SIAM J. OPTIM.
Vol. 12, No. 3, pp. 740–755

ROBUST FILTERING VIA SEMIDEFINITE PROGRAMMING WITH APPLICATIONS TO TARGET TRACKING*

LINGJIE LI†, ZHI-QUAN LUO†, TIMOTHY N. DAVIDSON†, K. MAX WONG†, AND ELOI BOSSÉ‡

Abstract. In this paper we propose a novel finite-horizon, discrete-time, time-varying filtering method based on the robust semidefinite programming (SDP) technique. The proposed method provides robust performance in the presence of norm-bounded parameter uncertainties in the system model. The robust performance of the proposed method is achieved by minimizing an upper bound on the worst-case variance of the estimation error for all admissible systems. Our method is recursive and computationally efficient. In our simulations, the new method provides superior performance to some of the existing robust filtering approaches. In particular, when applied to the problem of target tracking, the new method has led to a significant improvement in tracking performance. Our work shows that the robust SDP technique and the interior point algorithms can bring substantial benefits to practically important engineering problems.

Key words. robust filtering, Kalman filtering, semidefinite programming, target tracking

AMS subject classifications. 90C90, 90C22, 90C51

PII. S1052623499358586

1. Introduction.

Consider the following classical discrete-time linear state-space model:

$$\left\{ \begin{array}{lcll} \underline{x}_{i+1} & = & \mathbf{F}_i \underline{x}_i + \mathbf{G}_i \underline{u}_i, & \underline{x}_0 \text{ given}, \\ \underline{y}_i & = & \mathbf{H}_i \underline{x}_i + \underline{v}_i, & i \geq 0, \end{array} \right. \tag{1.1}$$

where $\mathbf{F}_i \in \mathcal{R}^{n\times n}$, $\mathbf{G}_i \in \mathcal{R}^{n\times m}$, and $\mathbf{H}_i \in \mathcal{R}^{p\times n}$ are known matrices which describe the dynamic system, and $\underline{x}_i$ describes the state of the system at time i, while $\underline{u}_i$ and $\underline{v}_i$ denote the process and measurement noise terms, respectively. In many linear filtering applications, we are faced with the problem of estimating the states of the dynamic system (1.1) from the noisy measurements $\underline{y}_i$ (see [6, 11, 8]). A popular solution to this problem is given by the Kalman filter [6, 11, 8] which, under some standard assumptions on the statistics of the noise sources and initial state, minimizes the mean squared estimation error (MSE). The MSE is the trace of $\mathcal{E}\{(\underline{x}_i - \hat{\underline{x}}_i)(\underline{x}_i - \hat{\underline{x}}_i)^T\}$, where $\mathcal{E}$ denotes the statistical expectation and $\hat{\underline{x}}_i$ denotes the estimate of $\underline{x}_i$ at time i. Moreover, the Kalman filter is recursive and computationally efficient. In its "innovations form," the Kalman filter is given by

$$\hat{\underline{x}}_{i+1} = \mathbf{F}_i \hat{\underline{x}}_i + \mathbf{K}_{K,i}\big(\underline{y}_i - \mathbf{H}_i \hat{\underline{x}}_i\big), \quad \hat{\underline{x}}_0 = 0, \tag{1.2}$$

*Received by the editors June 23, 1999; accepted for publication (in revised form) June 13, 2001; published electronically February 8, 2002. This research was supported by a grant from the Defense Research Establishment of Canada at Valcartier, QC, Canada.
http://www.siam.org/journals/siopt/12-3/35858.html

†Department of Electrical and Computer Engineering, McMaster University, Hamilton, ON, L8S 4L7, Canada (lli@redlinecommunications.com, luozq@mcmaster.ca, davidson@sunrise.crl.mcmaster.ca, wong@mail.ece.mcmaster.ca). The second author is also supported by the Canada Research Chair program.

‡Defense Research Establishment Valcartier, Decision Support Technologies Section, 2459 PIE XI Nord, B.P. 8800, Courcelette, QC, G0A 1R0, Canada (eloi.bosse@drev.dnd.ca).

where the so-called Kalman gain matrix $\mathbf{K}_{K,i}$ can be computed via the following (analytic) recursion:

$$\mathbf{K}_{K,i} = \mathbf{F}_i\mathbf{P}_i\mathbf{H}_i^T\left(\mathbf{R}_i + \mathbf{H}_i\mathbf{P}_i\mathbf{H}_i^T\right)^{-1},$$
$$\mathbf{P}_{i+1} = \left(\mathbf{F}_i - \mathbf{K}_{K,i}\mathbf{H}_i\right)\mathbf{P}_i\left(\mathbf{F}_i - \mathbf{K}_{K,i}\mathbf{H}_i\right)^T + \left[\,\mathbf{G}_i \quad -\mathbf{K}_{K,i}\,\right]\begin{bmatrix}\mathbf{Q}_i & \mathbf{0}\\ \mathbf{0} & \mathbf{R}_i\end{bmatrix}\begin{bmatrix}\mathbf{G}_i^T\\ -\mathbf{K}_{K,i}^T\end{bmatrix},$$

where $\mathbf{Q}_i = \mathcal{E}\{\underline{u}_i\underline{u}_i^T\}$ and $\mathbf{R}_i = \mathcal{E}\{\underline{v}_i\underline{v}_i^T\}$ are the noise covariance matrices. (The statistical assumptions made here are stated in section 2.) The matrix $\mathbf{P}_i$ in the recursion is the error covariance matrix $\mathcal{E}\{(\underline{x}_i-\hat{\underline{x}}_i)(\underline{x}_i-\hat{\underline{x}}_i)^T\}$. However, one drawback of the Kalman filter is that it requires precise knowledge of the system matrices $\mathbf{F}_i$, $\mathbf{G}_i$, and $\mathbf{H}_i$ and noise covariances $\mathbf{Q}_i$ and $\mathbf{R}_i$, because even a small deviation from the "nominal" values of these matrices can induce substantial performance loss in the Kalman filter. As a result, the Kalman filter can be ineffective in practice especially when we are faced with imprecise knowledge of the dynamic system mode or, in other words, when the matrices $\mathbf{F}_i$, $\mathbf{G}_i$, and $\mathbf{H}_i$ are known only approximately. This sensitivity of the Kalman filter has led researchers to tackle *robust filtering* problems, in which the objective is to design estimators which provide acceptable performance in the presence of uncertainties in the models of the dynamic system and the noise.

One approach to robust filtering is that of H^∞ filtering (see [5] and references therein). In that approach no statistical model of the disturbances $\underline{u}_i$ and $\underline{v}_i$ is employed; they are merely assumed to have finite energy. The idea is to obtain an estimator which minimizes (or, in the suboptimal case, bounds) the maximum energy gain from the disturbances to the estimation errors. This modelling paradigm also allows us to incorporate unstructured uncertainties in the dynamic system model (1.1) (see, for example, [4, 17]). An advantage of the H^∞ approach is that the solution closely resembles the Kalman filter and can be efficiently implemented. Therefore, in applications in which statistical knowledge of the disturbances and information regarding the structure of the modelling uncertainties are difficult to acquire, H^∞ filters are an appropriate choice. Unfortunately, when the system model and the noise processes are known quite accurately, the Kalman filter may actually perform substantially better than the H^∞ filter. This is because the uncertainty model for the H^∞ filter is unstructured, and hence the H^∞ filter may be attempting to provide robustness to disturbances and modelling errors which rarely, or never, occur, at the expense of filter performance in the presence of more likely disturbances and modelling errors. In many applications, including target tracking, we have some knowledge of the structure of the uncertainties in the system model and partial knowledge of disturbance statistics. It is natural to expect that careful incorporation of this knowledge into the estimator will lead to appreciable improvement in estimator performance. A major challenge is to determine whether this can be done in a computationally efficient manner. From recent work in the control field, it appears that determining filters which provide optimal robustness to highly structured uncertainties can be computationally expensive [1].

An alternative to the Kalman and H^∞ filtering methods is to find a "robust Kalman filter" which minimizes (an upper bound on) the variance of the estimation error in the presence of a system model with norm-bounded structured parametric uncertainty and bounded uncertainty in the noise statistics. Models of this type are common in control theory (e.g., [7] and references therein) and are particularly appropriate in the context of target tracking. Previous approaches to this problem, with no uncertainty in the noise statistics, have been based on analytic recursions on some performance bounds [13, 15]. Note that robust H^∞ designs which bound the worst-case error energy gain in the presence of the same system model uncertainties are also available [9, 16].

In this paper we derive a new robust filtering algorithm using the recently developed robust semidefinite programming (SDP) technique [2]. The new method is recursive in the sense that the subproblem solved at each step depends on the solution at the previous step, and is computationally efficient since each subproblem is a semidefinite program of a fixed size which can be efficiently solved using an interior point algorithm. We demonstrate the performance of the novel algorithm in a standard benchmark example and in a target-tracking example, and show that it can provide superior performance to the existing approaches to this particular problem [13, 15], and to the Kalman and H^∞ approaches. Our work shows that the robust SDP technique and the interior point algorithms [12, 14] can bring substantial benefits to a practically important engineering problem.

The paper is organized as follows. In section 2 the robust state estimator problem is introduced. Then, in section 3, this problem is formulated as convex optimization and solved in polynomial-time using the recent robust SDP technique. In section 4, simulation results are presented and, in section 5, some concluding remarks are given.

Throughout this paper, for a square matrix $\mathbf{X}$, the notation $\mathbf{X} \geq 0$ (resp., $\mathbf{X} \leq 0$) means $\mathbf{X}$ is symmetric and positive semidefinite (resp., negative semidefinite).

2. Problem formulation. Consider the following time-varying, discrete-time, uncertain linear state-space model:

$$\left\{ \begin{array}{rcll} \underline{x}_{i+1} & = & [\mathbf{F}_i + \Delta\mathbf{F}_i]\,\underline{x}_i + \mathbf{G}_i\underline{u}_i, & \underline{x}_0, \\ \underline{y}_i & = & [\mathbf{H}_i + \Delta\mathbf{H}_i]\,\underline{x}_i + \underline{v}_i, & i \geq 0, \end{array} \right. \tag{2.1}$$

where $\mathbf{F}_i \in \mathcal{R}^{n\times n}$, $\mathbf{G}_i \in \mathcal{R}^{n\times m}$, and $\mathbf{H}_i \in \mathcal{R}^{p\times n}$ are known matrices which describe the nominal system. The matrices $\Delta\mathbf{F}_i$ and $\Delta\mathbf{H}_i$ represent the parameter uncertainties in the dynamic model. They are assumed to have the following structure:

$$\begin{bmatrix} \Delta\mathbf{F}_i \\ \Delta\mathbf{H}_i \end{bmatrix} = \begin{bmatrix} \mathbf{C}_{1,i} \\ \mathbf{C}_{2,i} \end{bmatrix} \mathbf{Z}_i\mathbf{E}_i \quad \text{with} \quad \mathbf{Z}_i^T\mathbf{Z}_i \leq \mathbf{I}, \tag{2.2}$$

where $\mathbf{C}_{1,i} \in \mathcal{R}^{n\times r}$, $\mathbf{C}_{2,i} \in \mathcal{R}^{p\times r}$, and $\mathbf{E}_i \in \mathcal{R}^{t\times n}$ are known matrices. We remark that the above model (2.2) of uncertainties has been used extensively in the robust control literature (e.g., [7] and references therein). The process noise $\{\underline{u}_i\}$, the measurement noise $\{\underline{v}_i\}$, and the initial state $\underline{x}_0$ in (2.1) are all assumed to be random. These random variables have known mean values, which we can take to be zero without loss of generality, and partially known covariances, as follows:

$$\mathcal{E}\left\{ \begin{bmatrix} \underline{u}_i \\ \underline{v}_i \\ \underline{x}_0 \end{bmatrix} \begin{bmatrix} \underline{u}_j \\ \underline{v}_j \\ \underline{x}_0 \end{bmatrix}^T \right\} = \begin{bmatrix} \mathbf{Q}_i\delta_{ij} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_i\delta_{ij} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{\Pi_0} \end{bmatrix}, \tag{2.3}$$

where δ_{ij} denotes the Kronecker delta function that is equal to unity for $i = j$ and zero elsewhere, $\mathbf{Q}_i = \bar{\mathbf{Q}}_i + \Delta\mathbf{Q}_i$, and $\mathbf{R}_i = \bar{\mathbf{R}}_i + \Delta\mathbf{R}_i$. The matrices $\bar{\mathbf{Q}}_i \in \mathcal{R}^{m\times m}$, $\bar{\mathbf{R}}_i \in \mathcal{R}^{p\times p}$, and $\mathbf{\Pi}_0 \in \mathcal{R}^{n\times n}$ are assumed to be known and describe the nominal second-order statistics of the noise and the initial state. The matrices $\Delta\mathbf{Q}_i$ and $\Delta\mathbf{R}_i$ represent the uncertainties in the noise statistics and satisfy the following bounds:

$$-\epsilon\mathbf{I} \leq \Delta\mathbf{Q}_i \leq \epsilon\mathbf{I}, \quad -\epsilon\mathbf{I} \leq \Delta\mathbf{R}_i \leq \epsilon\mathbf{I}. \tag{2.4}$$

Notice that when there is no uncertainty in the system model (2.1), namely $\epsilon = 0$ and $\mathbf{E}_i = 0$, then we recover the standard linear time-varying state-space model (1.1).

Let us use $\Theta_i = \{\Delta\mathbf{Q}_i, \Delta\mathbf{R}_i, \mathbf{Z}_i\}$ to denote the uncertainty variable at stage i and define the uncertainty region at stage i as

$$\Omega_i = \{\ \Theta_i : \Theta_i \text{ satisfies (2.2) and (2.4) }\}.$$

The problem is to estimate the state-sequence $\{\underline{x}_i,\ i \geq 0\}$, or some linear combination of this sequence $\{\underline{s}_i = \mathbf{L}_i\underline{x}_i, i \geq 0\}$, where $\mathbf{L}_i$ is a known matrix, from the corrupted measurements. The goal of the robust filter is to provide a uniformly small estimation error for any process and measurement noise satisfying (2.3) and (2.4) and for all admissible modelling uncertainties satisfying (2.2). These a priori bounds on the uncertainties represent the designer's partial knowledge of the noise statistics and system model. They are to be incorporated into the problem formulation to guarantee robust performance.

To formulate the robust filtering problem, consider the following form of state estimator:

$$\hat{\underline{x}}_{i+1} = \mathbf{A}_i\hat{\underline{x}}_i + \mathbf{K}_i(\underline{y}_i - \mathbf{H}_i\hat{\underline{x}}_i), \quad \hat{\underline{x}}_0 = 0, \tag{2.5}$$

where $\mathbf{A}_i$, $\mathbf{K}_i$ are filtering matrices to be determined, and $\hat{\underline{x}}_i$ denotes the estimate of the state $\underline{x}_i$. The above estimator is written in an innovation form that is similar to the structure of the Kalman filter given in (1.2). Notice that we use the nominal innovation $(\underline{y}_i - \mathbf{H}_i\hat{\underline{x}}_i)$, even though $\Delta\mathbf{H}_i$ may be nonzero. This structure is used for convenience, but it is general enough to generate all the full-order estimators, since $\mathbf{A}_i$ and $\mathbf{K}_i$ are free parameters. The goal of a robust filtering algorithm is to choose these free parameters to minimize (a function of) the estimation error covariance $\mathcal{E}\{(\underline{x}_i - \hat{\underline{x}}_i)(\underline{x}_i - \hat{\underline{x}}_i)^T\}$.

To express that goal precisely, we consider the following augmented system, which represents the cascade of the system in (2.1) and the estimator in (2.5):

$$\bar{\underline{x}}_{i+1} = \left[\bar{\mathbf{F}}_i + \bar{\mathbf{C}}_i\mathbf{Z}_i\bar{\mathbf{E}}_i\right]\bar{\underline{x}}_i + \bar{\mathbf{G}}_i\bar{\underline{u}}_i, \tag{2.6}$$

where

$$\begin{cases} \bar{\underline{x}}_i &= \begin{bmatrix} \underline{x}_i \\ \hat{\underline{x}}_i \end{bmatrix}, \quad \bar{\underline{u}}_i = \begin{bmatrix} \underline{u}_i \\ \underline{v}_i \end{bmatrix}, \\ \bar{\mathbf{F}}_i &= \begin{bmatrix} \mathbf{F}_i & \mathbf{0} \\ \mathbf{K}_i\mathbf{H}_i & \mathbf{A}_i - \mathbf{K}_i\mathbf{H}_i \end{bmatrix}, \quad \bar{\mathbf{G}}_i = \begin{bmatrix} \mathbf{G}_i & \mathbf{0} \\ \mathbf{0} & \mathbf{K}_i \end{bmatrix}, \\ \bar{\mathbf{C}}_i &= \begin{bmatrix} \mathbf{C}_{1,i} \\ \mathbf{K}_i\mathbf{C}_{2,i} \end{bmatrix}, \quad \bar{\mathbf{E}}_i = [\mathbf{E}_i \ \ \mathbf{0}]. \end{cases}$$

Note that the state vector of the cascade, $\bar{\underline{x}}_i$, contains both $\underline{x}_i$ (the states of the model) and the estimates $\hat{\underline{x}}_i$, and hence the dimension of the state vector is doubled. The Lyapunov equation that governs the evolution of the covariance matrix $\mathbf{\Sigma}_i = \mathcal{E}\{\bar{\underline{x}}_i\bar{\underline{x}}_i^T\}$ can be written as

$$\mathbf{\Sigma}_{i+1} = \left[\bar{\mathbf{F}}_i + \bar{\mathbf{C}}_i\mathbf{Z}_i\bar{\mathbf{E}}_i\right]\mathbf{\Sigma}_i\left[\bar{\mathbf{F}}_i + \bar{\mathbf{C}}_i\mathbf{Z}_i\bar{\mathbf{E}}_i\right]^T + \bar{\mathbf{G}}_i\mathbf{W}_i\bar{\mathbf{G}}_i^T, \tag{2.7}$$

where $\mathbf{W}_i = \text{blockdiag}(\mathbf{Q}_i, \mathbf{R}_i)$. The error covariance $\mathbf{P}_{i+1}$ can be obtained from (2.7) by premultiplying $[\mathbf{I} \quad -\mathbf{I}]$ and postmultiplying $[\mathbf{I} \quad -\mathbf{I}]^T$; i.e.,

$$\mathbf{P}_{i+1} = \hat{\mathbf{F}}_i \mathbf{\Sigma}_i \hat{\mathbf{F}}_i^T + \mathbf{G}_i \mathbf{Q}_i \mathbf{G}_i^T + \mathbf{K}_i \mathbf{R}_i \mathbf{K}_i^T, \tag{2.8}$$

where

$$\hat{\mathbf{F}}_i = \left[\begin{array}{cc} (\mathbf{F}_i + \mathbf{C}_{1,i}\mathbf{Z}_i\mathbf{E}_i - \mathbf{K}_i\mathbf{H}_i - \mathbf{K}_i\mathbf{C}_{2,i}\mathbf{Z}_i\mathbf{E}_i) & (\mathbf{K}_i\mathbf{H}_i - \mathbf{A}_i) \end{array} \right].$$

Now the finite-horizon robust state estimator problem can be stated as follows.

PROBLEM. *At each stage i, choose the filtering matrices $\{\mathbf{A}_j\}_{j=0}^i$ and $\{\mathbf{K}_j\}_{j=0}^i$ so as to minimize the worst-case weighted error covariance matrix $\mathbf{DP}_{i+1}$; i.e.,*

$$\min_{\substack{\mathbf{K}_j, \mathbf{A}_j \\ \forall j \le i}} \quad \max_{\substack{\Theta_j \in \Omega_j \\ \forall j \le i}} \text{Tr}\,(\mathbf{DP}_{i+1}), \tag{2.9}$$

or equivalently,

$$\min_{\substack{\mathbf{K}_j, \mathbf{A}_j \\ \forall j \le i}} \quad \max_{\substack{\Theta_j \in \Omega_j \\ \forall j \le i}} \text{Tr}\left(\mathbf{D}\,[\mathbf{I} \quad -\mathbf{I}]\,\mathbf{\Sigma}_{i+1}\,[\mathbf{I} \quad -\mathbf{I}]^T\right), \tag{2.10}$$

where $\text{Tr}\,(\cdot)$ *denotes the trace of a matrix* $(\cdot)$ *and* $\mathbf{D} \in \mathcal{R}^{n \times n}$ *is a positive semidefinite weighting matrix.*

We have stated the robust state estimation problem in a rather general weighted form which includes many special cases. If we wish to estimate $\{\underline{x}_i\}$, choosing $\mathbf{D} = \mathbf{I}$ will suffice, whereas to estimate $\{\underline{s}_i = \mathbf{L}_i \underline{x}_i\}$, choosing $\mathbf{D} = \mathbf{L}_i \mathbf{L}_i^T$ will suffice. We can also weight the estimation accuracy of the states as desired, or add additional terms to $\mathbf{D}$, as long as it remains positive semidefinite. As we will observe later in section 4, adding additional terms to $\mathbf{D}$ may improve the numerical stability of the finite-horizon filtering solutions.

The above minimax formulation is intended to incorporate robustness into the filter solution. In particular, $\text{Tr}\,(\mathbf{DP}_{i+1})$, as recursively defined by (2.8), depends on all the uncertainties $\Theta_0, \ldots, \Theta_i$ as well as on the filtering matrices $\mathbf{K}_0, \mathbf{A}_0, \ldots, \mathbf{K}_i, \mathbf{A}_i$. The maximum weighted trace of $\mathbf{P}_{i+1}$,

$$\max_{\substack{\Theta_j \in \Omega_j \\ \forall j \le i}} \text{Tr}\,(\mathbf{DP}_{i+1}),$$

represents the worst-case weighted error covariance when subject to the prescribed uncertainties. Therefore, the goal of robust filter design is to select the filtering matrices so that the worst-case weighted error covariance is minimized.

As given by (2.9) or (2.10), the robust filter design problem is nonlinear and nonsmooth and hence is computationally difficult. Furthermore, the problem apparently lacks convexity, which is essential in the development of computationally efficient algorithms. A further difficulty with the formulation (2.9) or (2.10) is that it is nonrecursive, in the sense that the problem dimension increases linearly in i. This nonrecursive feature makes it necessary to solve from scratch for the filtering matrices $\mathbf{K}_0, \mathbf{A}_0, \ldots, \mathbf{K}_i, \mathbf{A}_i$ at each stage i, which is clearly undesirable and impractical.

In practice, we typically fix $\mathbf{K}_0, \mathbf{A}_0, \ldots, \mathbf{K}_{i-1}, \mathbf{A}_{i-1}$ at stage i and solve only for $\mathbf{K}_i, \mathbf{A}_i$. However, such simplification only partially fixes the problem since the uncertainties $\Theta_0, \ldots, \Theta_i$ still enter into the maximization of $\text{Tr}\,(\mathbf{DP}_{i+1})$, indicating that the problem dimension still increases linearly with i. Our objective is to reformulate problem (2.9) in a recursive way such that at each stage i we have only to determine $\mathbf{K}_i, \mathbf{A}_i$ by solving a subproblem with a fixed dimension (i.e., independent of i).

To reformulate problem (2.9), we consider a sequence of matrices

$$\{\mathbf{\Gamma}_{i+1}(\mathbf{K}_i, \mathbf{A}_i) : \ i = 1, 2, \ldots\},$$

which are *not* dependent on the uncertainties $\{\Theta_i : \ i = 1, 2, \ldots\}$. These matrices will serve as upper bounds for the covariance matrices $\{\mathbf{\Sigma}_{i+1} : \ i = 1, 2, \ldots\}$ which *are* dependent on the uncertainty vectors $\{\Theta_i : \ i = 1, 2, \ldots\}$, as well as on $\mathbf{K}_i$ and $\mathbf{A}_i$. In particular, we will have

$$\mathbf{\Gamma}_{i+1}(\mathbf{K}_i, \mathbf{A}_i) \geq \mathbf{\Sigma}_{i+1} \qquad \forall \Theta_i \in \Omega_i, \quad i = 1, 2, \ldots. \tag{2.11}$$

There are, of course, many choices for an upper bound $\mathbf{\Gamma}_{i+1}(\mathbf{K}_i, \mathbf{A}_i)$ that will satisfy (2.11). Our objective should be to choose the one which, together with some $\mathbf{K}_i$ and $\mathbf{A}_i$, will yield the minimum weighted error covariance $\mathbf{DP}_{i+1}$. By the relation

$$\mathbf{P}_{i+1} = \begin{bmatrix} \mathbf{I} & -\mathbf{I} \end{bmatrix} \mathbf{\Sigma}_{i+1} \begin{bmatrix} \mathbf{I} & -\mathbf{I} \end{bmatrix}^T,$$

we see that an upper bound on $\mathbf{\Sigma}_{i+1}$ naturally leads to an upper bound on $\mathbf{P}_{i+1}$. Thus we can approximately minimize $\mathbf{DP}_{i+1}$ by minimizing the trace of the matrix

$$\mathbf{D} \begin{bmatrix} \mathbf{I} & -\mathbf{I} \end{bmatrix} \mathbf{\Gamma}_{i+1}(\mathbf{K}_i, \mathbf{A}_i) \begin{bmatrix} \mathbf{I} & -\mathbf{I} \end{bmatrix}^T,$$

which is an upper bound of $\mathbf{DP}_{i+1}$. In particular, we choose $\mathbf{\Gamma}_{i+1}$, $\mathbf{K}_i$, and $\mathbf{A}_i$ to

$$\begin{array}{ll} \text{minimize} & \mathrm{Tr}\left(\mathbf{D} \begin{bmatrix} \mathbf{I} & -\mathbf{I} \end{bmatrix} \mathbf{\Gamma}_{i+1} \begin{bmatrix} \mathbf{I} & -\mathbf{I} \end{bmatrix}^T\right) \\ \text{subject to} & \mathbf{\Gamma}_{i+1},\ \mathbf{K}_i,\ \mathbf{A}_i \text{ satisfying (2.11).} \end{array} \tag{2.12}$$

The optimization problem (2.12) involves the constraint (2.11), which involves all of the uncertainty vectors $\{\Theta_i : \ i = 1, 2, \ldots\}$ and $\{\mathbf{K}_i, \mathbf{A}_i : \ i = 1, 2, \ldots\}$, thus making the amount of computation increase with i. To resolve this issue of dimensionality increase, we shall define the constraint recursively as follows. Specifically, let $b > 0$ be a chosen scalar bound and let $\bar{\mathbf{\Sigma}}_0 = \mathbf{\Sigma}_0$. For $i \geq 0$, suppose $\bar{\mathbf{\Sigma}}_i$, an upper bound on $\mathbf{\Sigma}_i$, has been computed and is already available. Consider the following minimization problem in the matrix variables $\{\mathbf{\Gamma}_{i+1}, \mathbf{K}_i, \mathbf{A}_i\}$:

$$\begin{array}{lll} \text{minimize} & \mathrm{Tr}\left(\mathbf{D} \begin{bmatrix} \mathbf{I} & -\mathbf{I} \end{bmatrix} \mathbf{\Gamma}_{i+1} \begin{bmatrix} \mathbf{I} & -\mathbf{I} \end{bmatrix}^T\right) & \\ \text{subject to} & \mathbf{\Gamma}_{i+1} \geq \left[\bar{\mathbf{F}}_i + \bar{\mathbf{C}}_i \mathbf{Z}_i \bar{\mathbf{E}}_i\right] \bar{\mathbf{\Sigma}}_i \left[\bar{\mathbf{F}}_i + \bar{\mathbf{C}}_i \mathbf{Z}_i \bar{\mathbf{E}}_i\right]^T + \bar{\mathbf{G}}_i \mathbf{W}_i \bar{\mathbf{G}}_i^T & \forall \Theta_i \in \Omega_i, \\ & \mathrm{Tr}\,(\mathbf{\Gamma}_{i+1}) \leq b. & \end{array} \tag{2.13}$$

We choose $\bar{\mathbf{\Sigma}}_{i+1}$ to be the optimal value of $\mathbf{\Gamma}_{i+1}$ in (2.13). Therefore our reformulation of (2.9) can now be stated as the following.

REFORMULATION OF THE ROBUST FILTERING PROBLEM. *Let* $\bar{\mathbf{\Sigma}}_0 = \mathbf{\Sigma}_0$. *For each* $i \geq 0$ *compute, recursively, the matrix* $\bar{\mathbf{\Sigma}}_{i+1}$ *and the robust filtering matrices* $\mathbf{A}_i$ *and* $\mathbf{K}_i$ *as the minimizing solution of* (2.13).

We remark that the second constraint in (2.13), $\mathrm{Tr}\,(\mathbf{\Gamma}_{i+1}) \leq b$, is used to ensure that the matrix $\mathbf{\Gamma}_{i+1}$ is bounded. This is important because otherwise the optimal solution of (2.13), $\bar{\mathbf{\Sigma}}_{i+1}$, may become progressively ill-conditioned as i becomes large.

An alternative way of preventing ill-conditioning is to impose the following structure on $\mathbf{\Gamma}_{i+1}$,

$$\mathbf{\Gamma}_{i+1} = \begin{bmatrix} \bar{\mathbf{\Gamma}} + \hat{\mathbf{\Gamma}} & \bar{\mathbf{\Gamma}} \\ \bar{\mathbf{\Gamma}} & \bar{\mathbf{\Gamma}} \end{bmatrix} \qquad \text{for some symmetric matrices } \bar{\mathbf{\Gamma}},\ \hat{\mathbf{\Gamma}}, \tag{2.14}$$

and to use the following constraint:

$$\operatorname{Tr}(\hat{\mathbf{\Gamma}}) \geq \beta \operatorname{Tr}(\bar{\mathbf{\Gamma}}), \tag{2.15}$$

where $\beta > 0$ is a constant. The above structure (2.14) for $\mathbf{\Gamma}_{i+1}$ mimics the structure of the joint covariance matrix of the state of a system and its optimal estimate in the Kalman sense, and is maintained in [13]. The bound (2.15) is used to ensure that the condition number of $\mathbf{\Gamma}_{i+1}$ does not become unbounded when $\bar{\mathbf{\Gamma}}$ and $\hat{\mathbf{\Gamma}}$ become large. Indeed, notice that

$$\mathbf{\Gamma}_{i+1} = \begin{bmatrix} \bar{\mathbf{\Gamma}} + \hat{\mathbf{\Gamma}} & \bar{\mathbf{\Gamma}} \\ \bar{\mathbf{\Gamma}} & \bar{\mathbf{\Gamma}} \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{I} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{\Gamma}} & \mathbf{0} \\ \mathbf{0} & \bar{\mathbf{\Gamma}} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{I} & \mathbf{I} \end{bmatrix},$$

so we only need to bound the condition number for the matrix blockdiag$\{\hat{\mathbf{\Gamma}}, \bar{\mathbf{\Gamma}}\}$. By the above factorization of $\mathbf{\Gamma}_{i+1}$ and the fact that the right-hand side of the first constraint in (2.13) is bounded from below by a positive definite matrix, we obtain that blockdiag$\{\hat{\mathbf{\Gamma}}, \bar{\mathbf{\Gamma}}\}$ is bounded from below by a positive definite matrix. Thus, the smallest eigenvalue of the matrix blockdiag$\{\hat{\mathbf{\Gamma}}, \bar{\mathbf{\Gamma}}\}$ is bounded away from zero. In the meantime, the constraint (2.15) and the fact that we are minimizing $\hat{\mathbf{\Gamma}}$ implies that the largest eigenvalue of the matrix blockdiag$\{\hat{\mathbf{\Gamma}}, \bar{\mathbf{\Gamma}}\}$ is also bounded. This implies the boundedness of the condition number of $\mathbf{\Gamma}_{i+1}$ at optimal solution.

As a result of the above discussion, we have the following alternative formulation (to (2.13)):

$$\begin{aligned} &\text{minimize} && \operatorname{Tr}\left(\mathbf{D} \begin{bmatrix} \mathbf{I} & -\mathbf{I} \end{bmatrix} \mathbf{\Gamma}_{i+1} \begin{bmatrix} \mathbf{I} & -\mathbf{I} \end{bmatrix}^T\right) \\ &\text{subject to} && \mathbf{\Gamma}_{i+1} \geq \left[\bar{\mathbf{F}}_i + \bar{\mathbf{C}}_i \mathbf{Z}_i \bar{\mathbf{E}}_i\right] \bar{\mathbf{\Sigma}}_i \left[\bar{\mathbf{F}}_i + \bar{\mathbf{C}}_i \mathbf{Z}_i \bar{\mathbf{E}}_i\right]^T + \bar{\mathbf{G}}_i \mathbf{W}_i \bar{\mathbf{G}}_i^T \qquad \forall \Theta_i \in \Omega_i, \\ &&& \mathbf{\Gamma}_{i+1} \text{ satisfying (2.14) and (2.15).} \end{aligned} \tag{2.16}$$

In the remainder of this paper, we will focus on the first formulation (2.12), but the second formulation (2.16) can also be treated in an analogous fashion.

We point out that the dimension of problem (2.13) is fixed rather than growing linearly with i. Moreover, it will be shown that (2.13) is convex and can be reformulated as a semidefinite program. The latter can be solved very efficiently via interior point methods [14, 10, 12]. Before we explain how to solve (2.13), we need to show that $\bar{\mathbf{\Sigma}}_i$ defined by (2.13) does provide an upper bound for $\mathbf{\Sigma}_i$ for all $i \geq 0$. We have the following theorem.

THEOREM 2.1. *Let $\bar{\mathbf{\Sigma}}_0 = \mathbf{\Sigma}_0$. For $i \geq 1$, let $\bar{\mathbf{\Sigma}}_i$ be defined as in* (2.13). *Then there holds*

$$\bar{\mathbf{\Sigma}}_i \geq \mathbf{\Sigma}_i \qquad \forall \Theta_j \in \Omega_j, \quad j = 1, 2, \ldots, i-1. \tag{2.17}$$

Proof. The theorem can be proved by mathematical induction. In particular, for $i = 0$ we have $\bar{\mathbf{\Sigma}}_0 = \mathbf{\Sigma}_0$. Suppose that (2.17) holds for $i = k$. Since $\bar{\mathbf{\Sigma}}_{k+1}$ is the optimal solution of (2.13), it follows from the constraint of (2.13) that

$$\bar{\mathbf{\Sigma}}_{k+1} \geq \left[\bar{\mathbf{F}}_k + \bar{\mathbf{C}}_k \mathbf{Z}_k \bar{\mathbf{E}}_k\right] \bar{\mathbf{\Sigma}}_k \left[\bar{\mathbf{F}}_k + \bar{\mathbf{C}}_k \mathbf{Z}_k \bar{\mathbf{E}}_k\right]^T + \bar{\mathbf{G}}_k \mathbf{W}_k \bar{\mathbf{G}}_k^T \qquad \forall \Theta_k \in \Omega_k. \tag{2.18}$$

By the inductive hypothesis we have

$$\bar{\mathbf{\Sigma}}_k \geq \mathbf{\Sigma}_k \qquad \forall \Theta_j \in \Omega_j, \quad j = 1, 2, \ldots, (k-1).$$

Combining this with (2.18), we obtain

$$\begin{aligned}\bar{\mathbf{\Sigma}}_{k+1} &\geq \left[\bar{\mathbf{F}}_k + \bar{\mathbf{C}}_k \mathbf{Z}_k \bar{\mathbf{E}}_k\right] \mathbf{\Sigma}_k \left[\bar{\mathbf{F}}_k + \bar{\mathbf{C}}_k \mathbf{Z}_k \bar{\mathbf{E}}_k\right]^T + \bar{\mathbf{G}}_k \mathbf{W}_k \bar{\mathbf{G}}_k^T \\ &= \mathbf{\Sigma}_{k+1} \qquad\qquad \forall \Theta_j \in \Omega_j, \quad j = 1, 2, \ldots, k,\end{aligned}$$

where the last step is due to (2.7) for the particular value of Θ_j which represents the actual error in the model. This completes the induction proof. □

In common with the existing approaches to the finite-horizon robust filtering problem, we do not have a sufficient condition for the convergence of the estimator $\bar{\mathbf{\Sigma}}_i$ as i tends to infinity. However, we now provide some necessary conditions. (These conditions are analogous to those in [13].)

THEOREM 2.2. *Suppose the system* (2.1)–(2.4) *is time-invariant in the sense that the data matrices* $\mathbf{H}_i$, $\mathbf{C}_{1,i}$, $\mathbf{C}_{2,i}$, $\mathbf{G}_i$, $\mathbf{E}_i$, $\bar{\mathbf{R}}_i$, *and* $\bar{\mathbf{Q}}_i$ *are fixed and independent of* i. *Then the solution* $\bar{\mathbf{\Sigma}}_i$ *converges to some* $\bar{\mathbf{\Sigma}}$ *only if the set of uncertain systems* (2.1)–(2.2) *is quadratically stable.*

Proof. Let $\underline{u}_i$ and $\underline{v}_i$ be zero. By constraint (2.13) and the fact that $\bar{\mathbf{\Sigma}}_i \to \bar{\mathbf{\Sigma}}$, we have

$$\bar{\mathbf{\Sigma}} \geq \left[\bar{\mathbf{F}} + \bar{\mathbf{C}} \mathbf{Z}_i \bar{\mathbf{E}}\right] \bar{\mathbf{\Sigma}} \left[\bar{\mathbf{F}} + \bar{\mathbf{C}} \mathbf{Z}_i \bar{\mathbf{E}}\right]^T \qquad \forall \mathbf{Z}_i \text{ with } \|\mathbf{Z}_i\| \leq 1.$$

This shows that the augmented linear system (2.6) is quadratically stable. This is because the above relation easily implies that the quadratic Lyapunov function $V(\bar{\underline{x}}, i) = -\bar{\underline{x}}_i^T \bar{\mathbf{\Sigma}} \bar{\underline{x}}_i \geq 0$ and that, for all admissible systems, $V(\bar{\underline{x}}, i+1) \leq V(\bar{\underline{x}}, i)$ if the process noise $\bar{\underline{u}}_i = 0$. By construction, $\underline{x}_i$ is a component of $\bar{\underline{x}}_i$; therefore the quadratic stability of (2.6) (in this time-invariant case) implies the quadratic stability of (2.1)–(2.2) for all admissible systems. □

3. Robust SDP solution. In this section, we shall develop an SDP [14] formulation for the robust state estimator problem (in particular, the problem (2.13)). This will then allow for efficient numerical solutions via recent interior point methods. We begin by noting that the finite-horizon robust state estimator problem (2.13) has a constraint of the form

$$\mathbf{\Gamma}_{i+1} \geq \left[\bar{\mathbf{F}}_i + \bar{\mathbf{C}}_i \mathbf{Z}_i \bar{\mathbf{E}}_i\right] \bar{\mathbf{\Sigma}}_i \left[\bar{\mathbf{F}}_i + \bar{\mathbf{C}}_i \mathbf{Z}_i \bar{\mathbf{E}}_i\right]^T + \bar{\mathbf{G}}_i \mathbf{W}_i \bar{\mathbf{G}}_i^T \;\; \forall \Theta_i = (\Delta \mathbf{Q}_i, \Delta \mathbf{R}_i, \mathbf{Z}_i) \in \Omega_i, \tag{3.1}$$

which contains an uncertainty vector $\Theta_i = (\Delta \mathbf{Q}_i, \Delta \mathbf{R}_i, \mathbf{Z}_i)$. Recall that $\mathbf{W}_i = \text{blockdiag}(\bar{\mathbf{Q}}_i + \Delta \mathbf{Q}_i, \bar{\mathbf{R}}_i + \Delta \mathbf{R}_i)$ and that by (2.4) we have

$$-\epsilon \mathbf{I} \leq \Delta \mathbf{Q}_i \leq \epsilon \mathbf{I}, \quad -\epsilon \mathbf{I} \leq \Delta \mathbf{R}_i \leq \epsilon \mathbf{I}.$$

Therefore, by choosing the upper bound for $\mathbf{W}_i$, the constraint (3.1) holds for all $\Theta_i = (\Delta \mathbf{Q}_i, \Delta \mathbf{R}_i, \mathbf{Z}_i) \in \Omega_i$ if and only if the following holds:

$$\mathbf{\Gamma}_{i+1} \geq \left[\bar{\mathbf{F}}_i + \bar{\mathbf{C}}_i \mathbf{Z}_i \bar{\mathbf{E}}_i\right] \bar{\mathbf{\Sigma}}_i \left[\bar{\mathbf{F}}_i + \bar{\mathbf{C}}_i \mathbf{Z}_i \bar{\mathbf{E}}_i\right]^T + \bar{\mathbf{G}}_i \bar{\mathbf{W}}_i \bar{\mathbf{G}}_i^T \qquad \forall \mathbf{Z}_i \text{ with } \|\mathbf{Z}_i\| \leq 1,$$

where

$$\bar{\mathbf{W}}_i = \text{blockdiag}(\bar{\mathbf{Q}}_i + \epsilon \mathbf{I}, \bar{\mathbf{R}}_i + \epsilon \mathbf{I}).$$

We rearrange the above inequality as follows:

$$\mathbf{\Gamma}_{i+1} - \left[\bar{\mathbf{F}}_i + \bar{\mathbf{C}}_i\mathbf{Z}_i\bar{\mathbf{E}}_i \;\; \bar{\mathbf{G}}_i\right] \begin{bmatrix} \bar{\mathbf{\Sigma}}_i & \mathbf{0} \\ \mathbf{0} & \bar{\mathbf{W}}_i \end{bmatrix} \left[\bar{\mathbf{F}}_i + \bar{\mathbf{C}}_i\mathbf{Z}_i\bar{\mathbf{E}}_i \;\; \bar{\mathbf{G}}_i\right]^T \geq \mathbf{0}$$
$$\forall \mathbf{Z}_i \text{ with } \|\mathbf{Z}_i\| \leq 1.$$

Using the Schur complement, the above constraint is equivalent to

$$\begin{bmatrix} \bar{\mathbf{\Sigma}}_i^{-1} & \mathbf{0} & (\bar{\mathbf{F}}_i + \bar{\mathbf{C}}_i\mathbf{Z}_i\bar{\mathbf{E}}_i)^T \\ \mathbf{0} & \bar{\mathbf{W}}_i^{-1} & \bar{\mathbf{G}}_i^T \\ (\bar{\mathbf{F}}_i + \bar{\mathbf{C}}_i\mathbf{Z}_i\bar{\mathbf{E}}_i) & \bar{\mathbf{G}}_i & \mathbf{\Gamma}_{i+1} \end{bmatrix} \geq \mathbf{0} \quad \forall \mathbf{Z}_i \text{ with } \|\mathbf{Z}_i\| \leq 1. \tag{3.2}$$

Note that both $\bar{\mathbf{\Sigma}}_i$ and $\bar{\mathbf{W}}_i$ are positive definite and hence invertible.

For each fixed $\mathbf{Z}_i$ with $\|\mathbf{Z}_i\| \leq 1$, the above constraint (3.2) is a so-called linear matrix inequality (LMI) in the matrix variables $\{\mathbf{\Gamma}_{i+1},\ \mathbf{A}_i,\ \mathbf{K}_i\}$ which is convex. (Recall that the matrix variables $\{\mathbf{A}_i,\ \mathbf{K}_i\}$ are buried, linearly, in $\bar{\mathbf{F}}_i,\ \bar{\mathbf{G}}_i$, and $\bar{\mathbf{C}}_i$.) Thus the feasible region described by the above constraint is the intersection of convex regions described by an infinite number of linear matrix inequalities parameterized by $\mathbf{Z}_i$. This implies that the feasible region of (2.13) is convex. It is now clear that the original robust filtering problem (2.13) is equivalent to

$$\begin{aligned} &\text{minimize} && \mathrm{Tr}\left(\mathbf{D}\begin{bmatrix} \mathbf{I} & -\mathbf{I} \end{bmatrix}\mathbf{\Gamma}_{i+1}\begin{bmatrix} \mathbf{I} & -\mathbf{I} \end{bmatrix}^T\right) \\ &\text{subject to} && \{\mathbf{\Gamma}_{i+1},\ \mathbf{A}_i,\ \mathbf{K}_i\} \text{ satisfying (3.2)}, \\ &&& \mathrm{Tr}\,(\mathbf{\Gamma}_{i+1}) \leq b. \end{aligned} \tag{3.3}$$

The formulation (3.3) is given as an SDP, except that the data matrices are subject to uncertainty $\mathbf{Z}_i$. Therefore it cannot be solved by standard SDP methods. The constraints in (3.3) imply that the solution must remain feasible for all allowable perturbations. This is precisely the intent of a robust filter solution. An SDP problem for which the data matrices are uncertain is called a robust SDP. In the next subsection, we introduce a technique for converting a robust SDP into a standard SDP, which can then be solved efficiently by the recent interior point methods.

3.1. The robust SDP. SDP is a convex optimization problem and can be solved in polynomial time using efficient algorithms such as the primal-dual interior point methods [14, 10, 12]. An SDP consists of minimizing a linear objective subject to an LMI constraint,

$$\begin{aligned} &\text{minimize} && \underline{c}^T\underline{\alpha} \\ &\text{subject to} && \mathbf{B}(\underline{\alpha}) = \mathbf{B}_0 + \sum_{k=1}^{q} \alpha_k \mathbf{B}_k \geq \mathbf{0}, \end{aligned}$$

where $\underline{c} \in \mathcal{R}^q$, $\underline{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_q)^T$, and the symmetric matrices $\mathbf{B}_k = \mathbf{B}_k^T \in \mathcal{R}^{l\times l}$, $k = 0, \ldots, q$, are some given data matrices. In our case, these data matrices are subject to uncertainty. We can incorporate some linear uncertainty into $\mathbf{B}(\underline{\alpha})$ in the following way. Let $\mathbf{B}(\underline{\alpha}, \Delta)$ be a symmetric matrix-valued function of two variables $\underline{\alpha}$ and Δ of the form

$$\mathbf{B}(\underline{\alpha}, \Delta) = \mathbf{B}(\underline{\alpha}) + \mathbf{N}\Delta\mathbf{M}(\underline{\alpha}) + \mathbf{M}(\underline{\alpha})^T\Delta^T\mathbf{N}^T, \tag{3.4}$$

where $\mathbf{B}(\underline{\alpha})$ is defined in (3.1), $\mathbf{N}$ and $\mathbf{M}(\underline{\alpha})$ are given matrices, Δ is a perturbation which is unknown but bounded. We define the robust feasible set by

$$\mathcal{A} = \{\underline{\alpha} \in \mathcal{R}^q \mid \mathbf{B}(\underline{\alpha}, \Delta) \geq 0 \text{ for every } \Delta \text{ with } \|\Delta\| \leq 1\}\,.$$

The robust SDP is then defined as

$$\begin{array}{ll} \text{minimize} & \underline{c}^T \underline{\alpha} \\ \text{subject to} & \underline{\alpha} \in \mathcal{A}. \end{array} \tag{3.5}$$

The following lemma shows how such a robust SDP can be solved using a conventional SDP. It is a simple corollary of a classical result on quadratic inequalities referred to as the $\mathcal{S}$-procedure, and its proof is detailed in [2].

LEMMA 3.1. *Let $\mathbf{B} = \mathbf{B}^T$, $\mathbf{N}$, and $\mathbf{M}$ be real matrices of appropriate size. We have*

$$\mathbf{B} + \mathbf{N}\Delta\mathbf{M} + \mathbf{M}^T\Delta^T\mathbf{N}^T \geq \mathbf{0} \tag{3.6}$$

for every Δ, $\|\Delta\| \leq 1$, if and only if there exists a scalar ρ such that

$$\begin{bmatrix} \mathbf{B} - \rho\mathbf{N}\mathbf{N}^T & \mathbf{M}^T \\ \mathbf{M} & \rho\mathbf{I} \end{bmatrix} \geq \mathbf{0}. \tag{3.7}$$

As a consequence, the robust SDP (3.5) *can be formulated as the following standard SDP in variables $\underline{\alpha}$ and ρ:*

$$\begin{array}{ll} \textit{minimize} & \underline{c}^T \underline{\alpha} \\ \textit{subject to} & \begin{bmatrix} \mathbf{B}(\underline{\alpha}) - \rho\mathbf{N}\mathbf{N}^T & \mathbf{M}(\underline{\alpha})^T \\ \mathbf{M}(\underline{\alpha}) & \rho\mathbf{I} \end{bmatrix} \geq \mathbf{0}. \end{array} \tag{3.8}$$

We now return to the problem in (3.3) and factorize the LMI constraint matrix (3.2) according to the structure in (3.4). In such a factorization, the decision variable $\underline{\alpha}$ in (3.4) will correspond to a concatenation of the elements of the matrix variables $\mathbf{\Gamma}_{i+1}$, $\mathbf{A}_i$, and $\mathbf{K}_i$ in (3.2), and the perturbation Δ in (3.4) will correspond to $\mathbf{Z}_i$ in (3.2). The factorization is given by

$$\mathbf{B}(\underline{\alpha}) = \begin{bmatrix} \bar{\mathbf{\Sigma}}_i^{-1} & \mathbf{0} & \bar{\mathbf{F}}_i^T \\ \mathbf{0} & \bar{\mathbf{W}}_i^{-1} & \bar{\mathbf{G}}_i^T \\ \bar{\mathbf{F}}_i & \bar{\mathbf{G}}_i & \mathbf{\Gamma}_{i+1} \end{bmatrix}, \tag{3.9}$$

where

$$\bar{\mathbf{F}}_i = \begin{bmatrix} \mathbf{F}_i & \mathbf{0} \\ \mathbf{K}_i\mathbf{H}_i & \mathbf{A}_i - \mathbf{K}_i\mathbf{H}_i \end{bmatrix},\ \bar{\mathbf{G}}_i = \begin{bmatrix} \mathbf{G}_i & \mathbf{0} \\ \mathbf{0} & \mathbf{K}_i \end{bmatrix},\ \bar{\mathbf{W}}_i = \begin{bmatrix} \bar{\mathbf{Q}}_i + \epsilon\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \bar{\mathbf{R}}_i + \epsilon\mathbf{I} \end{bmatrix},$$

and

$$\mathbf{N}\Delta\mathbf{M}(\underline{\alpha}) + \mathbf{M}(\underline{\alpha})^T\Delta^T\mathbf{N}^T = \begin{bmatrix} \mathbf{0} & \mathbf{0} & (\bar{\mathbf{C}}_i\mathbf{Z}_i\bar{\mathbf{E}}_i)^T \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \bar{\mathbf{C}}_i\mathbf{Z}_i\bar{\mathbf{E}}_i & \mathbf{0} & \mathbf{0} \end{bmatrix} \tag{3.10}$$

with

$$\bar{\mathbf{C}}_i\mathbf{Z}_i\bar{\mathbf{E}}_i = \begin{bmatrix} \mathbf{C}_{1,i}\mathbf{Z}_i\mathbf{E}_i & \mathbf{0} \\ \mathbf{K}_i\mathbf{C}_{2,i}\mathbf{Z}_i\mathbf{E}_i & \mathbf{0} \end{bmatrix}.$$

The matrices $\mathbf{N}$ and $\mathbf{M}(\underline{\alpha})$ are given by

$$\mathbf{M}(\underline{\alpha}) = \left[\begin{array}{cccc} \mathbf{0} & \mathbf{0} & \mathbf{C}_{1,i}^T & \mathbf{C}_{2,i}^T \mathbf{K}_i^T \end{array} \right], \quad \mathbf{N} = \left[\begin{array}{c} \mathbf{E}_i^T \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{array} \right]. \tag{3.11}$$

Now we are in a position to apply Lemma 3.1 to convert the robust SDP (3.3) into the following standard SDP in the variables $\mathbf{\Gamma}_{i+1}$, $\mathbf{A}_i$, $\mathbf{K}_i$, and ρ:

$$\begin{aligned} &\text{minimize} \quad \operatorname{Tr}\left(\mathbf{D}\left[\begin{array}{cc} \mathbf{I} & -\mathbf{I} \end{array}\right]\mathbf{\Gamma}_{i+1}\left[\begin{array}{cc} \mathbf{I} & -\mathbf{I} \end{array}\right]^T\right) \\ &\text{subject to} \quad \left[\begin{array}{cc} \mathbf{B}(\underline{\alpha}) - \rho\mathbf{N}\mathbf{N}^T & \mathbf{M}(\underline{\alpha})^T \\ \mathbf{M}(\underline{\alpha}) & \rho\mathbf{I} \end{array}\right] \geq \mathbf{0}, \\ &\qquad\qquad\quad \operatorname{Tr}(\mathbf{\Gamma}_{i+1}) \leq b, \end{aligned} \tag{3.12}$$

where the variable $\underline{\alpha}$ contains columns of the matrices $\mathbf{\Gamma}_{i+1}$, $\mathbf{K}_i$, and $\mathbf{A}_i$, and the matrices $\mathbf{B}(\underline{\alpha})$, $\mathbf{N}$, and $\mathbf{M}(\underline{\alpha})$ are given by (3.9) and (3.11), respectively.

Note that, for each i, problem (3.12) is fixed in dimension (i.e., does not grow with i). It is a standard SDP problem which has a unique solution and satisfies the usual regularity condition, provided that the primal and dual of (3.12) are strictly feasible and for every $\underline{\alpha}$, $\mathbf{M}(\underline{\alpha}) \neq \mathbf{0}$ and $[\begin{array}{cc} \mathbf{N} & \mathbf{M}^T(\underline{\alpha}) \end{array}]^T$ is full column-rank. As such, the problem can be solved very efficiently by an interior point method; in particular, by the homogeneous self-dual method [10, 12]. In our computational experience, the number of iterations required to solve each SDP is fixed (no more than 8), and therefore the proposed technique can be regarded as a recursive filtering method.

To make a formal comparison of the computational complexity of our robust filtering method with those of [15, 13], we need to recall the notations of our model (2.1): n denotes the number of states, m denotes the number of inputs, and p denotes the number of measured outputs. Xie's method [15] is a "one-shot" method, and hence the robust observer matrix is calculated only once. The cost of this computation is $O((n+p)^3)$. However, Xie's method [15] works only for time-invariant systems. On the other hand, Theodor's method [13] is iterative. The cost per iteration is $O((n+p)^3 + n^2 m)$. Our method is also iterative. Using a general purpose interior point SDP solver requires $O((n+m+p)^{5/2}(n^2+np)^2)$ per filtering iteration. It is interesting to examine the above costs as the number of states in the model, n, grows. In that case, the total computational cost of Xie's method [15] is $O(n^3)$, while the cost per (filtering) iteration of Theodor's method [13] and of our proposed method are $O(n^3)$ and $O(n^{6.5})$, respectively. It is also interesting to examine the above costs as the number of measured outputs, p, grows. In that case, the total cost of Xie's method [15] is $O(p^3)$, while the cost per (filtering) iteration of Theodor's method [13] and of our method are $O(p^3)$ and $O(p^{4.5})$, respectively. We believe it is possible to reduce the complexity per iteration for our method by exploiting the sparsity structure present in our problem. This is an interesting issue for future investigation.

We now make an observation regarding the scaling of the matrices $\bar{\mathbf{C}}_i$ and $\bar{\mathbf{E}}_i$. In particular, these two matrices can be scaled and replaced by $\bar{\mathbf{C}}_i/\mu$ and $\mu\bar{\mathbf{E}}_i$, respectively. Such a scaling does not change the formulation of (3.3), nor does it affect the formulation of (3.12), because the latter is completely equivalent to the former. This shows that the solutions to our reformulated robust filtering problem are independent of the scaling factor μ. This property is in contrast to the robust filter proposed in [13], where the solutions are "highly sensitive" [13] to the choice of μ. The scale invariance of our method with the choice of μ is a clear advantage.

However, our method also has a disadvantage in that it is sensitive to the choice of b in the second constraint in (3.12), $\mathrm{Tr}\,(\mathbf{\Gamma}_{i+1}) \leq b$. This constraint is used to ensure that the matrix $\mathbf{\Gamma}_{i+1}$ is bounded. This is important because otherwise the optimal solution of (3.12), $\bar{\mathbf{\Sigma}}_{i+1}$, may become progressively ill-conditioned as i becomes large. This phenomenon has been observed in computer simulations. In general, large values of b will allow the matrices $\{\bar{\mathbf{\Sigma}}_i : \ i = 1, 2, \ldots\}$ to become rather ill-conditioned, while small values of b may render the subproblem (3.12) infeasible. The same remark applies to the alternative formulation (2.16), where a value of $\beta > 0$ needs to be selected. Through computer experiments we found that both formulations led to filters with similar behavior and performance.

4. Numerical examples. In this section, the performance of the proposed robust state-estimation method is illustrated via simulation results. Two numerical examples are given here; the first one is the same problem as that used in [13, 15], and the second one is a target-tracking problem.

4.1. Example 1. In this example the following discrete-time linear uncertain state-space model is used:

$$\begin{aligned} \underline{x}_{i+1} &= \begin{bmatrix} 0 & -0.5 \\ 1 & 1+\delta \end{bmatrix} \underline{x}_i + \begin{bmatrix} -6 \\ 1 \end{bmatrix} u_i, \quad |\delta| < 0.3, \\ y_i &= \begin{bmatrix} -100 & 10 \end{bmatrix} \underline{x}_i + v_i, \\ s_i &= \begin{bmatrix} 1 & 0 \end{bmatrix} \underline{x}_i, \end{aligned} \tag{4.1}$$

where u_i and v_i are uncorrelated zero-mean white noise signals with variances $\bar{Q} = 1$ and $\bar{R} = 1$, respectively. The value of ϵ in (2.4) is set to zero, so that there is no uncertainty in the knowledge of noise statistics. The uncertainty in (4.1) is described by the matrices

$$\mathbf{C}_1 = [0 \ \ 10]^T, \quad \mathbf{C}_2 = 0, \quad \mathbf{E} = [0 \ \ 0.03]$$

and the scalar parameter z, $|z| \leq 1$.

To determine the robust filter at each instant i, we use the MATLAB toolbox `SeDuMi` [12] to solve the robust SDP (3.12). This code requires no initialization since it is based on the self-dual formulation of the SDP. Solving the SDP (3.12) at each instant i with $b = 900$ and $\mathbf{D} = \mathrm{diag}(1, 5)$ yields a robust state estimator [of the form (2.5)] which converges to

$$\mathbf{A} = \begin{bmatrix} -0.1711 & -0.4624 \\ 1.4080 & 1.1786 \end{bmatrix} \quad \text{and} \quad \mathbf{K} = \begin{bmatrix} -0.0051 \\ 0.0047 \end{bmatrix}.$$

Note that for stability reasons the estimator (as seen in $\mathbf{D}$) weights the second component of $\underline{x}_i$ more heavily than the first component even though the goal is to estimate the first component of $\underline{x}_i$. In our simulation studies, the proposed technique is compared with the Kalman and H^∞ filters and the robust filters of [13, 15]. For this purpose, steady-state Kalman and H^∞ filters are designed for the nominal system of (4.1), i.e., $\delta = 0$. We then apply these filters to system (4.1) with $\delta = 0$, $\delta = 0.3$,

Table 1
Steady-state estimation error variances for different filters (results are averaged over 100 *runs).*

Filter	$\delta = -0.3$	$\delta = 0$	$\delta = 0.3$
Nominal Kalman filter	551.2	36.0	8352.8
Nominal H^∞ filter	96.0	47.2	893.9
The robust filter of [13]	51.4	51.3	54.4
The robust filter of [15]	64.0	61.4	64.4
The robust filter of [3]	51.5	49.1	53.8
Proposed robust filter	46.2	45.6	51.9

Table 2
Steady-state estimation error variances for our method and the method of [3].

Filter	$\delta = -0.09$	$\delta = 0$	$\delta = 0.09$
The robust filter of [3]	37.75	38.19	41.47
Proposed robust filter	37.38	37.78	40.31

and $\delta = -0.3$. The steady-state estimation error variances (i.e., $\mathcal{E}\{(s_i - \hat{s}_i)^2\}$ for sufficiently large i) for the filters are displayed in Table 1. It is clear from the table that the proposed robust filter performs far better than the nominal Kalman and H^∞ filters in the presence of modelling error.

Both our method and the methods of [13, 15] require the tuning of a certain parameter. In our case, we need to adjust the parameter b in order to prevent the iterates from becoming ill-conditioned, and the diagonal elements of $\mathbf{D}$ in order to get the best estimator. The methods of [13, 15] require the adjustment of the factor μ in the scaling of $\mathbf{C}_i/\mu$ and $\mu\mathbf{E}_i$. Our experiments suggest that our method works for $b \in [880, 5000]$, while the method of [13] converges for $\mu \in (0, 1.703]$ and diverges for values outside this range. The best performance is achieved with $\mu = 1.703$. (Note that the authors of [13] reported their choice of $\mu = 2.2$, but our own implementation of their method showed that this value of μ leads to divergence instead.)

The filter performance for the robust filter of [15] stated in Table 1 is quoted from [13]. We should point out that we could not reproduce the design of the robust filter [13] using their design method. With our own (simple) MATLAB implementation of their method, we could only produce a filter with $\mu = 1.703$, whose error covariances are 51.4, 51.3, and 54.4, rather than 46.6, 45.2, and 54.1 (as claimed in [13]) for model errors of $\delta = -0.3$, 0, and 0.3, respectively. From Table 1, we can see that the performance of the robust filters [13, 15] are inferior to the filter designed by the robust SDP method: the worst-case performance (for $\delta = -0.3,\ 0,\ 0.3$) is 51.9 for our proposed robust filter, and is 54.2 and 64.4, respectively, for the robust filters of [13] and [15]. From this example, it appears that our robust filter design is slightly superior.

Recently our approach has been further extended by Fu, de Souza, and Luo [3], who introduced multiple scaling factors in the SDP formulation and showed performance improvement when compared to the single-scaling-factor case. It should be pointed out that the single-scaling-factor case of [3] corresponds to the algorithm considered in this paper, except that we have an additional boundedness constraint $\mathrm{Tr}\,(\mathbf{\Gamma}_{i+1}) \le b$ in our SDP subproblem (3.12). We simulated the single-scaling-factor case of [3] in Table 1 for comparison. From the simulation results, our method is slightly superior to the method of [3] in the single-scaling-factor case. This is due to the differences in the way the ill-conditioning of the bound on the covariance matrix is handled. The simulation results stated in [3] are for $\mathbf{C}_1 = [0\ \ 3]^T$ (instead of

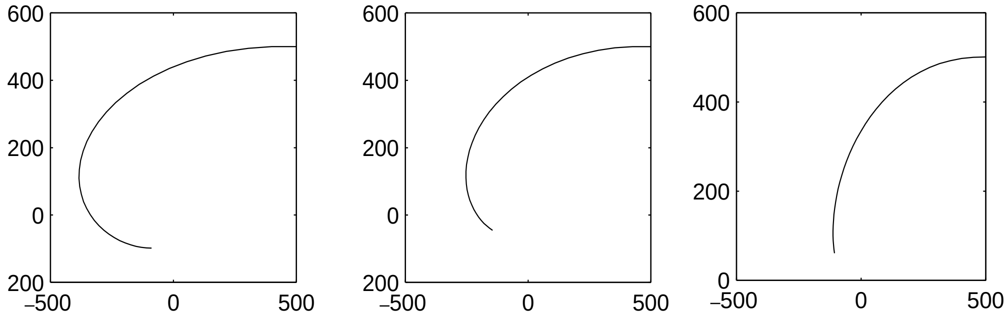

FIG. 1. *Trajectories of the target-tracking model with uncertainty* $\delta = -0.05$ *(left),* $\delta = 0$ *(middle),* $\delta = 0.05$ *(right).*

$\mathbf{C}_1 = [0 \;\; 10]^T$), which means that the simulated cases in [3] have only 30% of the uncertainty considered in Table 1. We also compared our method with the method of [3] for the case $\mathbf{C}_1 = [0 \;\; 3]^T$, and the simulation results show that our method is still slightly superior to the method of [3] (Table 2).

4.2. A tracking example. In this example a target-tracking case is considered. The discrete-time state-space model is given by

$$
\begin{aligned}
\underline{x}_{i+1} &= \begin{bmatrix} 0.95 & -0.1+\delta \\ 0.05 & 0.95 \end{bmatrix} \underline{x}_i + \begin{bmatrix} 1 \\ 1 \end{bmatrix} u_i, \quad |\delta| < 0.05, \\
y_i &= [\,1 \quad 0\,]\,\underline{x}_i + v_i, \\
s_i &= [\,1 \quad 0\,]\,\underline{x}_i,
\end{aligned} \tag{4.2}
$$

where u_i and v_i are uncorrelated zero-mean white noise signals with variances $\bar{Q} = 1$ and $\bar{R} = 1$, respectively. The value of ϵ in (2.4) is set to zero, so that there is no uncertainty in the knowledge of noise statistics. The uncertainty in (4.2) is described by the matrices

$$
\mathbf{C}_1 = [0.05 \;\; 0]^T, \quad \mathbf{C}_2 = 0, \quad \mathbf{E} = [0 \;\; 1]
$$

and the uncertainty parameter z, $|z| \leq 1$.

In this model, the state vector $\underline{x}_i$ represents the position of a target in a two-dimensional coordinate system, and the observation y_i is a noise-corrupted version of the first coordinate. The target is making a counter-clockwise turn starting from the position $\underline{x}_0 = [500, 500]^T$. The unknown parameter δ describes the uncertainty in the turning rate of the trajectory. Three possible trajectories from this model are shown in Figure 1.

Solving the SDP (3.12) for each value of i, with $b = 1100$ and $\mathbf{D} = \mathrm{diag}(1, 7)$, yields a robust state estimator (of the form (2.5)) which converges to

$$
\mathbf{A} = \begin{bmatrix} 0.9500 & -0.1016 \\ 0.0500 & 0.9644 \end{bmatrix} \quad \text{and} \quad \mathbf{K} = \begin{bmatrix} 0.7560 \\ 0.0130 \end{bmatrix}.
$$

Table 3
Steady-state estimation error variances for different filters for the tracking problem (results are averaged over 100 *runs).*

Filter	$\delta = -0.05$	$\delta = 0$	$\delta = 0.05$
Nominal Kalman filter	6425.2	1.4	11404.0
The robust filter of [13]	199.7	53.6	703.5
The robust filter of [15]	1309.6	666.9	549.2
Proposed robust filter	187.9	52.8	693.4

We have compared our method with the methods of [13, 15], as well as the nominal Kalman filter. The result is shown in Table 3.

From the simulation results, it appears that the filter designed by our method is superior to the filters obtained via the methods of [13] and [15]. In designing the filters by the methods of [13, 15], we have adjusted their corresponding adjustable parameters (e.g., the parameter μ in the scaling of $\mathbf{C}_i/\mu$ and $\mu\mathbf{E}_i$) and picked the filters which generate the best performance guarantees. The method of [15] requires that an additional parameter, denoted ϵ in [15], be tuned. We tuned this parameter to a value of 10 in our implementation. Note that, in the presence of uncertainty, the nominal Kalman filter performs far worse than the robust filters, as expected.

We have also compared our robust filter design to the robust filters of [13, 15] in higher-dimensional cases. We found that the relative steady-state performance of these filters is similar to that in the above examples. From the computational standpoint, our method is quite efficient, as the SDP solved at each instant has a fixed dimension, and the interior point method used to solve it is fast. However, our method does incur a greater per-sample computational cost than methods based on *analytic* recursions, such as the Kalman filter and the robust Kalman filter in [13]. (The robust filter in [15] is a "one-shot" filter which does not vary with i.) For example, on a 200MHz Pentium Pro workstation, using a general purpose SDP solver [12] under the Matlab environment, the per-sample computation time of our method in the above examples was around 1s, whereas that of the method in [13] was around 5ms. (Recall, however, that the performance of the method in [13] is "highly sensitive" to the parameter which must be tuned.) In future work, it will be useful to design special purpose interior point algorithms which exploit the matrix structure of the SDP in (3.12) to reduce the per-sample computational complexity of our new method. Such a reduction of computational complexity is essential if one is to implement the proposed robust filtering algorithm on a DSP (digital signal processing) chip for a real-time filtering application.

5. Conclusions. In this paper, we have proposed a new state estimator for linear uncertain systems. The method is robust to norm-bounded parameter uncertainties on the system model as well as to bounded uncertainties on the noise statistics. In the new technique, the estimation problem was formulated as a convex optimization problem, which is then solved using the recent primal-dual self-dual interior point method. This requires at most 8 iterations (or matrix inversions) and therefore can be regarded as a recursive filtering method. The formulation guarantees the existence of robust solutions via a semidefinite program and, under some conditions, the solution to that semidefinite program is unique. The proposed technique compared favorably with the well-known Kalman and H^∞ filters and the "robust" filters of [13, 15]. When applied to the problem of target tracking, the new method has led to a significant improvement in tracking performance.

REFERENCES

[1] V. D. Blondel and J. N. Tstitsiklis, *A survey of computational complexity results in systems and control*, Automatica J. IFAC, 36 (2000), pp. 1249–1274.

[2] L. El Ghaoui and H. Lebret, *Robust solution to least-squares problems with uncertain data*, SIAM J. Matrix Anal. Appl., 18 (1997), pp. 1035–1064.

[3] M. Fu, C. E. de Souza, and Z.-Q. Luo, *Finite horizon robust Kalman filter design*, IEEE Trans. Signal Processing, to appear.

[4] M. Green and D. J. N. Limebeer, *Linear Robust Control*, Prentice–Hall, Englewood Cliffs, NJ, 1995.

[5] B. Hassibi, A. H. Sayed, and T. Kailath, *Linear estimation in Krein spaces—Part* I*: Theory & Part* II*: Applications*, IEEE Trans. Automat. Control, 41 (1996), pp. 18–49.

[6] S. Haykin, *Adaptive Filter Theory*, 3rd ed., Prentice–Hall, Englewood Cliffs, NJ, 1996.

[7] D. Hinrichsen and A. J. Pritchard, *Robustness measures for linear systems with application to stability radii of Hurwitz and Schur polynomials*, Internat. J. Control, 55 (1992), pp. 809–844.

[8] T. Kailath, A. H. Sayed, and B. Hassibi, *Linear Estimation*, Prentice–Hall, Upper Saddle River, NJ, 2000.

[9] H. Li and M. Fu, *A linear matrix inequality approach to robust H_∞ filtering*, IEEE Trans. Signal Process., 45 (1997), pp. 2238–2250.

[10] Z.-Q. Luo, J. F. Sturm, and S. Zhang, *Conic convex programming and self-dual embedding*, Optim. Methods Softw., 14 (2000), pp. 169–218.

[11] A. H. Sayed and T. Kailath, *A state-space approach to adaptive RLS filtering*, IEEE Signal Processing Magazine, 11 (1994), pp. 18–60.

[12] J. F. Sturm, *Using SeDuMi* 1.02*, a MATLAB toolbox for optimization over symmetric cones*, Optim. Methods Softw., 11–12 (1999), pp. 625–653. Available online at http://fewcal.kub.nl/sturm/software/sedumi.html.

[13] Y. Theodor and U. Shaked, *Robust discrete-time minimum-variance filtering*, IEEE Trans. Signal Process., 44 (1996), pp. 181–189.

[14] L. Vandenberghe and S. Boyd, *Semidefinite programming*, SIAM Rev., 38 (1996), pp. 49–95.

[15] L. Xie, Y. C. Soh, and C. E. de Souza, *Robust Kalman filtering for uncertain discrete-time systems*, IEEE Trans. Automat. Control, 39 (1994), pp. 1310–1314.

[16] L. Xie, C. E. de Souza, and M. Fu, *H_∞ estimation for linear discrete-time uncertain systems*, Internat. J. Robust Nonlinear Control, 1 (1991), pp. 111–123.

[17] K. Zhou, J. C. Doyle, and K. Glover, *Robust and Optimal Control*, Prentice–Hall, Englewood Cliffs, NJ, 1996.