New Conditional Posterior Cramér-Rao Lower Bounds for Nonlinear Sequential Bayesian Estimation

Yujiao Zheng, Onur Ozdemir, Ruixin Niu, and Pramod K. Varshney

***Abstract*—The recursive procedure to compute the posterior Cramér-Rao lower bound (PCRLB) for sequential Bayesian estimators, derived by Tichavsky *et al.*, provides an off-line performance bound for a general nonlinear filtering problem. Since the corresponding Fisher information matrix (FIM) is obtained by taking the expectation with respect to all the random variables, this PCRLB is not well suited for online adaptive resource management for dynamic systems. For online estimation performance evaluation in a nonlinear system, the concept of conditional PCRLB was proposed by Zuo *et al.* in 2011. In this paper, two other online conditional PCRLBs are proposed which are alternatives to the one proposed by Zuo *et al.* Numerical examples are provided to show that the three online bounds, namely the conditional PCRLB proposed by Zuo *et al.* and the two conditional PCRLBs proposed in this paper, are very close to one another.**

***Index Terms*—Nonlinear filtering, particle filters, posterior Cramér-Rao lower bounds.**

I. Introduction

The posterior (or Bayesian) Cramér-Rao lower bound (PCRLB, or BCRLB) is defined to be the inverse of the Fisher information matrix (FIM)[1] for a random vector-valued parameter [1] and provides a performance bound for any Bayesian estimator of such a parameter. In [2], Tichavsky *et al.* provided a recursive approach for calculating the sequential PCRLB for a general multi-dimensional discrete-time nonlinear filtering problem. The predictive and smoothing Cramér-Rao lower bounds for discrete-time nonlinear dynamic systems and their relations with filtering CRLB were discussed in [3]. However, the useful measurement information is averaged out making the unconditional PCRLB [2], [3] an off-line bound which is independent of the measurement history and the trajectory realization up to the current time. Therefore, the unconditional PCRLB does not reflect the nonlinear filtering performance for a particular system state realization very faithfully.

There are several modified versions of the PCRLB proposed in the literature. They attempt to make the bound adaptive to the realization of the system state, so that it can be useful for online sensor management. In [4], a renewal strategy was used to restart the recursive unconditional PCRLB evaluation process, where the initial time is reset to a more recent past time, such that the prior knowledge of the initial system state is more useful and relevant to the sensor management problem. Therefore, the resulting PCRLB is conditioned on the measurements up to the reset initial time. Based on the PCRLB modified in this manner, a sensor deployment approach was developed to achieve better tracking accuracy with the efficient use of limited sensor resources. When a particle filter is used in the renewal strategy, the posterior probability density function (pdf) of the system state at the reset initial time is represented nonparametrically by a set of particles, from which it is difficult to derive the exact FIM. One may use Gaussian approximation as was done in [4], and then the FIM at the reset initial time can be taken as the inverse of the empirical covariance matrix estimated based on the particles. However, Gaussian approximation may incur large errors and discrepancy, especially in a highly nonlinear and non-Gaussian system. Another modified version of the PCRLB, motivated by the problem of adaptive radar waveform design for target tracking, has been presented in [5]. The authors in [5] consider a linear Gaussian state dynamic model and a nonlinear measurement model, and propose to retain the unconditional recursive PCRLB derived in [2] with the exception of one term which corresponds to the contribution of the future measurements to the Fisher Information. The term with the future measurement contribution is modified in a heuristic way so that it includes the measurement history. Although the proposed method was shown to result in good performance for adaptive waveform design, the authors did not provide any theoretical justification for this modification. Most recently, the notion of conditional PCRLB was introduced in [6], which was shown to be different from the modified PCRLBs proposed in [4] and [5]. The conditional PCRLB proposed in [6] provides a bound on the conditional mean squared error (MSE) of the system state estimate, based on the measurements up to the current time. Furthermore, the authors in [6] proposed a systematic recursive approach based on a certain approximation to evaluate the conditional PCRLB.

Our contributions in this paper are as follows. Using two different expressions for the conditional PCRLB, we propose two new conditional PCRLBs. The first bound we propose is based on the representation of the conditional PCRLB proposed in [6]. We call this bound the alternative conditional PCRLB (A-CPCRLB), since we discard the auxiliary FIM which is involved in the recursive update for the conditional PCRLB presented in [6]. Instead, an alternative approximate recursive update is proposed, which is direct, more compact and computationally efficient than the one proposed in [6]. Furthermore, when the state dynamic model is linear and Gaussian, we show that this bound reduces to the modified PCRLB proposed in [5]. Hence, the proposed A-CPCRLB provides a generalization and the theoretical justification for the bound used in [5]. The second bound we propose builds on our earlier work [7] and is based on the direct representation of the conditional PCRLB without using any recursions as in [6]. We call this bound the direct conditional PCLRB (D-CPCRLB). Note that, a certain approximation was made in [6], to enable recursive computation of the conditional PCRLB. Due to the recursions, it is possible for the approximation error to accumulate through propagation over time and result in divergence. In comparison, D-CPCRLB is not recursive in nature and it is based on the target state distribution at the current time only. We also provide an approximate computation of the exact bound, namely the D-CPCRLB with Gaussian approximation, which is computationally the least expensive one. Even though the approximation naturally induces an error, it does not propagate with time unlike the CPCRLB in [6] or the proposed A-CPCRLB. This new bound uses the prediction distribution as its prior, and, therefore, it utilizes the information contained in the measurement data up to the current time for a

Manuscript received December 19, 2011; revised May 16, 2012; accepted May 29, 2012. Date of publication June 22, 2012; date of current version September 11, 2012. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Petr Tichavsky. This work was supported in part by the U.S. Air Force Office of Scientific Research by Grant FA9550-10-1-0263. This work was presented in part at the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2011), Prague, Czech Republic, May 2011.

Y. Zheng and P. K. Varshney are with the Center for Science and Technology, Department of EECS, Syracuse University, Syracuse, NY 13244 USA (e-mail: yuzheng@syr.edu; varshney@syr.edu).

O. Ozdemir is with ANDRO Computational Solutions, Rome, NY 13440 USA (e-mail: oozdemir@androcs.com).

R. Niu is with the Department of Electrical and Computer Engineering, Virginia Commonwealth University, Richmond, VA 23284 USA (e-mail: rniu@vcu.edu.).

Digital Object Identifier 10.1109/TSP.2012.2205686

[1] Throughout this paper, FIM refers to the Bayesian information matrix which is used for random unknown vectors.

1053-587X/$31.00 © 2012 IEEE

particular realization of the state trajectory making it a track-adaptive bound. Hence, it provides a more accurate and effective real-time performance evaluation than the traditional PCRLB.

The analytical calculation of our proposed bounds is not tractable except for very restricted cases such as linear Gaussian systems. For this reason, we develop numerical computation methods to compute these new bounds using the sequential Monte Carlo methods, i.e., particle filters [8], and provide analysis of computational complexities associated with our bounds. Our particle based computation methods make the proposed bounds easily computable in real-time from the particles already available from the underlying particle filter which is used to sequentially estimate the state. In our earlier work [7], we did not provide an illustrative example that depicts the performance of D-CPCRLB in a nonlinear Bayesian sequential estimation problem. Here, after presenting derivations of the proposed bounds, we provide a benchmark numerical example to compare the original CPCRLB [6] with our proposed bounds, namely the A-CPCRLB and the D-CPCRLB. For this particular example, we observe that the results are quite similar. Although the numerical example chosen in this paper is a benchmark problem widely used in the nonlinear tracking literature, the proposed bounds can be applied to real world problems, such as target tracking with radar systems. In fact, the authors in [5] used a special case of the proposed A-CPCRLB to dynamically design polarized waveforms for a target tracking problem using radar measurements. Moreover, a bound similar to our proposed D-CPCRLB with Gaussian approximation was used by the authors in [4] to deploy sensors in a submarine tracking problem. Therefore, the proposed bounds in this paper not only provide state realization-specific performance bounds for a sequential Bayesian estimation problem, but they also provide us with tools that can be used as performance metrics for dynamic sensor management problems in real-world scenarios.

II. Conditional Posterior Cramér-Rao Lower Bounds

Consider a n_x-dimensional state vector at time k, $\mathbf{x}_k$, whose discrete time dynamics is defined by

$$\mathbf{x}_{k+1} = f_k(\mathbf{x}_k, \mathbf{u}_k) \tag{1}$$

where $f_k : \Re^{n_x} \times \Re^{n_u} \rightarrow \Re^{n_x}$ and $\mathbf{u}_k$ is the white process noise with dimension n_u. The measurement model is given by

$$\mathbf{z}_k = h_k(\mathbf{x}_k, \mathbf{v}_k), \tag{2}$$

where $h_k : \Re^{n_x} \times \Re^{n_v} \rightarrow \Re^{n_z}$, $\mathbf{v}_k$ is the white measurement noise, n_z and n_v are the dimensions of the measurement and measurement noise vectors, respectively. The process and the measurement noise distributions are denoted by $p_{\mathbf{u}_k}(\mathbf{u})$ and $p_{\mathbf{v}_k}(\mathbf{v})$, respectively. It is assumed that the estimator has complete information about the state dynamic model (1), the sensor measurement model (2) and the process and measurement noise distributions.

The conditional PCRLB sets a bound on the performance of estimating the state vector up to time $k+1$, $\mathbf{x}_{0:k+1}$, when the new measurement $\mathbf{z}_{k+1}$ becomes available given that the past measurements up to time k, $\mathbf{z}_{1:k}$, are all known, i.e., the measurements up to time k are taken as realizations rather than random vectors. The sequence of conditional Fisher information $\{L(\mathbf{x}_{k+1} \,|\, \mathbf{z}_{1:k})\}$ for estimating state vector $\{\mathbf{x}_{k+1}\}$ given the measurements up to time k can be computed as follows [6]:

$$L(\mathbf{x}_{k+1} \,|\, \mathbf{z}_{1:k}) = B_k^{22} - B_k^{21} \left[B_k^{11} + L_A(\mathbf{x}_k \,|\, \mathbf{z}_{1:k})\right]^{-1} B_k^{12} \tag{3}$$

where

$$\begin{aligned}
B_k^{11} &= E_{p_{k+1}^c} \left\{-\triangle_{\mathbf{x}_k}^{\mathbf{x}_k} \ln p(\mathbf{x}_{k+1} \,|\, \mathbf{x}_k)\right\} && (4)\\
B_k^{12} &= E_{p_{k+1}^c} \left\{-\triangle_{\mathbf{x}_k}^{\mathbf{x}_{k+1}} \ln p(\mathbf{x}_{k+1} \,|\, \mathbf{x}_k)\right\} \\
&= \left(B_k^{21}\right)^T && (5)\\
B_k^{22} &= E_{p_{k+1}^c} \left\{-\triangle_{\mathbf{x}_{k+1}}^{\mathbf{x}_{k+1}} [\ln p(\mathbf{x}_{k+1} \,|\, \mathbf{x}_k) \right.\\
&\qquad \left. + \ln p(\mathbf{z}_{k+1} \,|\, \mathbf{x}_{k+1})]\right\} && (6)
\end{aligned}$$

and

$$\begin{aligned}
L_A(\mathbf{x}_k \,|\, \mathbf{z}_{1:k}) &= A_k^{22} - A_k^{21} \left(A_k^{11}\right)^{-1} A_k^{12} && (7)\\
p_{k+1}^c &\triangleq p(\mathbf{x}_{0:k+1}, \mathbf{z}_{k+1} \,|\, \mathbf{z}_{1:k}) && (8)
\end{aligned}$$

with

$$\begin{aligned}
A_k^{11} &= E_{p(\mathbf{x}_{0:k} \,|\, \mathbf{z}_{1:k})} \left[-\triangle_{\mathbf{x}_{0:k-1}}^{\mathbf{x}_{0:k-1}} \ln p(\mathbf{x}_{0:k} \,|\, \mathbf{z}_{1:k})\right]\\
A_k^{12} &= E_{p(\mathbf{x}_{0:k} \,|\, \mathbf{z}_{1:k})} \left[-\triangle_{\mathbf{x}_{0:k-1}}^{\mathbf{x}_k} \ln p(\mathbf{x}_{0:k} \,|\, \mathbf{z}_{1:k})\right] = (A_k^{21})^T\\
A_k^{22} &= E_{p(\mathbf{x}_{0:k} \,|\, \mathbf{z}_{1:k})} \left[-\triangle_{\mathbf{x}_k}^{\mathbf{x}_k} \ln p(\mathbf{x}_{0:k} \,|\, \mathbf{z}_{1:k})\right]
\end{aligned}$$

An approximate recursion to compute $L_A(\mathbf{x}_k \,|\, \mathbf{z}_{1:k})$ is also proposed in [6], which is

$$L_A(\mathbf{x}_k \,|\, \mathbf{z}_{1:k}) \approx S_k^{22} - S_k^{21} \left[S_k^{11} + L_A(\mathbf{x}_{k-1} \,|\, \mathbf{z}_{1:k-1})\right]^{-1} S_k^{12} \tag{9}$$

where

$$\begin{aligned}
S_k^{11} &= E_{p(\mathbf{x}_{0:k} \,|\, \mathbf{z}_{1:k})} \left[-\triangle_{\mathbf{x}_{k-1}}^{\mathbf{x}_{k-1}} \ln p(\mathbf{x}_k \,|\, \mathbf{x}_{k-1})\right] && (10)\\
S_k^{12} &= E_{p(\mathbf{x}_{0:k} \,|\, \mathbf{z}_{1:k})} \left[-\triangle_{\mathbf{x}_{k-1}}^{\mathbf{x}_k} \ln p(\mathbf{x}_k \,|\, \mathbf{x}_{k-1})\right] = \left(S_k^{21}\right)^T \\
&&& (11)\\
S_k^{22} &= E_{p(\mathbf{x}_{0:k} \,|\, \mathbf{z}_{1:k})} \left\{-\triangle_{\mathbf{x}_k}^{\mathbf{x}_k} [\ln p(\mathbf{x}_k \,|\, \mathbf{x}_{k-1}) \right.\\
&\qquad \left. + \ln p(\mathbf{z}_k \,|\, \mathbf{x}_k)]\right\} && (12)
\end{aligned}$$

One should note that if (9) is used in (3), then (3) becomes an approximation, instead of equality.

III. Proposed Conditional PCRLBs

A. A-CPCRLB

As shown in Section II, an approximated recursive update of an auxiliary FIM is necessary to recursively compute the conditional Fisher information at each time step, which makes the process complex. In this section, an alternative compact solution is proposed, such that the process of computation is simplified and computation time can be saved at the same time.

Proposition 1: The sequence of conditional Fisher information $L(\mathbf{x}_{k+1} \,|\, \mathbf{z}_{1:k})$ for estimating state vectors $\mathbf{x}_{k+1}$ given measurements up to time k can be computed as follows:

$$L(\mathbf{x}_{k+1} \,|\, \mathbf{z}_{1:k}) \approx B_k^{22} - B_k^{21} \left[B_k^{11} + L(\mathbf{x}_k \,|\, \mathbf{z}_{1:k-1})\right]^{-1} B_k^{12} \tag{13}$$

where B_k^{11}, B_k^{12}, B_k^{21} and B_k^{22} are given by (4) through (6).

Proof: Since $p_k^c = p(\mathbf{x}_{0:k}, \mathbf{z}_k \,|\, \mathbf{z}_{1:k-1})$ according to (8), the conditional FIM given measurements up to time $k-1$ can be decomposed as follows:

$$\begin{aligned}
&I(\mathbf{x}_{0:k}, \mathbf{z}_k \,|\, \mathbf{z}_{1:k-1})\\
&\quad \triangleq \begin{bmatrix} O_k & P_k \\ P_k^T & Q_k \end{bmatrix}\\
&\quad = \begin{bmatrix} E\left\{-\triangle_{\mathbf{x}_{0:k-1}}^{\mathbf{x}_{0:k-1}} \ln p_k^c\right\} & E\left\{-\triangle_{\mathbf{x}_{0:k-1}}^{\mathbf{x}_k} \ln p_k^c\right\} \\ E\left\{-\triangle_{\mathbf{x}_k}^{\mathbf{x}_{0:k-1}} \ln p_k^c\right\} & E\left\{-\triangle_{\mathbf{x}_k}^{\mathbf{x}_k} \ln p_k^c\right\} \end{bmatrix} && (14)
\end{aligned}$$

Thus, by applying matrix inversion formula [9], the inverse of the lower-right block of $I^{-1}(\mathbf{x}_{0:k}, \mathbf{z}_k \,|\, \mathbf{z}_{1:k-1})$, i.e., the FIM for estimating $\mathbf{x}_k$ given the measurements up to $k-1$ is

$$L(\mathbf{x}_k \,|\, \mathbf{z}_{1:k-1}) = Q_k - P_k^T O_k^{-1} P_k$$

Now, considering Fisher information given measurements up to time k, we have

$$\begin{aligned} &I(\mathbf{x}_{0:k+1}, \mathbf{z}_{k+1} \,|\, \mathbf{z}_{1:k}) \\ &\quad = E\left[-1 \begin{bmatrix} \triangle_{\mathbf{x}_{0:k-1}}^{\mathbf{x}_{0:k-1}} & \triangle_{\mathbf{x}_{0:k-1}}^{\mathbf{x}_k} & \triangle_{\mathbf{x}_{0:k-1}}^{\mathbf{x}_{k+1}} \\ \triangle_{\mathbf{x}_k}^{\mathbf{x}_{0:k-1}} & \triangle_{\mathbf{x}_k}^{\mathbf{x}_k} & \triangle_{\mathbf{x}_k}^{\mathbf{x}_{k+1}} \\ \triangle_{\mathbf{x}_{k+1}}^{\mathbf{x}_{0:k-1}} & \triangle_{\mathbf{x}_{k+1}}^{\mathbf{x}_k} & \triangle_{\mathbf{x}_{k+1}}^{\mathbf{x}_{k+1}} \end{bmatrix} \ln p_{k+1}^c \right] \end{aligned} \tag{15}$$

where p_{k+1}^c is defined in (8), which can be decomposed as follows:

$$\begin{aligned} &p(\mathbf{x}_{0:k+1}, \mathbf{z}_{k+1} \,|\, \mathbf{z}_{1:k}) \\ &\quad = p(\mathbf{z}_{k+1} \,|\, \mathbf{x}_{k+1}) p(\mathbf{x}_{k+1} \,|\, \mathbf{x}_k) \frac{p(\mathbf{x}_{0:k}, \mathbf{z}_k \,|\, \mathbf{z}_{1:k-1})}{p(\mathbf{z}_k \,|\, \mathbf{z}_{1:k-1})} \end{aligned} \tag{16}$$

Therefore

$$\begin{aligned} \ln p_{k+1}^c = \ln p(\mathbf{z}_{k+1} \,|\, \mathbf{x}_{k+1}) + \ln p(\mathbf{x}_{k+1} \,|\, \mathbf{x}_k) \\ + \ln p_k^c - \ln p(\mathbf{z}_k \,|\, \mathbf{z}_{1:k-1}). \end{aligned}$$

Hence, we have

$$\begin{aligned} &I(\mathbf{x}_{0:k+1}, \mathbf{z}_{k+1} \,|\, \mathbf{z}_{1:k}) \\ &= \begin{bmatrix} -E_{p_{k+1}^c} \triangle_{\mathbf{x}_{0:k-1}}^{\mathbf{x}_{0:k-1}} \ln p_k^c & -E_{p_{k+1}^c} \triangle_{\mathbf{x}_{0:k-1}}^{\mathbf{x}_k} \ln p_k^c & 0 \\ -E_{p_{k+1}^c} \triangle_{\mathbf{x}_k}^{\mathbf{x}_{0:k-1}} \ln p_k^c & -E_{p_{k+1}^c} \triangle_{\mathbf{x}_k}^{\mathbf{x}_k} \ln p_k^c + B_k^{11} & B_k^{12} \\ 0 & B_k^{21} & B_k^{22} \end{bmatrix} \end{aligned}$$

where $B_k^{ij}, \quad i = 1, 2, \; j = 1, 2$ are defined in (4) through (6). Since the top-left submatrix is a function of $\mathbf{z}_k$, we can approximate it by its expectation with respect to $p(\mathbf{z}_k \,|\, \mathbf{z}_{1:k-1})$. Then,

$$\begin{aligned} &- E_{p_{k+1}^c} \Delta_{\mathbf{x}_{0:k-1}}^{\mathbf{x}_{0:k-1}} \ln p_k^c \\ &\approx -E_{p(\mathbf{z}_k \,|\, \mathbf{z}_{1:k-1})} E_{p_{k+1}^c} \Delta_{\mathbf{x}_{0:k-1}}^{\mathbf{x}_{0:k-1}} \ln p_k^c \\ &= -\int p(\mathbf{z}_k \,|\, \mathbf{z}_{1:k-1}) p_{k+1}^c \Delta_{\mathbf{x}_{0:k-1}}^{\mathbf{x}_{0:k-1}} \ln p_k^c d\mathbf{x}_{0:k+1} d\mathbf{z}_{k+1} d\mathbf{z}_k \quad (17) \\ &\overset{(a)}{=} O_k. \quad (18) \end{aligned}$$

Note that (a) follows from plugging (16) in (17), and using the definition in (14). Similarly,

$$\begin{aligned} - E_{p_{k+1}^c} \Delta_{\mathbf{x}_{0:k-1}}^{\mathbf{x}_k} \ln p_k^c \approx P_k \\ - E_{p_{k+1}^c} \Delta_{\mathbf{x}_k}^{\mathbf{x}_k} \ln p_k^c \approx Q_k \end{aligned}$$

Hence

$$I(\mathbf{x}_{0:k+1}, \mathbf{z}_{k+1} \,|\, \mathbf{z}_{1:k}) \approx \begin{bmatrix} O_k & P_k & 0 \\ P_k^T & Q_k + B_k^{11} & B_k^{12} \\ 0 & B_k^{21} & B_k^{22} \end{bmatrix}.$$

The conditional Fisher information $L(\mathbf{x}_{k+1} \,|\, \mathbf{z}_{1:k})$ is equal to the inverse of the lower right submatrix of $I^{-1}(\mathbf{x}_{0:k+1}, \mathbf{z}_{k+1} \,|\, \mathbf{z}_{1:k})$. Therefore, according to the matrix inversion formula,

$$\begin{aligned} &L(\mathbf{x}_{k+1} \,|\, \mathbf{z}_{1:k}) \\ &\approx B_k^{22} - [0 \quad B_k^{21}] \begin{bmatrix} O_k & P_k \\ P_k^T & Q_k + B_k^{11} \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ B_k^{12} \end{bmatrix} \\ &= B_k^{22} - B_k^{21} \left(B_k^{11} + \left(Q_k - P_k^T O_k^{-1} P_k \right) \right)^{-1} B_k^{12} \\ &= B_k^{22} - B_k^{21} \left(B_k^{11} + L(\mathbf{x}_k \,|\, \mathbf{z}_{1:k-1}) \right)^{-1} B_k^{12} \end{aligned}$$

■

Based on Proposition 1, it is easy to show that the modified PCRLB in [5] is a special case of the A-CPCRLB, as stated in the following corollary.

Corollary 1: For the special case of linear state model with additive Gaussian noise, i.e., $\mathbf{x}_{k+1} = F_k \mathbf{x}_k + \mathbf{u}_k$, the conditional Fisher information $L(\mathbf{x}_{k+1} \,|\, \mathbf{z}_{1:k})$ is given by

$$\begin{aligned} L(\mathbf{x}_{k+1} \,|\, \mathbf{z}_{1:k}) \approx \left(Q_k + F_k (L(\mathbf{x}_k \,|\, \mathbf{z}_{1:k-1}))^{-1} F_k^T \right)^{-1} \\ + E_{p_{k+1}^c} \left\{ -\triangle_{\mathbf{x}_{k+1}}^{\mathbf{x}_{k+1}} \ln p(\mathbf{z}_{k+1} \,|\, \mathbf{x}_{k+1}) \right\} \end{aligned} \tag{19}$$

where F_k is the state transition matrix of the system state equation, and Q_k is the covariance matrix of the additive Gaussian noise u_k.

Proof: Since we consider a linear state model with additive Gaussian noise

$$\begin{aligned} p(\mathbf{x}_{k+1} \,|\, \mathbf{x}_k) = &\frac{1}{(2\pi)^{\frac{n_x}{2}} |Q_k|^{\frac{1}{2}}} \\ &\times \exp \left\{ -\frac{1}{2} [\mathbf{x}_{k+1} - F_k \mathbf{x}_k]^T Q_k^{-1} [\mathbf{x}_{k+1} - F_k \mathbf{x}_k] \right\} \end{aligned}$$

Taking the logarithm of the probability density function (pdf) above, we can get

$$-\ln p(\mathbf{x}_{k+1} \,|\, \mathbf{x}_k) = c_0 + \frac{1}{2} [\mathbf{x}_{k+1} - F_k \mathbf{x}_k]^T Q_k^{-1} [\mathbf{x}_{k+1} - F_k \mathbf{x}_k]$$

where c_0 is a constant. Then, it is straightforward to get

$$B_k^{11} = F_k^T Q_k^{-1} F_k \tag{20}$$

$$B_k^{12} = -F_k^T Q_k^{-1} \tag{21}$$

$$B_k^{22} = Q_k^{-1} + E_{p_{k+1}^c} \left\{ -\triangle_{\mathbf{x}_{k+1}}^{\mathbf{x}_{k+1}} \ln p(\mathbf{z}_{k+1} \,|\, \mathbf{x}_{k+1}) \right\}. \tag{22}$$

Therefore, using (13), we have

$$\begin{aligned} &L(\mathbf{x}_{k+1} \,|\, \mathbf{z}_{1:k}) \\ &= Q_k^{-1} + E_{p_{k+1}^c} \left\{ -\triangle_{\mathbf{x}_{k+1}}^{\mathbf{x}_{k+1}} \ln p(\mathbf{z}_{k+1} \,|\, \mathbf{x}_{k+1}) \right\} \\ &\quad - Q_k^{-1} F_k \left(F_k^T Q_k^{-1} F_k + L(\mathbf{x}_k \,|\, \mathbf{z}_{1:k-1}) \right)^{-1} F_k^T Q_k^{-1} \\ &= \left(Q_k + F_k L^{-1}(\mathbf{x}_k \,|\, \mathbf{z}_{1:k-1}) F_k^T \right)^{-1} \\ &\quad + E_{p_{k+1}^c} \left\{ -\triangle_{\mathbf{x}_{k+1}}^{\mathbf{x}_{k+1}} \ln p(\mathbf{z}_{k+1} \,|\, \mathbf{x}_{k+1}) \right\} \end{aligned} \tag{23}$$

where the last equation is due to the application of Woodbury matrix identity [10]

$$(A + UCV)^{-1} = A^{-1} - A^{-1} U (V A^{-1} U + C^{-1})^{-1} V A^{-1} \tag{24}$$

where A, U, C, V are matrices with proper dimensions. ■

One should note that the result in Corollary 1 is the same as the one used in [5]. Hence, the approximation in [5] is a special case of Proposition 1. Moreover, since the bound proposed in [5] is a heuristic one and not theoretically justified therein, Corollary 1 in this paper provides a theoretical justification for it and Proposition 1 generalizes it.

Obviously, (13) is more compact than (3), since the conditional PCRLB is directly updated at each recursion in (13) without using the auxiliary FIM $L_A(\mathbf{x}_k \,|\, \mathbf{z}_{1:k})$. The computational efficiency of the A-CPCRLB will be analyzed in Section V in detail.

Another useful insight that can be deduced from Proposition 1 is that in a linear and Gaussian system, the A-CPCRLB is identical to the offline PCRLB.

Corollary 2: For the particular case of linear Gaussian dynamic model: $\mathbf{x}_{k+1} = F_k \mathbf{x}_k + \mathbf{u}_k$, $\mathbf{z}_k = H_k \mathbf{x}_k + \mathbf{v}_k$, where $\mathbf{u}_k$ and $\mathbf{v}_k$

are Gaussian noises with covariance matrices Q_k and R_k respectively, the recursive conditional Fisher information (13) in Proposition 1 is the same as the recursive offline Fisher information J_{k+1} proposed in [2], i.e., $J_{k+1} = L(\mathbf{x}_{k+1} \mid \mathbf{z}_{1:k})$, given that $J_0 = L(\mathbf{x}_0)$.

Proof: From Corollary 1, we already have (20), (21), and (22). Given the linear Gaussian observation model, we have

$$p(\mathbf{z}_{k+1} \mid \mathbf{x}_{k+1}) = \frac{1}{(2\pi)^{\frac{n_z}{2}} |R_{k+1}|^{\frac{1}{2}}} \times \exp\left\{ -\frac{1}{2} [\mathbf{z}_{k+1} - H_{k+1}\mathbf{x}_{k+1}]^T R_{k+1}^{-1} [\mathbf{z}_{k+1} - H_{k+1}\mathbf{x}_{k+1}] \right\}$$

Then,

$$\begin{aligned} E_{p^c_{k+1}} &\left\{ -\triangle_{\mathbf{x}_{k+1}}^{\mathbf{x}_{k+1}} \ln p(\mathbf{z}_{k+1} \mid \mathbf{x}_{k+1}) \right\} \\ &= E_{p^c_{k+1}} \left\{ H_{k+1}^T R_{k+1}^{-1} H_{k+1} \right\} \\ &= H_{k+1}^T R_{k+1}^{-1} H_{k+1} \end{aligned}$$

Thus,

$$B_k^{22} = Q_k^{-1} + H_{k+1}^T R_{k+1}^{-1} H_{k+1}$$

In [2], it has been shown that $J_{k+1} = D_k^{22} - D_k^{21}(J_k + D_k^{11})^{-1} D_k^{12}$, where $D_k^{11} = F_k^T Q_k^{-1} F_k$, $D_k^{12} = -F_k^T Q_k^{-1} = (D_k^{21})^T$, and $D_k^{22} = Q_k^{-1} + H_{k+1}^T R_{k+1}^{-1} H_{k+1}$ when the system is linear and Gaussian. Hence, $J_{k+1} = L(\mathbf{x}_{k+1} \mid \mathbf{z}_{1:k})$, given the same initialization, i.e., $J_0 = L(\mathbf{x}_0)$. ■

B. D-CPCRLB

For a given track at time k, all the information that is needed in order to estimate $\mathbf{x}_{k+1}$ is contained in the prediction distribution $p(\mathbf{x}_{k+1} \mid \mathbf{z}_{1:k})$, which is readily available from the tracking filter, and the measurement distribution $p(\mathbf{z}_{k+1} \mid \mathbf{x}_{k+1})$. If the tracking filter is a particle filter, $p(\mathbf{x}_{k+1} \mid \mathbf{z}_{1:k})$ is available in the form of propagated particles. Using this fact, we proposed a modified posterior CRLB [7], which we call the D-CPCRLB. We used the prediction distribution at time k, i.e., $p(\mathbf{x}_{k+1} \mid \mathbf{z}_{1:k})$, as our *a priori* distribution for time $k+1$. First, we present the following definitions:

Definition 1: Conditional estimator $\hat{\mathbf{x}}_{k+1}(\mathbf{z}_{k+1} \mid \mathbf{z}_{1:k})$ is defined as a function of the observed data $\mathbf{z}_{k+1}$ given the existing measurements $\mathbf{z}_{1:k}$.

Definition 2: Mean squared error of the conditional estimator at time $k+1$ is defined as follows:

$$\begin{aligned} \mathrm{MSE}(\hat{\mathbf{x}}_{k+1} \mid \mathbf{z}_{1:k}) &\triangleq E\{\tilde{\mathbf{x}}_{k+1}\tilde{\mathbf{x}}_{k+1}^T \mid \mathbf{z}_{1:k}\} \\ &= \int \tilde{\mathbf{x}}_{k+1}\tilde{\mathbf{x}}_{k+1}^T p_{k+1}^j d\mathbf{x}_{k+1} d\mathbf{z}_{k+1} \end{aligned} \tag{25}$$

where $\tilde{\mathbf{x}}_{k+1} \triangleq \hat{\mathbf{x}}_{k+1} - \mathbf{x}_{k+1}$ is the estimation error, and $p_{k+1}^j \triangleq p(\mathbf{x}_{k+1}, \mathbf{z}_{k+1} \mid \mathbf{z}_{1:k})$.

We denote D-CPCRLB as I_{k+1}^{-1} where

$$I_{k+1} = E_{p(\mathbf{z}_{k+1}, \mathbf{x}_{k+1} \mid \mathbf{z}_{1:k})} \left\{ -\Delta_{\mathbf{x}_{k+1}}^{\mathbf{x}_{k+1}} \ln p(\mathbf{z}_{k+1}, \mathbf{x}_{k+1} \mid \mathbf{z}_{1:k}) \right\}. \tag{26}$$

From the posterior Cramér-Rao inequality, $\mathrm{MSE}(\hat{\mathbf{x}}_{k+1} \mid \mathbf{z}_{1:k})$ is bounded as follows:

$$\mathrm{MSE}(\hat{\mathbf{x}}_{k+1} \mid \mathbf{z}_{1:k}) \geq I_{k+1}^{-1} \tag{27}$$

Using the fact that $\mathbf{z}_{1:k}$, $\mathbf{x}_{k+1}$ and $\mathbf{z}_{k+1}$ form a first order Markov chain, the joint conditional density in (26) can be written as

$$p(\mathbf{z}_{k+1}, \mathbf{x}_{k+1} \mid \mathbf{z}_{1:k}) = p(\mathbf{z}_{k+1} \mid \mathbf{x}_{k+1}) p(\mathbf{x}_{k+1} \mid \mathbf{z}_{1:k}). \tag{28}$$

Then I_{k+1} in (26) can be decomposed into two parts as

$$I_{k+1} = I_{k+1}^D + I_{k+1}^P \tag{29}$$

where I_{k+1}^D represents the information gained from the new measurements averaged over the *a priori* distribution, and I_{k+1}^P represents the information contained in the *a priori* distribution

$$I_{k+1}^D = E_{p(\mathbf{z}_{k+1} \mid \mathbf{x}_{k+1}) p(\mathbf{x}_{k+1} \mid \mathbf{z}_{1:k})} \left\{ -\Delta_{\mathbf{x}_{k+1}}^{\mathbf{x}_{k+1}} \ln p(\mathbf{z}_{k+1} \mid \mathbf{x}_{k+1}) \right\}, \tag{30}$$

$$I_{k+1}^P = E_{p(\mathbf{x}_{k+1} \mid \mathbf{z}_{1:k})} \left\{ -\Delta_{\mathbf{x}_{k+1}}^{\mathbf{x}_{k+1}} \ln p(\mathbf{x}_{k+1} \mid \mathbf{z}_{1:k}) \right\}. \tag{31}$$

Note that the expectations in (30)–(31) are taken with respect to the joint conditional density $p(\mathbf{z}_{k+1}, \mathbf{x}_{k+1} \mid \mathbf{z}_{1:k})$. Therefore, I_{k+1}^D in (29) is in fact an expectation of the standard FIM over the prediction distribution $p(\mathbf{x}_{k+1} \mid \mathbf{z}_{1:k})$, where the standard FIM is defined by

$$J_{k+1}^S(\mathbf{x}_{k+1}) = E_{p(\mathbf{z}_{k+1} \mid \mathbf{x}_{k+1})} \left\{ -\Delta_{\mathbf{x}_{k+1}}^{\mathbf{x}_{k+1}} \ln p(\mathbf{z}_{k+1} \mid \mathbf{x}_{k+1}) \right\}. \tag{32}$$

Hence,

$$I_{k+1}^D = E_{p(\mathbf{x}_{k+1} \mid \mathbf{z}_{1:k})} \left\{ J_{k+1}^S(\mathbf{x}_{k+1}) \right\}. \tag{33}$$

For a given track, D-CPCRLB can be calculated by adding I_{k+1}^D and I_{k+1}^P given in (30) and (31), respectively.

IV. Computation of the Proposed Conditional PCRLBs Using Particle Filters

In this section, particle filter based methods to compute the two modified PCRLBs proposed in this paper are given.

A. Particle Filter Computation of A-CPCRLB

We use the same particle filter based method as that given in [6] to compute the A-CPCRLB. The details are not given here for the sake of brevity. Interested readers are referred to [6] for more information.

B. Particle Filter Computation of D-CPCRLB

Note that the distribution of $\mathbf{x}_{k+1}$ is represented by the propagated particles, $\mathbf{x}_{k+1}^{(j)} \sim p(\mathbf{x}_{k+1} \mid \mathbf{z}_{1:k})$. In a standard particle filtering algorithm, the propagated particles are computed as

$$\mathbf{x}_{k+1}^{(j)} = f_k\left(\mathbf{x}_k^{(j)}, \mathbf{u}_k^{(j)}\right) \tag{34}$$

where $\mathbf{u}_k^{(j)} \sim p_{\mathbf{u}_k}(\mathbf{u})$. Therefore, the prediction distribution is approximated using the weighted propagated particles $\{\mathbf{x}_{k+1}^{(j)}, w_k^{(j)}\}_{j=1}^M$, i.e.,

$$p(\mathbf{x}_{k+1} \mid \mathbf{z}_{1:k}) \approx \sum_{j=1}^M w_k^{(j)} \delta\left(\mathbf{x}_{k+1} - \mathbf{x}_{k+1}^{(j)}\right). \tag{35}$$

Using these propagated particles, we can approximate I_{k+1}^D as follows:

$$I_{k+1}^D \approx \sum_{j=1}^M w_k^{(j)} J_{k+1}^S\left(\mathbf{x}_{k+1}^{(j)}\right), \tag{36}$$

where J_{k+1}^S is the standard Fisher Information quantity defined in (32), which in many cases has a closed-form solution.

In order to compute the second term in (29), i.e., (31), $p(\mathbf{x}_{k+1}\,|\,\mathbf{z}_{1:k})$ should have a parametric expression. However, in our case, we have an approximation based on particles. One way to approximate I_{k+1}^{P} is to use a simple Gaussian approximation for the prediction distribution as proposed in [7], i.e., $p(\mathbf{x}_{k+1}\,|\,\mathbf{z}_{1:k}) \approx \mathcal{N}(\mu_k, \mathbf{\Sigma}_k)$, where

$$\mu_k \approx \sum_{j=1}^{M} w_k^{(j)} \mathbf{x}_{k+1}^{(j)}, \tag{37}$$

$$\mathbf{\Sigma}_k \approx \sum_{j=1}^{M} w_k^{(j)} \left(\mathbf{x}_{k+1}^{(j)} - \mu_k\right)\left(\mathbf{x}_{k+1}^{(j)} - \mu_k\right)^T. \tag{38}$$

Then, I_{k+1}^{P} is approximated by the inverse of the covariance matrix in (38),

$$I_{k+1}^{P} \approx \mathbf{\Sigma}_k^{-1}. \tag{39}$$

The difference between this approach and the bound used in [4] is the following. In [4], the authors adopt the unconditional PCRLB with the exception that, at time k, the initial FIM is updated by using the updated particles at that particular time. This update is carried out by using a Gaussian approximation, i.e., by computing the covariance of the updated particles. This new FIM is then plugged into the recursive unconditional PCRLB equations to compute the FIM at time $k+1$ and onwards. However, in our approach, at time k, we first propagate the particles and then compute the prior FIM by using a Gaussian approximation, i.e., by computing the covariance of the propagated particles. This procedure eliminates the need for computing the recursive unconditional PCRLB equations, which require the computation of extra terms as well as extra matrix operations.

Although a Gaussian approximation is appealing due to its simplicity, it may not represent the actual distribution $p(\mathbf{x}_{k+1}\,|\,\mathbf{z}_{1:k})$ for highly nonlinear scenarios, especially for multi-modal distributions and/or distributions with heavy tails. In those cases, the Gaussian approximation will provide inaccurate results for I_{k+1}^{P}. For this reason, inspired by the method proposed in [11] to compute entropy based on the particle filter, we propose a numerical computation method which can approximate I_{k+1}^{P} by using the particles already available through the particle filter. We start deriving our approximation by first rewriting the expression for I_{k+1}^{P} (31):

$$\begin{aligned} I_{k+1}^{P} &= E_{p(\mathbf{x}_{k+1}\,|\,\mathbf{z}_{1:k})}\left\{-\Delta_{\mathbf{x}_{k+1}}^{\mathbf{x}_{k+1}} \ln\, p(\mathbf{x}_{k+1}\,|\,\mathbf{z}_{1:k})\right\} \\ &\approx -\sum_{l=1}^{M}\left\{\Delta_{\mathbf{x}_{k+1}^{(l)}}^{\mathbf{x}_{k+1}^{(l)}} \ln\, p\left(\mathbf{x}_{k+1}^{(l)}\middle|\,\mathbf{z}_{1:k}\right)\right\} w_k^{(l)}. \end{aligned} \tag{40}$$

The second step of (40) is due to the approximation in (35). Note that $p(\mathbf{x}_{k+1}\,|\,\mathbf{z}_{1:k})$ is given as

$$\begin{aligned} p(\mathbf{x}_{k+1}\,|\,\mathbf{z}_{1:k}) &= \int_{\mathbf{x}} p(\mathbf{x}_{k+1}\,|\,\mathbf{x}_k)p(\mathbf{x}_k\,|\,\mathbf{z}_{1:k})d\mathbf{x}_k \\ &\approx \sum_{j=1}^{M} p\left(\mathbf{x}_{k+1}\,|\,\mathbf{x}_k^{(j)}\right) w_k^{(j)}. \end{aligned} \tag{41}$$

Using (41) in (40), the elements of the I_{k+1}^{P} can be computed as

$$\begin{aligned} &\left[I_{k+1}^{P}\right]_{mn} \\ &\approx -\sum_{l=1}^{M}\left\{\frac{\partial^2 \ln\left(\sum_{j=1}^{M} p\left(\mathbf{x}_{k+1}^{(l)}\middle|\,\mathbf{x}_k^{(j)}\right) w_k^{(j)}\right)}{\partial x_{k+1_m}^{(l)}\partial x_{k+1_n}^{(l)}}\right\} w_k^{(l)}. \end{aligned} \tag{42}$$

Further simplifying (42) results in (43) and (44):

$$\begin{aligned} &\left[I_{k+1}^{P}\right]_{mn} \\ &\approx -\sum_{l=1}^{M}\left\{\frac{\frac{\partial^2\phi\left(\mathbf{x}_{k+1}^{(l)}\right)}{\partial x_{k+1_m}^{(l)}\partial x_{k+1_n}^{(l)}}\phi\left(\mathbf{x}_{k+1}^{(l)}\right) - \frac{\partial\phi\left(\mathbf{x}_{k+1}^{(l)}\right)}{\partial x_{k+1_m}^{(l)}}\frac{\partial\phi\left(\mathbf{x}_{k+1}^{(l)}\right)}{\partial x_{k+1_n}^{(l)}}}{\phi^2\left(\mathbf{x}_{k+1}^{(l)}\right)}\right\} w_k^{(l)} \end{aligned} \tag{43}$$

where

$$\phi\left(\mathbf{x}_{k+1}^{(l)}\right) = \sum_{j=1}^{M} p\left(\mathbf{x}_{k+1}^{(l)}\middle|\,\mathbf{x}_k^{(j)}\right) w_k^{(j)}. \tag{44}$$

Note that in order for the approximation in (43) to be analytically tractable, the function f defined by $f(\mathbf{x}_{k+1}) \triangleq p(\mathbf{x}_{k+1}\,|\,\mathbf{x}_k)$ should be twice differentiable in $\Re^{n_x}$. As long as $p(\mathbf{x}_{k+1}\,|\,\mathbf{x}_k)$ satisfy this regularity condition, I_{k+1}^{P} given in (43) can be computed using the particles readily available through the particle filter which is used to track the unknown system state. Note that this regularity condition is satisfied for most of the real tracking scenarios.

V. Computational Complexity

In this section, we analyze the computational complexity of the proposed bounds based on the total number of the floating-point operations (*flops*).

The exact flops required for the derivative operations in (4)–(12) depend on the structures of the pdfs $p(\mathbf{x}_{k+1}\,|\,\mathbf{x}_k)$ and $p(\mathbf{z}_k\,|\,\mathbf{x}_k)$ and there is no universal count. Due to this dependence, we define new notations to represent the flops for these derivative operations: $\Delta_1^s \triangleq \mathrm{fl}(\frac{\partial p(\mathbf{x}_{k+1}\,|\,\mathbf{x}_k)}{\partial x_{k+1_i}})$, $\Delta_2^s \triangleq \mathrm{fl}(\frac{\partial^2 p(\mathbf{x}_{k+1}\,|\,\mathbf{x}_k)}{\partial x_{k+1_i}\partial x_{k+1_j}})$, $\Delta_3^s \triangleq \mathrm{fl}(\frac{\partial p(\mathbf{x}_{k+1}\,|\,\mathbf{x}_k)}{\partial x_{k_i}})$, $\Delta_4^s \triangleq \mathrm{fl}(\frac{\partial^2 p(\mathbf{x}_{k+1}\,|\,\mathbf{x}_k)}{\partial x_{k_i}\partial x_{k_j}})$, $\Delta_5^s \triangleq \mathrm{fl}(\frac{\partial^2 p(\mathbf{x}_{k+1}\,|\,\mathbf{x}_k)}{\partial x_{k+1_i}\partial x_{k_j}})$, $\Delta_1^z \triangleq \mathrm{fl}(\frac{\partial p(\mathbf{z}_k\,|\,\mathbf{x}_k)}{\partial \mathbf{x}_{k_i}})$, $\Delta_2^z \triangleq \mathrm{fl}(\frac{\partial^2 p(\mathbf{z}_k\,|\,\mathbf{x}_k)}{\partial \mathbf{x}_{k_i}\partial \mathbf{x}_{k_j}})$, $1 \leq i, j \leq n_x$, where $\mathrm{fl}(\,\cdot\,)$ represents the number of flops required for a given operation. When defining these notations, for simplicity, we have made the implicit assumption that the derivatives with respect to different elements of the state vector require the same flops. In the following calculations, we also assume that each particle has a non-identical weight, i.e., there is no resampling.

We start with the calculation of flops required for the original CPCRLB in [6]. Note that the term B_k^{22} has two terms. Let us denote the first and the second terms in (6) as $B_k^{22,a}$ and $B_k^{22,b}$, respectively. Then, the flops of the CPCRLB, i.e., (3), can be represented as:

$$\begin{aligned} \mathrm{fl}(L_{k+1}) = {} & \mathrm{fl}\left(B_k^{22,a}\right) + \mathrm{fl}\left(B_k^{22,b}\right) + \mathrm{fl}\left(B_k^{12}\right) + \mathrm{fl}\left(B_k^{11}\right) \\ & + \mathrm{fl}\left(S_k^{11}\right) + \mathrm{fl}\left(S_k^{12}\right) + \mathrm{fl}\left(S_k^{22}\right) + \mathcal{O}\left(n_x^3\right). \end{aligned} \tag{45}$$

where $\mathcal{O}(n_x^3)$ represents the computational complexity associated with matrix inversions, multiplications and summations involved in (3) and (9). For matrix inversion, the exact flop count depends on the matrix and the specific technique used for inversion. Nevertheless, the flop count required for matrix inversion can be expressed as $\mathcal{O}(n_x^3)$ [12], which also subsumes the flops required for matrix multiplications and summations in (3) and (9). From the particle based computation in [6], we can calculate the flops required for the B terms and S terms as:

$$\mathrm{fl}\left(B_k^{22,a}\right) = \frac{3}{2}Mn_x^2 + Mn_x\Delta_1^s + \frac{3}{2}Mn_x + 2M - 1 \quad (46)$$

$$\mathrm{fl}\left(B_k^{12}\right) = \frac{3}{2}Mn_x^2 + Mn_x\Delta_3^s + \frac{3}{2}Mn_x + 2M - 1 \quad (47)$$

$$\mathrm{fl}\left(B_k^{11}\right) = \frac{3}{2}Mn_x^2 + \frac{3}{2}Mn_x + 2M - 1 \quad (48)$$

$$\begin{aligned}\mathrm{fl}\left(S_k^{11}\right) = &\left(\frac{5}{2} + \frac{\Delta_4^s}{2}\right)Mn_x^2\\ &+ \left(\frac{5}{2} + \Delta_3^s + \frac{\Delta_4^s}{2}\right)Mn_x + 2M - 1\end{aligned} \quad (49)$$

$$\mathrm{fl}\left(S_k^{12}\right) = \left(5 + \Delta_5^s\right)Mn_x^2 + \Delta_1^s Mn_x + M - 1 \quad (50)$$

$$\begin{aligned}\mathrm{fl}\left(S_k^{22}\right) = &\left(\frac{5}{2} + \frac{\Delta_2^z}{2}\right)Mn_x^2\\ &+ \left(\frac{5}{2} + \Delta_1^z + \frac{\Delta_2^z}{2}\right)Mn_x + 2M - 1\end{aligned} \quad (51)$$

By carefully investigating the computations required for the CPCRLB in [6], i.e., expression in (3) and (9), and that of the A-CPCRLB, i.e., expression in (13), we note that the extra computation of the CPCRLB at time k comes from the computations of S_k^{11}, S_k^{12} and S_k^{22}, as well as from a matrix inversion, multiplication and subtraction in (9). Therefore, the A-CPCRLB saves the following number of flops, compared to the CPCRLB in [6]:

$$\begin{aligned}\mathrm{fl}_A^{\mathrm{save}} &= \mathrm{fl}\left(S_k^{22}\right) + \mathrm{fl}\left(S_k^{11}\right) + \mathrm{fl}\left(S_k^{12}\right) + \mathcal{O}\left(n_x^3\right)\\ &= \left(10 + \frac{\Delta_4^s}{2} + \Delta_5^s + \frac{\Delta_2^z}{2}\right)Mn_x^2 + 5M + \mathcal{O}\left(n_x^3\right)\\ &\quad + \left(5 + \Delta_3^s + \frac{\Delta_4^s}{2} + \Delta_1^s + \Delta_1^z + \frac{\Delta_2^z}{2}\right)Mn_x.\end{aligned} \quad (52)$$

From the above expression, it is clear that the savings are significant especially for large M and/or n_x.

Next, we calculate the flops required to compute D-CPCRLB. From (6) and (36), we notice that $B_k^{22,b} = I_{k+1}^D$. In fact, the approximation proposed in [6] to compute $B_k^{22,b}$ is essentially identical to the computation of I_{k+1}^D given in (36), i.e., $\mathrm{fl}(B_k^{22,b}) = \mathrm{fl}(I_{k+1}^D)$. The inner expression inside the large brackets in (43) requires Mn_x first derivative operations and $Mn_x(n_x+1)/2$ second derivative operations. After adding the flops required for weighted summations, squaring operations and divisions, we arrive at the following:

$$\begin{aligned}\mathrm{fl}(I_{k+1}) = &\frac{1}{2}M^2\left(n_x^2 + n_x\right)\Delta_2^s + M^2 n_x\Delta_1^s + M^2 n_x^2\\ &+ 3M^2 n_x - \frac{1}{2}Mn_x^2 - \frac{3}{2}Mn_x + 7M - 1 + \mathrm{fl}\left(B_k^{22,b}\right).\end{aligned} \quad (53)$$

Next, we calculate the flops required for D-CPCRLB with Gaussian approximation. Let I_{k+1}^G denote the corresponding Bayesian FIM. We note that (37) requires $\mathrm{fl}(\mu_k) = 2Mn_x - n_x$. Combining computations in (37)–(38) with the matrix inversion results in the following:

$$\mathrm{fl}\left(I_{k+1}^G\right) = \frac{1}{2}Mn_x^2 + \frac{7}{2}Mn_x + 2M + \mathcal{O}\left(n_x^3\right) + \mathrm{fl}\left(B_k^{22,b}\right). \quad (54)$$

Both the original CPCRLB and the A-CPCRLB are derived using the whole time series, i.e., the conditional FIM of the whole time series from time $t = 0$ to $t = k + 1$ is considered. The key strategy during the derivation is matrix decomposition, due to which, the recursive formula is derived. However, D-CPCRLB is derived by using the current time as a snapshot, i.e., the FIM for the current time is explored directly, which avoids large matrix manipulations. However, the complexity of the D-CPCRLB computed without Gaussian approximation is $\mathcal{O}(M^2)$ according to (53), while the complexity of both the conditional PCRLB and A-CPCRLB is $\mathcal{O}(M)$. The complexity of the D-CPCRLB computed using the Gaussian approximation in (39) is $\mathcal{O}(M)$ and it does not require any heavy matrix manipulations, except for the inversion of the empirical covariance matrix. D-CPCRLB with Gaussian approximation saves the following number of flops compared to original CPCRLB:

$$\begin{aligned}\mathrm{fl}_{D,1}^{\mathrm{save}} = &4Mn_x^2 + Mn_x\left(\Delta_1^s + \Delta_3^s + 1\right) + \mathrm{fl}\left(S_k^{11}\right)\\ &+\mathrm{fl}\left(S_k^{12}\right) + \mathrm{fl}\left(S_k^{21}\right) + \mathrm{fl}\left(S_k^{22}\right) + 4M - 3.\end{aligned} \quad (55)$$

It is also clear that D-CPCRLB with Gaussian approximation saves the following number of flops with respect to A-CPCRLB:

$$\mathrm{fl}_{D,2}^{\mathrm{save}} = 4Mn_x^2 + Mn_x\left(\Delta_1^s + \Delta_3^s + 1\right) + \mathrm{fl}\left(S_k^{11}\right) + 4M - 3. \quad (56)$$

Finally, we note from (53) that the D-CPCRLB computed by using (43) is computationally the most expensive bound as its complexity is $\mathcal{O}(M^2)$ and as $M \gg n_x$ in a typical particle filtering problem.

VI. Numerical Results

In this section, we consider the univariate nonstationary growth model (UNGM), which is a highly nonlinear and bimodal model, and perform a number of numerical experiments to compare the performance of the following bounds: 1) Offline PCRLB [2], 2) Conditional PCRLB proposed in [6], 3) A-CPCRLB, 4) D-CPCRLB with Gaussian approximation, and 5) D-CPCRLB computed using (43). The UNGM has been widely used in the nonlinear tracking literature as a benchmark problem [6], [13], [14]. The dynamic state space equations for a UNGM are given by

$$x_{k+1} = \alpha x_k + \beta\frac{x_k}{1 + x_k^2} + \gamma\cos(1.2k) + u_k \quad (57)$$

$$z_k = \kappa x_k^2 + v_k \quad (58)$$

where u_k and v_k are zero mean white Gaussian with variances σ_u^2 and σ_v^2, respectively. The conditional mean-squared error (MSE) is calculated as follows. At time k, the posterior pdf is computed by the particle filter given the measurements up to time k. Then, 1000 independent realizations of z_{k+1} are generated according to (58), and the conditional MSE, (i.e., $\mathrm{MSE}(\hat{x}_{k+1} \mid z_{1:k})$), is obtained based on 1000 Monte Carlo runs. A single realization of z_{k+1} is randomly picked among the 1000 realizations above, and concatenated with the past measurement history to form $z_{1:k+1}$. The particles and weights corresponding to this particular z_{k+1} are stored and used for the iteration at time $(k+1)$.

A. Highly Nonlinear Case

In this experiment, we set the parameters for UNGM as $\alpha = 1$, $\beta = 15$, $\gamma = 8$, $\sigma_u^2 = 4$, $\sigma_v^2 = 1$ and $\kappa = 1/20$ to make it highly nonlinear. Since it is difficult for conventional methods such as the Kalman Filter or the extended Kalman Filter to track the state when the system model is highly nonlinear, a particle filter is applied in the simulation.

In Fig. 1, the conditional PCRLB in [6], A-CPCRLB derived in Section III-A, D-CPCRLB as well as D-CPCRLB with Gaussian approximation derived in Section III-B, the conditional MSE, and (unconditional/offline) PCRLB are plotted as functions of time. We can observe that all the three CPCRLBs follow the trends of the conditional MSE more faithfully than the (offline) PCRLB. It can also be observed

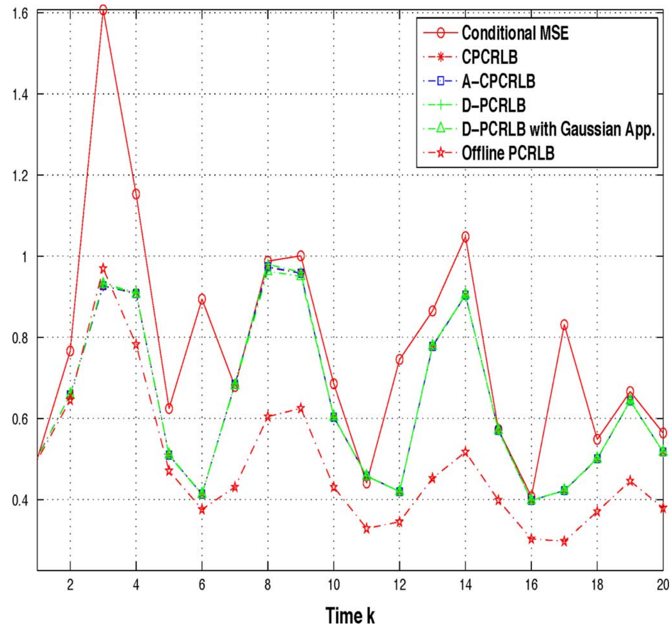

Fig. 1. Highly nonlinear case.

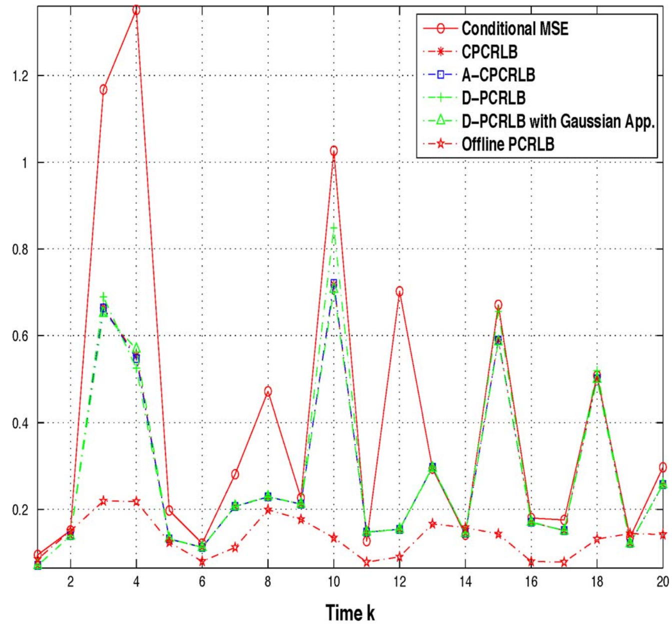

Fig. 2. Weakly nonlinear case.

that the original conditional PCRLB, A-CPCRLB, and D-CPCRLB almost overlap with each other everywhere.

B. Weakly Nonlinear Case

In order to show that the proposed recursive update procedure in this paper is independent of the nonlinearity of the state process model, a weakly nonlinear example is provided in this section. Here, we set $\beta = 1$, resulting in a much smaller nonlinear component in the state process equation. The variance of the process noise is set as $\sigma_u^2 = 1$, smaller than the highly nonlinear case. We set the measurement noise variance $\sigma_v^2 = 0.01$, such that the signal-to-noise ratio (SNR) is high for the observation. Other parameters are kept the same as in Section VI-A. Similar to Fig. 1, Fig. 2 illustrates that there is almost no difference between the original conditional PCRLB and the proposed bounds, and all the three CPCRLBs follow the trends of the conditional MSE more faithfully than the (offline) PCRLB.

C. Discussion

In our particular examples shown in Figs. 1 and 2, it can be observed that the offline PCRLB is more optimistic than the conditional PCRLBs. It should be mentioned that this is not always the case, since for some specific realizations of the system state, the conditional PCRLB may result in values that are smaller than the unconditional PCRLB. This is due to the fact that the online (conditional) bounds depend on a specific realization of the system state and they provide bounds for that specific conditional MSE. Figs. 1 and 2 each depicts one particular realization, i.e., single Monte Carlo run.

Simulation results show that the conditional bounds proposed in this paper are almost the same as the original one proposed in [6]. Since the original CPCRLB in [6] and the A-CPCRLB both use approximations at each iteration, it is possible that the error due to approximations accumulates over time. It is difficult to perform an exact error comparison between the two recursive procedures. Nevertheless, the following intuitive analysis is helpful to interpret the simulation results. In [6], the approximation is only applied to the top left block of the auxiliary FIM in (7), while in A-PCRLB the approximation is applied to four blocks of the conditional FIM in (15). However, in [6], the approximated block is involved in three inversions to complete the update at each iteration, which makes the approximation propagate to all the elements of the conditional PCRLB. Therefore, both approximations result in almost the same order of errors, which explains why the gap between the two corresponding lower bounds is negligible as shown by numerical examples.

VII. Conclusion

In this paper, two conditional PCRLBs have been proposed, namely the A-CPCRLB and the D-CPCRLB for nonlinear sequential Bayesian estimation. They achieve almost the same performance as the conditional PCRLB proposed in [6] as demonstrated by numerical examples. The proposed A-CPCRLB is more compact and more computationally efficient than the one in [6]. The proposed D-CPCRLB is not recursive in nature and it is an exact bound, while the D-CPCRLB with Gaussian approximation is computationally the least expensive. Even though the D-CPCRLB with Gaussian approximation induces an approximation error, the error does not propagate with time which may happen in the CPCRLB in [6] or in the A-CPCRLB. The proposed bounds do not have any general closed forms. Therefore, we provided numerical methods to compute our bounds using the particles available through the particle filter. Possible future work includes application of the proposed bounds to resource management problems for target tracking and exploration of alternative approximations to compute D-CPCRLB such as using spatial data structures [15], [16].

References

[1] H. V. Trees and K. Bell*, Bayesian Bounds for Parameter Estimation and Nonlinear Filtering/Tracking*. Hoboken, NJ: Wiley-IEEE , 2007.

[2] P. Tichavský, C. H. Muravchik, and A. Nehorai, "Posterior Cramér-Rao bounds for discrete-time nonlinear filtering," *IEEE Trans. Signal Process.*, vol. 46, no. 2, pp. 1386–1395, May 1998.

[3] M. Šimandl, J. Královec, and P. Tichavský, "Filtering, predictive, and smoothing Cramér-Rao bounds for discrete-time nonlinear dynamic systems," *Automatica.*, vol. 11, pp. 1703–1716, Apr. 2001.

[4] M. L. Hernandez, T. Kirubarajan, and Y. Bar-Shalom, "Multisensor resource deployment using posterior Cramér-Rao bounds," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 40, no. 2, pp. 399–416, Apr. 2004.

[5] M. Hurtado, T. Zhao, and A. Nehorai, "Adaptive polarized waveform design for target tracking based on sequential Bayesian inference," *IEEE Trans. Signal Process.*, vol. 56, no. 3, pp. 1120–1133, Mar. 2008.

[6] L. Zuo, R. Niu, and P. K. Varshney, "Conditional posterior Cramér-Rao lower bounds for nonlinear sequential Bayesian estimation," *IEEE Trans. Signal Process.*, vol. 59, no. 1, pp. 1–14, Jan. 2011.

[7] O. Ozdemir, R. Niu, P. K. Varshney, and A. L. Drozd, "Modified Bayesian Cramér-Rao lower bound for nonlinear tracking," in *Proc. Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Prague, Czech Republic, May 2011.

[8] A. Doucet, N. de Freitas, and N. Gordon, *Sequential Monte Carlo Methods in Practice*. New York: Springer, 2001.

[9] R. Horn and C. R. Johnson, *Matrix Analysis*. New York: Cambridge Univ. Press, 1985.

[10] A. Householder, *The Theory of Matrices in Numerical Analysis*. New York: Blaisdell, 1964.

[11] Y. Boers, H. Driessen, A. Bagchi, and P. Mandal, "Particle filter based entropy," in *Proc. Int. Conf. Inf. Fusion*, Edinburgh, Scotland, Jul. 2010.

[12] S. C. Chapra and R. P. Canale, *Numerical Methods for Engineers*. New York: McGraw-Hill, 2002.

[13] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 174–188, Feb. 2002.

[14] J. H. Kotecha and P. M. Djurić, "Gaussian particle filtering," *IEEE Trans. Signal Process.*, vol. 51, no. 10, pp. 2592–2601, Oct. 2003.

[15] A. G. Gray and A. W. Moore, "'N-body' problems in statistical learning," in *Proc. Ann. Conf. Neural Inf. Process. Syst. (NIPS)*, 2000.

[16] M. Klaas, M. Briers, N. Freitas, A. Doucet, S. Maskell, and D. Lang, "Fast particle smoothing: If I had a million particles," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2006.

Robust Fault Isolation With Statistical Uncertainty in Identified Parameters

Jianfei Dong, Michel Verhaegen, and Fredrik Gustafsson

***Abstract*—This correspondence is a companion paper to [J. Dong, M. Verhaegen, and F. Gustafsson, "Robust Fault Detection With Statistical Uncertainty in Identified Parameters," *IEEE Trans. Signal Process.*, vol. 60, no. 10, Oct. 2012], extending it to fault isolation. Also, here, use is made of a linear in the parameters model representation of the input-output behavior of the nominal system (i.e. fault-free). The projection of the residual onto directions only sensitive to individual faults is robustified against the stochastic errors of the estimated model parameters. The correspondence considers additive error sequences to the input and output quantities that represent failures like drift, biased, stuck, or saturated sensors/actuators.**

***Index Terms*—Additive faults, closed-form solution, fault isolation, parameter uncertainty, statistical analysis.**

I. Introduction

In classical FDI literature, fault isolation is usually enabled by projecting a residual vector onto the left null space of all but one fault input directions in the matrix that maps faults to outputs (referred to as *fault transfer matrix* in what follows), e.g. [2], [3]. But if these projection vectors are identified from data as in [3], it is difficult to quantify the statistical distribution of this solution against identification uncertainty. In this paper, we develop a new optimization-based solution, which searches for the projection directions in the subspace spanned by the non-principal components of the error covariance matrix of the identified fault transfer matrix. In other words, the residual vectors are projected onto the least variant subspace of the error covariance matrix, where the components of the identified parameters are most likely to be close to the their true values.

The rest of the paper is organized as follows. We start in Section II with the preliminaries and problem formulation. Section III goes further to derive a closed-from optimal isolation solution against the parameter identification errors. Section IV shows the improvements in fault isolation performance by our robustified method on aircraft dynamics. The notations in this paper are the same as those defined in [1, Sec. II.A].

II. Preliminaries and Problem Formulation

A. Fault Isolation Connected to the VARX Description

We consider the following discrete-time state-space model with additive faults

$$x(k+1) = Ax(k) + B\left[u(k) + f_a(k)\right] + Fw(k), \tag{1}$$

$$y(k) = Cx(k) + f_s(k) + v(k). \tag{2}$$

Here, $x(k) \in \mathbb{R}^n$, $y(k) \in \mathbb{R}^\ell$, $u(k) \in \mathbb{R}^m$, and $f_a(k) \in \mathbb{R}^m$ and $f_s(k) \in \mathbb{R}^\ell$ respectively stand for additive actuator and sensor faults. For brevity, we will collect all the faults into $f(k) \triangleq \left[f_a^T(k), f_s^T(k)\right]^T \in \mathbb{R}^{m+\ell}$, and denote $n_f \triangleq m + \ell$.

Compared with the more general model in the companion paper, i.e. [1, Eqs. (1,2)], the fault model here correspond to the case where $E = [B, 0]$ and $G = [0, I]$; i.e. actuator faults share the same input channels with the control signals, and sensor faults directly add to the output measurements. This model can describe many commonly encountered additive faults, e.g. drifted, biased, stuck, or saturated actuators and sensors. The advantage of this model is that the Markov parameters from $f(k)$ to $y(k)$ are equal to those from $u(k)$ and $y(k)$ to $y(k)$, and can hence be estimated from I/O data.

Under the existence conditions of the stabilizing Kalman gain K, as specified in [1, Assumption 1], a closed-loop observer form of (1), (2) is (with $\Phi \triangleq A - KC$)

$$\begin{aligned} \hat{x}(k+1) &= \Phi\hat{x}(k) + Bu(k) + [B \quad -K]f(k) + Ky(k), \\ y(k) &= C\hat{x}(k) + [0 \quad I]f(k) + e(k). \end{aligned}$$

Here, $e(k)$ is the innovation signal defined in [1, Sec. II.D], and has a covariance matrix Σ_e.

As detailed in [1], a residual generator for fault detection along the horizon $[k - L + 1, k]$ takes the following form:

$$\begin{aligned} \boldsymbol{r}_{k,L} &= \left(I - \boldsymbol{T}_y^L\right)\boldsymbol{y}_{k,L} - \boldsymbol{H}_z^{L,p}\boldsymbol{z}_{k-L,p} - \boldsymbol{T}_u^L\boldsymbol{u}_{k,L} && (3) \\ &= \boldsymbol{\varphi}_f + \boldsymbol{b}_{k,L} + \boldsymbol{e}_{k,L}. && (4) \end{aligned}$$

To avoid repetition, we shall refer to [1] for the definition of the signal vectors $\boldsymbol{r}_{k,L}, \boldsymbol{u}_{k,L}, \boldsymbol{y}_{k,L}, \boldsymbol{b}_{k,L}, \boldsymbol{e}_{k,L}, \boldsymbol{\varphi}_f$ and the parametric matrices $\boldsymbol{H}_z^{L,p}, \boldsymbol{T}_u^L, \boldsymbol{T}_y^L$. We also denote these matrices with identified parameters by a bar on their top, e.g. $\bar{\boldsymbol{H}}_z^{L,p}$.

It is useful to recall that $\boldsymbol{\varphi}_f = [\boldsymbol{H}_f^{L,p}\ \boldsymbol{T}_f^L] \cdot \boldsymbol{f}_{k,p+L}$, with $\boldsymbol{f}_{k,p+L} = \left[f^T(k-L-p+1), \cdots, f^T(k)\right]^T$. Due to its role in mapping the fault signals to the outputs, we shall call $[\boldsymbol{H}_f^{L,p}\ \boldsymbol{T}_f^L]$ fault transfer matrix, which will be explicitly specified later.

The fault detection method in [1] aims at detecting the change in the mean of $\boldsymbol{r}_{k,L}$ due to a nonzero $\boldsymbol{\varphi}_f$. But to tell what components of

Manuscript received April 23, 2012; revised June 26, 2012; accepted June 29, 2012. Date of publication July 13, 2012; date of current version September 11, 2012. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Petr Tichavsky.

J. Dong was with the Delft University of Technology. He is now with Philips Research, 5656 AE, Eindhoven, The Netherlands (e-mail: jfeidong@hotmail.com).

M. Verhaegen is with the Delft University of Technology, 2628CD, Delft, The Netherlands (e-mail: m.verhaegen@tudelft.nl).

F. Gustafsson is with the Department of Electrical Engineering, Linköpings Universitet, SE-581 83 Linköping, Sweden (e-mail: fredrik@isy.liu.se).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSP.2012.2208639

1053-587X/$31.00 © 2012 IEEE