k+1}$ and proceeding to the filtering at the next stage, we choose $\hat{\boldsymbol{x}}_{k/k}$, $\hat{\boldsymbol{x}}_{k+1/k}$, $P_{k/k}$ and $P_{k+1/k}$, and take the smoothing algorithm until the initial stage.

At the initial stage, we replace $\hat{\boldsymbol{x}}_{0/-1}$ and $P_{0/-1}$ by $\hat{\boldsymbol{x}}_{0/k}$ and $P_{0/k}$ respectively and then proceed the filtering algorithm from the initial stage to the k-th stage.

And repeat smoothing and filtering some times. If $\|\hat{\boldsymbol{x}}_{0/-1}-\hat{\boldsymbol{x}}_{0/k}\|^2$ is small enough, we stop repeating smoothing and filtering, and proceed to the filtering algorithm at the $(k+1)$-th stage, and so on. The flow-chart of this process is shown in *Fig.* 1 where ϵ is some small positive.

5. Example

We consider the second-order system.[4)]

$$\boldsymbol{x}_{k+1}=\begin{bmatrix} x_k^1/(1+x_k^1x_k^2) \\ x_k^2 \end{bmatrix}+\boldsymbol{w}_k \tag{20}$$

where

$$\boldsymbol{x}_k=[x_k^1,\ x_k^2]^T$$

$$Q_k=\begin{bmatrix} 25\cdot 10^{-4} & 0 \\ 0 & 10^{-6} \end{bmatrix}$$

The measurement is scalar.

$$y_k=(x_k^1)^3+v_k \tag{21}$$

where

$$R_k=10^{-2}$$

The error covariance matrix is denoted by

$$P=E\{(\boldsymbol{x}-\hat{\boldsymbol{x}})(\boldsymbol{x}-\hat{\boldsymbol{x}})^T\}=\begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix}. \tag{22}$$

We take the next starting conditions.

(a) True initial state: $\boldsymbol{x}_0 =[20,\ 0.30]^T$

(b) A priori state: $\hat{\boldsymbol{x}}_{0/-1}=[18,\ 0.34]^T$

(c) Error covariance matrix:

$$P_{0/-1}=\begin{bmatrix} 25 & 0 \\ 0 & 25\cdot 10^{-4} \end{bmatrix}$$

Table 1. Multi-stage iteration filtering-smoothing estimate when y_0, y_1 and y_2 are obtained

Iteration	1st-stage		2nd-stage		3rd-stage	
	x^1	x^2	x^1	x^2	x^1	x^2
0	21.0 (20.0)	0.34 (0.30)	2.99 (2.87)	0.279 (0.299)	1.54 (1.49)	0.284 (0.299)
2	20.2	0.273	2.91	0.279	1.53	0.280
4	20.1	0.283	2.88	0.285	1.52	0.285
6	20.1	0.287	2.87	0.288	1.52	0.290
8	20.1	0.289	2.87	0.290	1.52	0.290
10	20.1	0.291	2.87	0.292	1.52	0.292
15	20.0	0.294	2.86	0.294	1.52	0.294
20	20.0	0.295	2.86	0.295	1.52	0.295
25	20.0	0.296	2.86	0.296	1.52	0.296

( ): True value

Table 2. Error variances

Iteration	1st-stage		2nd-stage		3rd-stage	
	P_{11}	P_{22}	P_{11}	P_{22}	P_{11}	P_{22}
0	$0.141 \cdot 10^{-7}$	$0.250 \cdot 10^{-2}$	$0.332 \cdot 10^{-4}$	$0.570 \cdot 10^{-4}$	$0.132 \cdot 10^{-3}$	$0.503 \cdot 10^{-4}$
2	$0.238 \cdot 10^{-8}$	$0.110 \cdot 10^{-4}$	$0.106 \cdot 10^{-4}$	$0.882 \cdot 10^{-5}$	$0.143 \cdot 10^{-3}$	$0.963 \cdot 10^{-5}$
4	$0.138 \cdot 10^{-8}$	$0.601 \cdot 10^{-5}$	$0.123 \cdot 10^{-4}$	$0.603 \cdot 10^{-5}$	$0.153 \cdot 10^{-3}$	$0.964 \cdot 10^{-5}$
6	$0.977 \cdot 10^{-9}$	$0.425 \cdot 10^{-5}$	$0.132 \cdot 10^{-4}$	$0.476 \cdot 10^{-5}$	$0.158 \cdot 10^{-3}$	$0.576 \cdot 10^{-5}$
8	$0.758 \cdot 10^{-9}$	$0.332 \cdot 10^{-5}$	$0.138 \cdot 10^{-4}$	$0.402 \cdot 10^{-5}$	$0.162 \cdot 10^{-3}$	$0.498 \cdot 10^{-5}$
10	$0.620 \cdot 10^{-9}$	$0.274 \cdot 10^{-5}$	$0.142 \cdot 10^{-4}$	$0.353 \cdot 10^{-5}$	$0.164 \cdot 10^{-3}$	$0.451 \cdot 10^{-5}$
15	$0.427 \cdot 10^{-9}$	$0.192 \cdot 10^{-5}$	$0.149 \cdot 10^{-4}$	$0.282 \cdot 10^{-5}$	$0.168 \cdot 10^{-3}$	$0.380 \cdot 10^{-5}$
20	$0.326 \cdot 10^{-9}$	$0.148 \cdot 10^{-5}$	$0.152 \cdot 10^{-4}$	$0.242 \cdot 10^{-5}$	$0.170 \cdot 10^{-3}$	$0.341 \cdot 10^{-5}$
25	$0.263 \cdot 10^{-9}$	$0.121 \cdot 10^{-5}$	$0.155 \cdot 10^{-4}$	$0.217 \cdot 10^{-5}$	$0.171 \cdot 10^{-3}$	$0.316 \cdot 10^{-5}$

Table 3. Time histories of estimation errors by multi-stage iteration filtering-smoothing (M.I.F.S.) and filtering

Stage	$\tilde{x}^1$		$\tilde{x}^2$	
	M.I.F.S.	Filtering	M.I.F.S.	Filtering
3	0.030	——	0.003	——
4	0.020	0.038	0.005	0.009
6	0.022	0.066	0.003	0.010
8	0.007	0.068	0.000	0.01
10	0.091	0.022	0.001	0.01
12	0.187	——	0.000	——
14	0.107	——	0.000	——
16	0.120	——	0.002	——
18	0.090	——	0.001	——
20	0.066	——	0.001	——

6. Results and Discussion

The results are shown in Table 1～Table 3. Table 1 gives the multi-stage iteration filtering-smoothing estimates at the 3rd-stage, and Table 2 gives the corresponding approximate error variances. Time histories of estimation errors by the multi-stage iteration filter-smoother and the filter are given in Table 3. In Table 1 and Table 2, it is seen that the estimates are improved at the initial stages by iterations. From this example, iterations are not taken from the fourth-stage when the criterion of the convergence of the estimate is 10^{-6}. The performances of the multi-stage iteration filter-smoother and the filter are compared on the basis of the errors of the estimates in Table 3. It is seen that the multi-stage iteration filter-smoother gives significantly better performance, when compared to the filter only.

7. Conclusion

A multi-stage iteration smoothing-filtering algorithm has been obtained in conjunction with the prediction, filtering and smoothing in an on-line fashion to update the estimates of the initial state until new measurements become available. However, the sampling period must be large as the some iterations are possible in this approach. As a numerical example, we considered the second-order nonlinear system discussed by Meditch.[4]

The performance of the multi-stage iteration filter-smoother is successively improved. The author is presently engaged in comparing the multi-stage iteration filter-smoother with the locally iterated filter-smoother under the various conditions.[3]

8. References

1) R. P. Wishner, J. A. Jabaczynski and M. Athans: Automatica, **5,** 478–496 (1969)
2) M. Athans, R. P. Wishner and A. Bertolini: IEEE Trans. **AC-13,** 504–514 (1968)
3) R. T. N. Chen and B. J. Eulrich: JACC, 731–740 (1971)
4) J. S. Meditch: Proc. Ann Allerton Conf. Circuit Syst. Theory, **7,** 293–302 (1969)