On the robustness analysis of nonlinear systems subject to time-invariant and/or time-varying uncertainty

Declan G. Bates, Sajjad Fekri, Prathyush P. Menon and Ian Postlethwaite

Control & Instrumentation Research Group
Department of Engineering, University of Leicester,
University Road, Leicester LE1 7RH, UK.
E-mails: {dgb3,sf111,ppm6,ixp}@le.ac.uk
Tel: +44-116-2522567, Fax: +44-116-2522619.

Abstract—We propose a novel approach for computing lower bounds on robust stability or performance for nonlinear systems subject to time-invariant and/or time-varying uncertainty. The approach exploits the binary coding mechanism used in defining chromosomes for evolutionary search algorithms, in order to allow the uncertain parameter space to be searched over time, thus allowing the computation of destabilising time-varying uncertain parameters. The resulting lower bounds can be used to check the conservatism of upper bounds computed using tools such as the Popov Criterion, IQC’s and various extensions of the structured singular value theory. The usefulness of the proposed approach is illustrated via an example.

Index Terms—Robustness analysis, time-varying uncertainties, Popov criterion, optimisation, Genetic algorithms

I. INTRODUCTION

Upper bounds on stability and performance robustness for linear or nonlinear plants subject to time-invariant and/or time-varying uncertainty can now be generated relatively easily using a number of different analytical tools. For linear systems, the extension of the structured singular value theory described in [1] allows upper bounds on allowable levels of LTI/LTV uncertainty to be calculated using Linear Matrix Inequalities (LMI’s). For nonlinear systems, a standard approach is to "cover" the system’s nonlinear elements with nonlinear uncertain parameters. Linear Fractional Transformation-based modelling approaches, [2], [3], may then be used to extract these parameters along with the other actual uncertain parameters in the system into a structured uncertainty matrix $\Delta$, which may contain LTI, time-varying, and nonlinear elements, and is connected to the known linear part of the system as shown in Figure 1. Once the system is in this form, various tools such as the Popov Criterion, [4]–[6], Integral Quadratic Constraints (IQC’s), [7], nonlinear generalisations of $\mu$, [8], [9], etc. may be used to compute reliable, but often highly conservative, upper bounds on robust stability and performance.

A major limitation of all of the above approaches to robustness analysis is that no systematic methods are available for computing the corresponding lower bounds on robust stability and performance. That is, for either linear or nonlinear systems subject to time-varying uncertainty, there are no equivalents to the algorithms for computing lower bounds on $\mu$ which exist in the standard structured singular value theory for LTI systems subject to LTI uncertainty. This is a problem for several reasons. Firstly, without lower bounds it is impossible to estimate the level of conservatism in the upper bound results. Secondly, the computation of worst-case uncertainties can often shed light on the way in which uncertainty effects the system, and can hence provide insight which may be used in re-designs to improve the performance of the controller. Thirdly, the focus on upper bounds (i.e. the search for mathematical proofs of stability/performance robustness using often very simplified mathematical models) is far removed from much of industrial control engineering practice, which relies mainly on exhaustive simulation with complex high-fidelity models coupled with statistical/grididding methods to estimate lower bounds on the worst-case behaviour of the system, [10].

In this paper, we attempt to address some of the above issues by proposing a computational method for calculating lower bounds on worst-case stability or performance for nonlinear systems. The proposed approach is in the same spirit as the methods described in [11]–[13]. The main new contribution is that it allows worst-case time-varying uncertainties for nonlinear systems to be computed, by exploiting the particular binary coding mechanism used in defining chromosomes for evolutionary search algorithms. The resulting worst-case uncertainty combinations provide lower bounds on stability/performance robustness, which may then be used to estimate the level of conservatism in robustness analysis.

This work was supported by EPSRC Research Grant No. GR/S61874/01

Fig. 1. Known LTI system $G$ in a feedback interconnection with an uncertain LTI/LTV/NL system.
the corresponding upper bounds.

The paper is organised as follows. In Section II we describe the example nonlinear robustness analysis problem which will be used to illustrate the proposed approach, and show how the system may be represented in the form of Figure 1. Upper bounds on robust stability for this system are computed using the Popov Criterion in Section III. In Section IV, the conservatism of these bounds is estimated by computing lower bounds on robust stability using the proposed approach. Finally, Section V presents some conclusions.

II. EXAMPLE NONLINEAR ROBUSTNESS ANALYSIS PROBLEM

We consider a linear plant with a nonlinear saturation at the plant input and a time-delay at the output. The state-space description of the plant, which is similar to the one considered in [7], is given by

\begin{align}
\dot{x}(t) &= Ax(t) + Bsat(u(t)) \\
u(t) &= -kCz(t - \theta(t))
\end{align}

where \( \theta \) is the time-delay, \( sat(u(t)) \) represents the saturation as a sector bounded nonlinearity and \( k \) is the scalar feedback controller gain. Uncertainty is introduced into this system by two means. Firstly, the magnitude of the feedback controller gain. Uncertainty is introduced into the plant input and a time-delay at the output. The state-space description of the plant, which is similar to the one given level of time-varying uncertainty on \( \alpha \) for which the system defined above remains stable.

To address this question, we first of all show how the system of Figure 2 may be recast in the form of Figure 1 using LFT-based uncertainty modelling. Once the system is in this form, upper bounds on the allowable value of \( k \) for various levels of \( \theta_0 \) will be computed using the Popov criterion.

In order to model the uncertain time delay in an LFT framework, we represent it as a multiplicative error:

\[ e_M(s) = e^{-\theta s} - 1 \]

as shown in Fig. 3. The multiplicative error magnitude can then be approximated by a high-pass transfer function \( W_{un}(s) \) with a pole placed near the frequency \( \beta = \pi/\theta \). The magnitude and phase of the transfer function \( W_{un}(s) \) is shaped by a delta block \( \Delta_\theta(s) \) which introduces a phase uncertainty in the range of \( \pm 180 \).
we assume that the saturation is decoupled, sector-bounded and static with a constant saturation limit of $u_M$. The input nonlinearity can then be transformed into a saturation parameter $\delta_{\text{sat}}$:

$$\delta_{\text{sat}} = \frac{\text{sat}(u)}{u}$$

with $\delta_{\text{sat}}(0) = 1$ for well-posedness. The nonlinear uncertain parameter $\delta_{\text{sat}}$ is a normalization of the input saturation by the unsaturated signal as shown in Fig. 6, which can be "pulled out" and placed in the $\Delta$ uncertainty matrix of Figure 1.

Finally, standard LFT modelling procedures can be used to pull out the the time-varying uncertainty parameter $\alpha$ and place it in the uncertainty matrix $\Delta$ of Figure 1. This diagonal matrix $\Delta = \text{diag}\{\Delta_1, \Delta_2, \delta_{\text{sat}}\}$ now contains three terms - a complex LTI uncertainty representing the uncertain time-delay, a time-varying real uncertainty representing $\alpha$, and a nonlinear uncertainty parameter which is used to "cover" the saturation nonlinearity in the original system.

Upper bounds on the allowable values of controller gain $k$ for this system can now be computed using the Popov Criterion, as described in the next section.

### III. UPPER BOUNDS ON ROBUST STABILITY

The Popov criterion provides a sufficient condition for the robust stability of the system interconnection shown in Fig. 7 where $G(s)$ is a known LTI system and $\phi = \text{diag}\{\phi_1, \ldots, \phi_n\}$ is a sector-bounded BIBO-stable uncertainty satisfying $\phi(0) = 0$. Note that the operator $\phi$ can be nonlinear and/or time-varying. The behavior of uncertain dynamical components can also be quantified in terms of sector bounds on their time-domain response [14]. A (possibly nonlinear) BIBO-stable system $\phi(.)$ is said to be in the sector $\{a, b\}$ if

$$au^2 \leq \phi(u)u \leq bu^2$$

An example of such sector bounds in the scalar case is shown in Figure 8. In vector form, with $a = [a_j], b = [b_j]$,

and $u = [u_1, \ldots, u_m]$, $\phi(.)$ is said to be in the sector $\{a, b\}$ if the mapping $\phi$ satisfies the quadratic constraint of the form [15]

$$\int_0^T (y(t) - au(t))^T (y(t) - bu(t))dt \leq 0, \forall u \in L_2$$

A state-space representation of this interconnection is given by

$$\begin{align*}
\dot{x} &= Ax + Bw \\
q &= Cx + Dw \\
w &= \phi(q)
\end{align*}$$

or equivalently in block-partitioned form:

$$\begin{align*}
\dot{x} &= Ax + \sum_j B_j w_j \\
q_j &= C_j x + D_j w_j \\
w &= \phi(q_j)
\end{align*}$$

Suppose that $\phi_j(.)$ satisfies the sector bound

$$\forall T > 0, \int_0^T (w_j - \alpha_j q_j)(w_j - \beta_j q_j)dt \leq 0$$

Note that this includes norm bounds as the special case $\beta_j = -\alpha_j$. To establish robust stability and assuming
nominal stability, the Popov criterion seeks a Lyapunov function $V(x, t)$ such that the conditions $V(x, t) > 0$ and $dV/dt < 0$ are satisfied. This problem can be cast as an LMI feasibility problem, and hence solved using standard software. The Popov criterion can also be used to analyze systems with uncertain real parameters, for which a more general form for the corresponding term in the Lyapunov function is found. The resulting test is discussed in detail in [14], [16], and is coded in the command (popov) in the MATLAB Robust Control Toolbox.

The Popov Criterion was used to analyse the robustness of the block diagram of Fig. 9. Recall that the pole uncertainty is real and time-varying, the saturation uncertainty parameter is nonlinear and the time-delay uncertainty is modeled as a complex LTI uncertainty. The maximum allowable values of $k$ for which the Popov Criterion guarantees closed loop stability are shown for different values of $\theta_0$ by the blue dashed line in Figure 10. For the purposes of comparison, the maximum values of $k$ were also computed assuming that the pole uncertainty was now time-invariant - these values are shown by the red line. According to these results, allowing the pole uncertainty to be time-varying as opposed to time-invariant significantly reduces the level of controller gain which may be used - for a maximum time delay of $\theta_0 = 0.5$ secs, for example, the maximum stabilizing $k$ is reduced from 1.596 to 0.9794. In the next section, we apply the proposed new approach for computing lower bounds on robust stability to this problem, with the aim of evaluating the level of conservatism in the above results.

IV. LOWER BOUNDS ON ROBUST STABILITY

According to the upper bounds on robust stability computed in the previous section, for a maximum time-delay uncertainty of $\theta_0 = 0.5$, the closed-loop system is guaranteed to be stable only for values of $k$ less than or equal to 0.9794. To test the conservatism of this result, we first set $k$ equal to a much larger value of 4.5 and attempt to find a combination of uncertain parameters that destabilises the system.

In the system under consideration, the controller acts as a regulator. Figure 11 shows the nominal response of the closed-loop system to a step disturbance signal. To check whether there exists a destabilising uncertainty combination for $k$ equal to 4.5, we formulate a time-domain optimisation problem, as follows: $J = \max \int_0^T y(t)y(t)' \ dt$ (4.1)

where $T$ represents some finite simulation time, and the input is a step disturbance signal. Since this function represents the finite time energy associated with the output signal, maximisation of this function corresponds to driving the system towards instability. The optimisation variables are $\delta_\alpha(t)$ and $\Delta\theta$ - note that $\delta_{sat}$ is not included in the lower bound computations since it is not an actual uncertainty in the system (rather, the actual nonlinearity itself is included in the system used for simulations). Also, rather than working with $\Delta\theta$, we include the actual time-delay in our simulation model and vary its value directly. To cope with the time-varying nature of $\delta_\alpha(t)$, it is assumed that the value of the uncertainty does not change continuously, but only at $N$ defined time intervals over the course of the simulation. During each time interval, $\delta_\alpha$ is assumed to be constant. This assumption reduces the infinite number of candidate solutions for the optimisation problem to a finite number, at the expense of reducing the tightness of the lower bound optimisation problem, as follows:

$$J = \max \int_0^T y(t)y(t)' \ dt$$
that can be computed.

To solve the above optimisation problem, which is clearly nonlinear and nonconvex in general, we use an optimization approach based on Genetic Algorithm’s (GAs), which are general purpose stochastic search and optimization procedures that use genetic and evolutionary principles [17]. A recent survey by Fleming and Purshouse in [18] provides an overview of applications of GA’s in the field of control engineering. The basic principle underlying GA’s is the assumption that the evolutionary process observed in nature can be simulated on a computer to generate a population of fittest candidate solutions for a given problem. In genetic search techniques, a randomly sourced population of candidates undergoes a repetitive evolutionary process of reproduction through selection for mating according to a fitness function, and recombination via crossover with mutation. A complete repetitive sequence of these genetic operations is called a generation. To use this evolutionary method, it is necessary to have a method of encoding each candidate solution as an artificial chromosome, as well as a means of discriminating between the fitness of different candidates. Each optimization variable, or gene, is binary coded according to the required accuracy level and combined sequentially to form the chromosome, which represents a potential candidate solution. In the case of the time-varying parameter \( \delta_\alpha(t) \), a discretisation into intervals of 5 seconds is used, following the assumptions described above. Then, at each discrete interval, the parameter is encoded using 4 bits to allow \( 2^4 = 16 \) different discrete amplitude levels between its uncertainty bounds of \([0.2, 1.8]\). Hence for a simulation time of \( T = 30 \) seconds, \( 6 \times 4 = 24 \) bits are required in the chromosome to represent the time-varying uncertainty. Table I shows the uncertainty intervals used for the time-varying parameter and the corresponding binary representation. For the LTI delay uncertainty, a 20 bit encoding is used which allows its worst case value to be computed with an accuracy of \( 1e-6 \). The above representation for the uncertain parameters results in a 44 bit chromosome for the GA, consisting of 24 bits for the LTV uncertainty information and 20 bits for the LTI delay uncertainty.

The search starts from an initial population consisting of a fixed number of randomly selected candidates. The size of this initial population is given by \( N_{\text{size}} = 30 \) in the present study. The candidates from the current generation are qualified to produce the successive generations depending on a selection scheme. A normalised geometric distribution selection scheme with a selection probability of 0.6 is applied in this study. During crossover, a recombination operator ensures mixing up of the information content between two different binary coded chromosomes. A single point crossover with a probability of crossover 0.6 is used here. The point of crossover is determined randomly over the length of bits. Mutation introduces random variations in the population in the search space, by randomly flipping a bit value. The probability of mutation is kept low and fixed at 0.05. The number of maximum generations is the termination criterion and is fixed at 150 generations. The reader is referred to [17] for more details of GA operators, binary coding schemes and the theory of genetic search.

<table>
<thead>
<tr>
<th>Table I</th>
<th>LTV UNCERTAINTY BINARY REPRESENTATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \delta_\alpha(t) )</td>
<td>Binary Levels</td>
</tr>
<tr>
<td>0.2</td>
<td>0 0 0 0</td>
</tr>
<tr>
<td>0.3067</td>
<td>0 0 1 1</td>
</tr>
<tr>
<td>0.4133</td>
<td>0 1 1 0</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>1.5867</td>
<td>1 1 0 1</td>
</tr>
<tr>
<td>1.6933</td>
<td>1 1 1 0</td>
</tr>
<tr>
<td>1.8000</td>
<td>1 1 1 1</td>
</tr>
</tbody>
</table>

Figure 12 shows the results from the optimisation-based analysis for the controller feedback gain \( k = -4.5 \). The figure clearly shows the destabilising effect of the computed worst-case uncertainty combination - the output does not diverge to infinity due to the presence of the saturation in the feedback loop but instead settles into a sustained oscillation. Note also that the optimisation finds the worst-case value for \( \delta_\alpha(t) \) to be time-varying over the course of the simulation. The evolution of the worst-case value of \( \delta_\alpha(t) \) over successive generations of the optimisation algorithm is shown in Figure 13. From the figure we can see how the optimisation algorithm changes the time-varying nature of \( \delta_\alpha(t) \) in order to maximise the energy of the output signal. To further test the conservatism of the upper bound result, we now iteratively lower the value of \( k \) and repeat the analysis until we reach a value for which no destabilising uncertainty combination can be found. For the present example, it turns out that the smallest value of \( k \) for which the optimisation returns a destabilising uncertainty combination is 4, as shown in Figure 14. If we further reduce the value of \( k \) to, say 2.5, we see from the results in Figure 15 that no destabilising uncertainty combination could be found by the proposed approach. This, of course,
does not mean that no destabilising uncertainty combination exists for this value of \( k \), since our various assumptions on the nature of the time-varying uncertainty, together with the fact that we are only considering one particular (step) disturbance signal, mean that there is always the possibility that the proposed approach will fail to find such uncertainties even when they do exist. This, however, is not our goal, since ruling out the existence of de-stabilising uncertainties is an upper bound problem. What we can say for sure, however, is that (i) no destabilising uncertainties exist for \( k \leq 0.9794 \) (from our upper bound result), and (i) at least one time-varying destabilising uncertainty does exist for \( k \geq 4 \) (from our lower bound result). This gives a reasonable estimate of the level of conservatism in the upper bound result. To further tighten the lower bound, at the expense of increasing the computation time, one can simply make the gridding for the time-varying parameter more fine, e.g. allow it to vary every 1 second as opposed to 5 seconds, investigate other types of input signals and/or increase the population size or maximum number of generations in the genetic algorithm.

V. Conclusions

We have presented a novel approach for computing lower bounds on robust stability or performance for nonlinear systems subject to time-invariant and/or time-varying uncertainty. The approach exploits the binary coding mechanism used in defining chromosomes for evolutionary search algorithms, in order to allow the uncertain parameter space to be searched over time, thus allowing the computation of destabilising time-varying uncertain parameters. The resulting lower bounds can be used to check the conservatism of upper bounds computed using tools such as the Popov Criterion, IQC’s and various extensions of the structured singular value theory. The usefulness of the proposed approach was illustrated via an example.

References