Computation of the Worst-Case Covariance for Linear Systems with Uncertain Parameters

V. Balakrishnan and S. Boyd*
Information Systems Laboratory
Electrical Engineering Department
Stanford University, Stanford CA 94305
(*In Proc. CDC, 1991*)

Abstract

For a class of linear systems with unknown parameters that lie in intervals, we present a branch and bound algorithm for computing the worst-case covariance of the state.

1 Introduction

We consider the family of linear time-invariant systems described by

$$\begin{array}{rcl} \dot{x} & = & Ax + B_u u + B_w w, \quad x(0) = 0, \\ y & = & C_y x, \\ z & = & C_z x, \\ u & = & \Delta\, y, \end{array} \tag{1}$$

where $x(t) \in \mathbf{R}^n$, $w(t) \in \mathbf{R}^{n_i}$, $z(t) \in \mathbf{R}^{n_o}$, $u(t), y(t) \in \mathbf{R}^p$, and A, B_u, B_w, C_y and C_z are real matrices of appropriate sizes. Δ is a diagonal matrix, parametrized by a vector of parameters $q = [q_1, q_2, \ldots, q_m]$, and is given by

$$\Delta = \mathrm{diag}(q_1 I_1, q_2 I_2, \ldots, q_m I_m), \tag{2}$$

where I_i is an identity matrix of size p_i. Of course, $\sum_i^m p_i = p$. The rectangle in which q lies is given by $\mathcal{Q}_{\rm init} = [l_1, u_1] \times [l_2, u_2] \times \cdots \times [l_m, u_m]$.

Eliminating u and y from equations (1) yields the closed-loop system equations:

$$\begin{array}{rcl} \dot{x} & = & \mathcal{A}(q)x + B_w w, \\ z & = & C_z x, \end{array} \tag{3}$$

where $\mathcal{A}(q) = A + B_u \Delta C_y$. We note that the entries of $\mathcal{A}(q)$ are *affine functions* of the parameter vector q.

Loosely speaking, the above framework describes a class of linear systems with fixed, unknown gains that lie in intervals. Many important questions arise for such systems: robust stability, stability margin, minimum stability degree *etc.* (see [1] for a brief discussion of such questions). In this paper, we will describe the computation of the of largest possible trace of the state covariance, when w is unit-intensity white noise, *i.e.*,

$$\mathcal{C}(\mathcal{Q}_{\rm init}) = \max_{q \in \mathcal{Q}_{\rm init}} \lim_{t \to \infty} \mathrm{Tr}\ \mathbf{E}\ \ x_q(t) x_q(t)^T, \tag{4}$$

where x_q is the solution to the state equations corresponding to the parameter vector q, $\mathbf{E}$ stands for the expected value and $\mathrm{Tr}\, M$ is the trace (sum of diagonal entries) of the square matrix M. We assume that the system (1) is robustly stable, that is $\mathcal{A}(q)$ has eigenvalues with negative real part for all $q \in \mathcal{Q}_{\rm init}$. For convenience, we let $X(q) = \lim_{t\to\infty} \mathbf{E}\ \ x_q(t)^T x_q(t)$.

For a fixed value of q, $X(q)$ can be computed as the unique solution to the Lyapunov equation

$$\mathcal{A}(q)X(q) + X(q)\mathcal{A}(q)^T + B_w B_w^T = 0. \tag{5}$$

We may therefore rewrite equation (4) as

$$\mathcal{C}(\mathcal{Q}_{\rm init}) = \max_{q \in \mathcal{Q}_{\rm init}} \left\{ \mathrm{Tr}\ (X(q)) \;\middle|\; \begin{array}{r} \mathcal{A}(q)X(q) + X(q)\mathcal{A}(q)^T \\ + B_w B_w^T = 0 \end{array} \right\}.$$

$\mathcal{C}(\mathcal{Q}_{\rm init})$ is the maximum possible sum of the covariance of the state components when the system is driven by unit-intensity white noise, and serves as a measure of the robustness of the system.

There are no known analytic methods that compute $\mathcal{C}(\mathcal{Q}_{\rm init})$ exactly. However, for any rectangle $\mathcal{Q}$, it is possible to compute upper and lower bounds for $\mathcal{C}(\mathcal{Q})$. These bounds may be used with a branch and bound technique to compute $\mathcal{C}(\mathcal{Q}_{\rm init})$ to within any given accuracy $\epsilon > 0$. We first describe a branch and bound algorithm, and then describe the computation of simple upper and lower bounds for $\mathcal{C}(\mathcal{Q})$.

2 The Branch and Bound Algorithm

The branch and bound algorithm we present here is a minor variation on the one presented in [2]. It finds the maximum of a function $f : \mathbf{R}^m \to \mathbf{R}$ over an m-dimensional rectangle $\mathcal{Q}_{\rm init}$ (the subscript "init" stands for *initial* rectangle).

For a rectangle $\mathcal{Q} \subseteq \mathcal{Q}_{\rm init}$ we define

$$\Phi_{\max}(\mathcal{Q}) = \max_{q \in \mathcal{Q}} f(q).$$

Then, the algorithm computes $\Phi_{\max}(\mathcal{Q}_{\rm init})$ to within an absolute accuracy of $\epsilon > 0$, using two functions $\Phi_{\rm lb}(\mathcal{Q})$ and $\Phi_{\rm ub}(\mathcal{Q})$ defined over $\{\mathcal{Q} : \mathcal{Q} \subseteq \mathcal{Q}_{\rm init}\}$ (which, presumably, are easier to compute than $\Phi_{\max}(\mathcal{Q})$). These two functions must satisfy the following conditions:

(R1) $\Phi_{\rm lb}(\mathcal{Q}) \le \Phi_{\max}(\mathcal{Q}) \le \Phi_{\rm ub}(\mathcal{Q})$.

*Research supported in part by NSF under ECS-85-52465, AFOSR under 89-0228, and Bell Communications Research.

(R2) As the maximum length of the sides of $\mathcal{Q}$, denoted by size($\mathcal{Q}$), goes to zero, the difference between upper and lower bounds *uniformly* converges to zero, *i.e.*,

$$\forall\, \epsilon > 0\; \exists\, \delta > 0 \text{ such that}$$
$$\forall\, \mathcal{Q} \subseteq \mathcal{Q}_{\rm init},\ \text{size}(\mathcal{Q}) \le \delta \Longrightarrow \Phi_{\rm ub}(\mathcal{Q}) - \Phi_{\rm lb}(\mathcal{Q}) \le \epsilon.$$

We now state the algorithm. The reader is referred to [2] for details.

The general branch and bound algorithm

In the following description, k stands for the iteration index. $\mathcal{L}_k$ denotes the list of rectangles, L_k the lower bound and U_k the upper bound for $\Phi_{\max}(\mathcal{Q}_{\rm init})$, at the end of k iterations.

$k = 0$;
$\mathcal{L}_0 = \{\mathcal{Q}_{\rm init}\}$;
$L_0 = \Phi_{\rm lb}(\mathcal{Q}_{\rm init})$;
$U_0 = \Phi_{\rm ub}(\mathcal{Q}_{\rm init})$;
while $U_k - L_k > \epsilon$, {
  pick $\mathcal{Q} \in \mathcal{L}_k$ *such that* $\Phi_{\rm ub}(\mathcal{Q}) = U_k$;
  split $\mathcal{Q}$ *into* $\mathcal{Q}_I$ *and* $\mathcal{Q}_{II}$
    along the longest edge;
  $\mathcal{L}_{k+1} := (\mathcal{L}_k - \{\mathcal{Q}\}) \cup \{\mathcal{Q}_I, \mathcal{Q}_{II}\}$;
  $L_{k+1} := \max_{\mathcal{Q} \in \mathcal{L}_{k+1}} \Phi_{\rm lb}(\mathcal{Q})$;
  $U_{k+1} := \max_{\mathcal{Q} \in \mathcal{L}_{k+1}} \Phi_{\rm ub}(\mathcal{Q})$;
  $k := k + 1$;
}

At the end of k iterations, U_k and L_k are upper and lower bounds respectively for $\Phi_{\max}(\mathcal{Q}_{\rm init})$. Since $\Phi_{\rm lb}(\mathcal{Q})$ and $\Phi_{\rm ub}(\mathcal{Q})$ satisfy condition (R2), $(U_k - L_k)$ is guaranteed to converge down to zero.

3 Bounds for $\mathcal{C}(\mathcal{Q})$

Lower Bound

A simple lower bound $\underline{\mathcal{C}}(\mathcal{Q})$ for $\mathcal{C}(\mathcal{Q})$ is just $\mathbf{Tr}\,(X(q_c))$ where q_c is the center of $\mathcal{Q}$. More sophisticated lower bounds may be obtained using local optimization methods (see the survey [4]).

Upper Bound

An upper bound $\overline{\mathcal{C}}(\mathcal{Q})$ for $\mathcal{C}(\mathcal{Q})$ is based on a simple perturbation analysis of the solution to a system of linear equations. We refer the reader to [1] for details.

$$\overline{\mathcal{C}}(\mathcal{Q}) = \begin{cases} \infty & \text{if } \sigma_{\min}(S_{A(q_c)}) \le \alpha, \\ \underline{\mathcal{C}}(\mathcal{Q}) + \dfrac{\alpha\sqrt{n}\|X(q_c)\|_F}{\sigma_{\min}(S_{A(q_c)}) - \alpha} & \text{if } \sigma_{\min}(S_{A(q_c)}) > \alpha. \end{cases}$$

S_M is the $n^2 \times n^2$ matrix representing the Lyapunov operator corresponding to the $n \times n$ matrix M, and is given by $S_M = M \otimes I + I \otimes M$, where "$\otimes$" denotes the Kronecker product [3]. $\sigma_{\min}(M)$ and $\sigma_{\max}(M)$ denote the smallest and largest singular values of M respectively, and $\|M\|_F$ denotes the Frobenius norm of M. For convenience α has been used to denote the quantity $2\sigma_{\max}(B_u)\text{size}(\mathcal{Q})\sigma_{\max}(C_y)$.

Other upper bounds are possible; see, for example, [5].

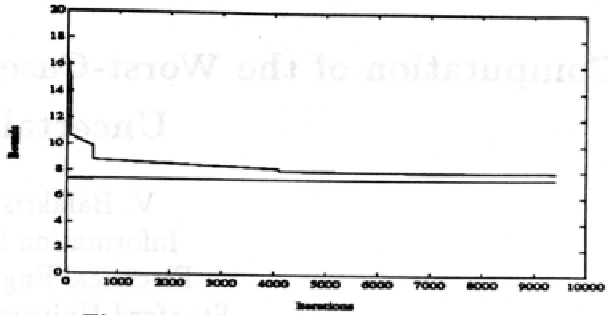

Figure 1: Bounds on $\mathcal{C}(\mathcal{Q}_{\rm init})$ for the example.

4 An Example

We consider an example with

$$A = \begin{bmatrix} -3.4121 & -0.3507 & -0.6183 \\ 1.5654 & 0.2706 & 0.9118 \\ -5.7336 & -12.6285 & -5.8585 \end{bmatrix},$$

$$B_u = \begin{bmatrix} 0.3323 & -0.1176 & 0.2036 \\ -0.4138 & 0.0659 & -0.1773 \\ -0.0152 & 0.1618 & -0.1675 \end{bmatrix},$$

$$C_y = \begin{bmatrix} -0.0989 & -0.0823 & 0.0015 \\ 0.1037 & 0.1956 & -0.0674 \\ -0.0058 & -0.0131 & -0.0525 \end{bmatrix},$$

$$B_w = \begin{bmatrix} 0.3840 & 0.6101 & -1.7705 \\ 0.8395 & 0.4785 & -0.3519 \\ 0.4718 & 0.6206 & -0.2265 \end{bmatrix}.$$

The perturbation matrix $\Delta(q) = \text{diag}[q_1, q_2, q_3]$ with $-1 \le q_i \le 1$. Figure 1 shows the convergence of upper and lower bounds with iterations. At the end of 9000 iterations, the algorithm yields $\mathcal{C}(\mathcal{Q}_{\rm init}) = 7.36$ to within a relative accuracy of about 8%.

References

[1] V. Balakrishnan and S. Boyd. Global optimization in control system analysis and design. In C.T. Leondes, editor, *Advances in Control Systems*. Academic Press, New York, New York, 1992.

[2] V. Balakrishnan, S. Boyd, and S. Balemi. Branch and bound algorithm for computing the minimum stability degree of parameter-dependent linear systems. *To appear, International Journal of Robust and Nonlinear Control,* 1992.

[3] R. A. Horn and C. A. Johnson. *Matrix Analysis.* Cambridge University Press, 1985.

[4] P. M. Mäkilä and H. T. Toivonen. Computational methods for parametric LQ minimization: A survey. *IEEE Trans. Aut. Control*, AC-32(8):658–671, August 1987.

[5] J.-H. Xu, R. E. Skelton, and G. Zhu. Upper and lower covariance bounds for perturbed linear systems. *Int. J. Control*, 35(8):944–948, August 1990.