Automatica 95 (2018) 33–43

Contents lists available at ScienceDirect

Automatica

journal homepage: www.elsevier.com/locate/automatica

Exact recursive updating of state uncertainty sets for linear SISO systems ☆

Robin Hill [a,*], Yousong Luo [b], Uwe Schwerdtfeger [c]

[a] Department of Electrical and Electronic Engineering, University of Melbourne, Melbourne, Vic 3010, Australia
[b] School of Science, RMIT University, 124 Latrobe St, Melbourne, 3001, Australia
[c] Fakultät für Mathematik, Technische Universität Chemnitz, D-09107 Chemnitz, Germany

ARTICLE INFO

Article history:
Received 14 December 2016
Received in revised form 3 February 2018
Accepted 19 April 2018

Keywords:
Estimation
Linear systems
Uncertainty
Linear programming
Duality

ABSTRACT

This paper addresses the classical problem of determining the set of possible states of a linear discrete-time SISO system subject to bounded disturbances, from measurements corrupted by bounded noise. These so-called uncertainty sets evolve with time as new measurements become available. We present two theorems which give a complete description of the relationship between uncertainty sets at two successive time instants, and this yields an efficient algorithm for recursively updating uncertainty sets. Numerical simulations demonstrate performance improvements over existing exact methods.

1. Introduction

Consider a linear, time-invariant dynamic SISO system driven by set-bounded process noise, and with measurements corrupted by set-bounded observation noise. The set of possible states of the system consistent with the measurements up to the current time is termed the *state uncertainty set* (or simply *uncertainty set*). In many applications having a representation of the uncertainty set is useful. This so-called set membership estimation problem is fundamental and has many applications, for example in fault detection (Alamo, Bravo, & Camacho, 2005; Casau, Rosa, Tabatabaeipour, Silvestre, & Stoustrop, 2015; Rosa, Silvestre, Shamma, & Athans, 2010; Tabatabaeipour, 2015; Tornil-Sin, Ocampo-Martinez, Puig, & Escobet, 2012), control under constraints in the presence of noise (Bertsekas & Rhodes, 1971; Glover & Schweppe, 1971), and model (in)validation (Poolla, Khargonekar, Tikku, Krause, & Nagpal, 1994; Rosa, Silvestre, & Athans, 2014). A closely related topic is identification of bounded-parameter models (Belforte, Bona, & Cerone, 1990; Clement & Gentil, 1990; Norton, 1987).

The first results on recursive determination of the uncertainty set are in Schweppe (1968) and Witsenhausen (1968). Since the appearance of these papers there has appeared an extensive literature on the topic. See Fogel and Huang (1982) and Ninness and Goodwin (1995) for background on the set-bounded approach to uncertainty, the survey paper (Milanese & Vicino, 1991) and the book (Blanchini & Biani, 2008). Some of the many other papers which consider this problem are Blanchini and Sznaier (2012), Stoorvogel (1996) and Tempo (1988).

In the first part of the seminal paper (Witsenhausen, 1968) an exact in principle solution to the problem of recursively determining polytopic uncertainty sets is given. It uses the $\mathcal{H}$-representation for the uncertainty sets, that is they are defined using inequality constraints. But the solution requires (Minkowski) addition, and intersections, of polytopes, both of which can be time-consuming. Exact, recursive $\mathcal{H}$-representation methods often use Fourier–Motzkin elimination or parametric linear programming, see Keerthi and Gilbert (1987), Rakovic and Mayne (2004) and Shamma and Tu (1999) for the former, and Jones, Kerrigan, and Maciejowski (2008) for the latter. In these implementations it is the identification and removal of redundant inequality constraints that is most demanding computationally. The redundant constraints can be removed by solving linear programs but this is not a trivial task, for which only weak polynomiality is known if only the $\mathcal{H}$-representation of the polytope is available. For this reason there has been a lot of research recently on the use of zonotopes and constrained zonotopes to approximate the exact polytopic uncertainty set, see for example Alamo, Bravo, Redondo, and Camacho (2008), Combastel (2015) and Scott, Raimondo, Marseglia, and

☆ The material in this paper was partially presented at the 56th IEEE Conference on Decision and Control, December 12–15, 2017, Melbourne, Australia. This paper was recommended for publication in revised form by Associate Editor Antonio Vicino under the direction of Editor Torsten Söderström.

* Corresponding author.
E-mail addresses: robin.hill@unimelb.edu.au (R. Hill), yluo@rmit.edu.au (Y. Luo), uwe.schwerdtfeger@mathematik.tu-chemnitz.de (U. Schwerdtfeger).

https://doi.org/10.1016/j.automatica.2018.05.010
0005-1098/

Braatz (2016). For hardness results on polytopic computations, see Tiwary (2008).

Another interesting recent approach using exact methods, based on geometric ideas, is in Hagemann (2014). Here also an inequality description is used, and projection followed by redundant inequality constraint elimination is necessary.

In this paragraph and the next we describe the idea introduced in this paper, and the key role played by duality in its implementation. A state in an uncertainty set at the current time has a history, the trajectory of the plant's state at previous times. For any current state there must be at least one such trajectory, and it is uniquely determined by the initial state and some sequence of noisy inputs up to the current time. Our starting point is the question: Given a state of the plant in the uncertainty set at the current time, and the current measurement, what possible states are there in the uncertainty set at the next time instant that lie on trajectories containing the current state? Now as stated the question is not well posed; knowing the state and measurement is not enough. Additionally, something about the current uncertainty set is required in order to determine the forward evolution of the trajectory. We show that the current measurement, the current state, and the directions of just those facets of the current uncertainty set that contain the current state, are precisely what is needed to determine all successor states which lie on trajectories containing the current state. In fact, using just this information, much more can be said. For any such successor to the current state, the directions of all the facets of the uncertainty set at the next time instant that contain the successor state can be found. This is our main result. It enables the vertex/facet description of the current uncertainty set to be efficiently and recursively updated to the vertex/facet description of the uncertainty set at the next time instant.

The results described in the previous paragraph are derived using properties of optimal solutions to a primal/dual pair of optimisation problems. Every state in the uncertainty set, including those in the interior, can be interpreted as being in the argmax of a mathematical programming problem related to the support function for the uncertainty set. This is the primal problem. The key to our results involves setting up a dual program, whose dual states have an interpretation as direction vectors which are arguments of the support function. Optimal solutions to the primal problem yield the time evolution of a state along a trajectory. Such an evolving state may either be on the boundary or in the interior of the current uncertainty set; if on the boundary then whether or not the state is a vertex can also be determined. Optimal solutions to the dual problem yield the time evolution of the directions of the facets which contain the evolving primal states. It is a consequence of this primal/dual framework that the proposed recursive method both requires, and makes optimal use of, vertex and facet descriptions of the uncertainty sets.

Some of the results in this paper build on ideas in Witsenhausen (1968), particularly that of support function evolution. We use linear programming rather than conjugate functions as our basic tool, and employ the familiar complementary slackness conditions relating primal and dual variables to prove our main results. However, our procedure for recursive updating of uncertainty sets does not require the numerical solution of any linear programs.

2. Basic setup

The plant P, a linear, time-invariant, causal discrete-time, m^{th} order scalar system, is assumed known. There are two sources of uncertainty, an input noise disturbance $(u_k)_{k=0}^{\infty} = \mathbf{u}$, and output measurement noise $(w_k)_{k=0}^{\infty} = \mathbf{w}$. The plant output is $(y_k)_{k=0}^{\infty} = \mathbf{y}$, and the measurement at time k is $z_k = y_k + w_k$. The initial state, at time $k = 0$, is assumed to be known exactly, but nothing is known about the uncertainties except that they satisfy $|u_k| \leq 1$ and $|w_k| \leq 1$. We will refer to this as the primal system.

Given an initial state $\mathbf{x}_0$, the measurement history $z_1, \ldots, z_{k-1}$, and the plant dynamics, we seek the uncertainty set at time k, denoted S_k; it is the set of possible states at time k consistent with the measurements up to and including z_{k-1}, and can be shown to be a closed, convex polytope.

2.1. Notation

Given a vector $\mathbf{y} = (y_0, y_1, \ldots)$ and any $s \in \mathbb{N}$, $t \in \mathbb{N}$ satisfying $s < t$, we denote $(y_s, y_{s+1}, \ldots, y_t)$ by $y_{s:t}$. Unless explicitly stated otherwise, vectors in matrix equations are column vectors, and the superscript$^{\mathrm{T}}$ denotes transpose, so $\mathbf{y}$ is a column vector and $\mathbf{y}^{\mathrm{T}}$ is a row vector. The λ-transform (generating function) of an arbitrary sequence $\mathbf{y} = (y_k)_{k=0}^{\infty}$ is defined to be $\hat{\mathbf{y}}(\lambda) := \sum_{k=0}^{\infty} y_k \lambda^k$. Real Euclidean space of dimension m is denoted $\mathbb{R}^m$, where m is the order (McMillan degree) of the plant P. States of P are represented by vectors, or points, in $\mathbb{R}^m$. Let $\mathbf{d} = d_{0:m} = (d_0, \ldots, d_m)$ and $\mathbf{n} = n_{0:m} = (n_0, \ldots, n_m)$, be real vectors, where $\hat{\mathbf{n}}(\lambda)$ and $\hat{\mathbf{d}}(\lambda)$ are the numerator and denominator of the transfer function representation of P. Denote by $\mathbf{D}_\infty$ and $\mathbf{N}_\infty$ the infinite, banded, lower-triangular Toeplitz matrices whose first columns are $\mathbf{d}$ and $\mathbf{n}$, respectively. Define the following lower- and upper-triangular submatrices of $\mathbf{D}_\infty$.

$$\mathbf{D}_{\mathrm{L}} := \begin{bmatrix} d_0 & 0 & \ldots & 0 \\ d_1 & d_0 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ d_{m-1} & \ldots & d_1 & d_0 \end{bmatrix}$$

$$\mathbf{D}_{\mathrm{U}} := \begin{bmatrix} d_m & d_{m-1} & \ldots & d_1 \\ 0 & d_m & \ddots & \vdots \\ \vdots & \ddots & \ddots & d_{m-1} \\ 0 & \ldots & 0 & d_m \end{bmatrix}.$$

The matrices $\mathbf{N}_{\mathrm{L}}$ and $\mathbf{N}_{\mathrm{U}}$ are defined similarly.

For any $k \geq 0$, the $k \times k$ upper left hand corner submatrix of $\mathbf{D}_\infty$ is denoted $\mathbf{D}_k$. We will often write simply $\mathbf{D}$ instead of $\mathbf{D}_k$ when k is clear from context. The symbols $\mathbf{N}_k$ and $\mathbf{N}$ are defined similarly. Note that $\mathbf{D}_m = \mathbf{D}_{\mathrm{L}}$ and $\mathbf{N}_m = \mathbf{N}_{\mathrm{L}}$.

The Toeplitz Bezoutian matrix of $\mathbf{n}$ and $\mathbf{d}$ is defined as $\mathbf{B}_{\mathrm{T}} := \mathbf{D}_{\mathrm{L}}\mathbf{N}_{\mathrm{U}} - \mathbf{N}_{\mathrm{L}}\mathbf{D}_{\mathrm{U}}$.

One form of the Gohberg–Semencul formulas (Fuhrmann, 1996; Gohberg & Semencul, 1972) states

$$\mathbf{B}_{\mathrm{T}} = \mathbf{N}_{\mathrm{U}}\mathbf{D}_{\mathrm{L}} - \mathbf{D}_{\mathrm{U}}\mathbf{N}_{\mathrm{L}}, \tag{1}$$

and this will be needed in the proof of Theorem 8, which underpins all of our results. The first row of $\mathbf{B}_{\mathrm{T}}$ plays an important role and will be denoted by $\mathbf{C}$.

The inverse of $\mathbf{B}_{\mathrm{T}}$ exists if the polynomials $\hat{\mathbf{n}}(\lambda)$ and $\hat{\mathbf{d}}(\lambda)$ are coprime, and $\mathbf{B}_{\mathrm{T}}^{-1}$ denotes the inverse of $\mathbf{B}_{\mathrm{T}}$. See Heinig and Rost (2010) for properties of Bezoutians.

2.2. Transfer function description and state-space representations

The plant for the primal system has the transfer function representation $P(\lambda) = \hat{\mathbf{n}}(\lambda)/\hat{\mathbf{d}}(\lambda)$ where

$$\hat{\mathbf{n}}(\lambda) = n_0 + n_1\lambda + n_2\lambda^2 + \ldots + n_m\lambda^m$$
$$\hat{\mathbf{d}}(\lambda) = d_0 + d_1\lambda + d_2\lambda^2 + \ldots + d_m\lambda^m,$$

$m \geq 1$ is an integer, $\hat{\mathbf{n}}(\lambda)$ and $\hat{\mathbf{d}}(\lambda)$ are assumed to be coprime polynomials with real coefficients, and it is assumed that both the plant $P(\lambda)$ and the plant $P^*(\lambda)$ for the dual system, defined below, are causal, implying $d_0 \neq 0$ and $d_m \neq 0$, in which case the