6-4-1997

Robust multiobjective feedback design via combined quantifier elimination and discretization

Chaouki T. Abdallah

Wei Yang

Peter Dorato

Recommended Citation

Abdallah, Chaouki T.; Wei Yang; and Peter Dorato. "Robust multiobjective feedback design via combined quantifier elimination and discretization." *Proceedings of the 1997 American Control Conference* (1997): 1843-1847. doi:10.1109/ACC.1997.610904.

Proceedings of the American Control Conference
Albuquerque, New Mexico June 1997

Robust Multiobjective Feedback Design via Combined Quantifier Elimination and Discretization

Wei Yang†, Peter Dorato†, and Chaouki T. Abdallah†
†Department of Electrical and Computer Engineering
University of New Mexico
Albuquerque, NM 87131-1356
wyang@eece.unm.edu

Abstract

This paper is concerned with the application of computerized quantifier elimination (QE) methods for robust multiobjective feedback design (RMOD), when design objectives are specified in the frequency domain. The class of design problems considered here has no *analytical solutions*, so that computerized solutions are of interest, even for relatively simple problems. However because of the computational complexity of pure QE algorithms, a combined QE-discretization approach is proposed and illustrated with a single example.

1. Introduction

In this paper, we are going to deal with the RMOD problems in the frequency-domain. The problem can be stated as follows: given a plant with uncertain parameters each of which lies in a given range ($\mathcal{P}$), find a family of compensators which satisfy multi-objectives, such as stability, tracking error, and control effort, etc.. For simplicity, here we consider the linear, time-invariant, single-input single-output system and assume that the compensator structure is given. In this case, it has been shown (Fiorio, Malan, and Milanese, 1993; Dorato, Yang, and Abdallah, 1996) that many interesting RMOD problems can be stated mathematically as follows: find the range of $\mathbf{q}$ such that the set of inequalities,

$$F_i(\omega, \mathbf{p}, \mathbf{q}) > 0, \omega \in \Omega,\ \mathbf{p} \in \mathcal{P},\ i = 1, 2, ..., m \quad (1)$$

hold, where ω is the frequency variable within the given range Ω, $\mathbf{p}$ is the vector of uncertain parameters in the given plant within the given range $\mathcal{P}$, $\mathbf{q}$ is the vector of design parameters in compensation, and F_i are the multivariable polynomial functions. For quantifier elimination algorithms, the further assumption that the coefficients in F_i must be integers is required. Since real numbers can always be closely approximated by rational numbers, this is not a serious constraint. Currently there are four ways of dealing with this problem. They are:

(1) **pure-discretization approach**, discretize each component of uncertain plant parameter vector $\mathbf{p}$ and design parameter vector $\mathbf{q}$ (ω if necessary) and check if all F_i are satisfied for the discretized variables.

(2) **stochastic approach**, e.g. Monte-Carlo and genetic algorithms. The idea is the same as pure-discretization approach. But here the discretized points are picked in terms of some kind of probability.

(3) **overestimating approach**, overestimate the maximum value ($\bar{F}_i$) and minimum value ($\underline{F_i}$) of polynomials over given intervals (Fiorio, Malan, and Milanese, 1993);

(4) **pure QE approach**, use quantifier elimination (QE) algorithms (Collins, 1975; Collins and Hong, 1991) symbolically eliminated the quantified variables $\mathbf{p}$ and ω in inequalities (1) and produce equivalent quantifier-free formulas ($\Psi(\mathbf{q})$) on the domain of vector $\mathbf{q}$, which represents a characterization of the compensator design. A QE software package, called QEPCAD-Quantifier Elimination by Partial Cylindrical Algebraic Decomposition-QEPCAD (Hong, 1992), is available for solving control problems.

The key disadvantage for the first three approaches listed above is that one must have "a prior" knowledge for the design parameters $\mathbf{q}$, i.e. one must know (or assume) the range of design parameters $\mathbf{q}$. Otherwise, one has to search the whole $\mathbf{q}$ space, which is impossible in limited time. QE algorithms are very attractive for control problems where there are no general analytical design approaches, e.g. output-feedback stabilization problem. QE methods were applied to output-feedback stabilization problem by (Anderson, Bose, and Jury, 1975). The advantage of QE algorithm is that it can give a necessary and sufficient condition on $\mathbf{q}$ by searching the whole $\mathbf{q}$ space. However QE algorithms, even recent ones, are very complex (Collins, 1975; Collins and Hong, 1991). Computing time and storage space complexity are double exponential in the number of variables and expressions. Due to this, a new method, called QE-discretization approach, is proposed here, which combines the QE algorithm and discretization methods. It turns out that this QE-discretization approach can solve some control problems, e.g. example 2, where QEPCAD fails to produce an output.

This paper is organized below. Section 2 reviews some QE theory and software. Section 3 presents the reduction of feedback design problems to QE problems. Section 4 introduces the algorithms of directly using QEPCAD software package to solve the control problems. Section 5 proposes the QE-discretization algorithm, while section 6 gives two examples. The first example is very simple control problem, which explains the algorithms in section 4. The second example is a little more complicated, in which we show the difficulties of directly using QEPCAD. But it turns out that the QE-discretization approach proposed here can solve this problem.

2. QE Algorithms and Software

In this section, we review the general QE problem and introduce the software package QEPCAD which we use to solve our control problems. A more detailed treatment may be found in (Tarski, 1951; Basu, Pollack, and Roy, 1994).

Given the set of polynomials with integer coefficients $P_i(X,Y)$, $1 \leq i \leq s$ where X represents a k dimensional vector of quantified real variables and Y represents a l dimensional vector of unquantified real variables, let $X^{[i]}$ be a block of k_i quantified variables, Q_i be one of the quantifiers $\exists$ (there exists) or $\forall$ (for all), and let $\Phi(Y)$ be the quantified formula

$$\Phi(Y) = (Q_1 X^{[1]}, ..., Q_w X^{[w]}) F(P_1, ..., P_s), \qquad (2)$$

where $F(P_1, ..., P_s)$ is a quantifier free *Boolean* formula, that is a formula containing the Boolean operators $\wedge$ (and) and $\vee$ (or), operating on *atomic predicates* of the form $P_i(Y, X^{[1]}, ..., X^{[w]}) \geq 0$ or $P_i(Y, X^{[1]}, ..., X^{[w]}) > 0$ or $P_i(Y, X^{[1]}, ..., X^{[w]}) = 0$. We can now state the general quantifier elimination problem

General Quantifier Elimination Problem: Find a quantifier-free Boolean formula $\Psi(Y)$ such that $\Phi(Y)$ is true if and only if $\Psi(Y)$ is true.

In control problems, the unquantified variables are generally the compensator parameters, represented by the parameter vector $Y = \mathbf{q}$, and the quantified variables are the plant parameters, represented by the plant parameter vector $\mathbf{p}$, and the frequency variable ω. Uncertainty in plant parameters are characterized by quantified formulas of the type $\forall(p_i)\ [\underline{p_i} \leq p_i \leq \overline{p_i}]$ where $\underline{p_i}$ and $\overline{p_i}$ are rational numbers. *The quantifier-free formula $\Psi(\mathbf{q})$ then represents a characterization of the compensator design.*

An important special problem is the QE problem with no unquantified variables (free variables), i.e. $l = 0$. This problem is referred to as the *General Decision Problem.*

General Decision Problem: With no unquantified variables, i.e. $l = 0$, determine if the quantified formula given in (2) is true or false.

The general decision problem may be applied to the problem of *existence* of compensators that meet given specifications, in which case an "existence" quantifier is applied to the compensator parameter $\mathbf{q}$. Algorithms for solving general QE problems were first given by (Tarski, 1951; and Seidenberg, 1954), and are commonly called Seidenberg-Tarski decision procedures. Tarski showed that QE is solvable in a finite number of "algebraic" steps, but his algorithm and later modifications are exponential in the size of the problem. Researchers in control theory have been aware of Tarski's results and their applicability to control problems since the 1970's (Anderson, Bose, and Jury, 1975), but the complexity of the computations and lack of software limited their applicability. Later, (Collins, 1975) introduced a theoretically more efficient QE algorithm that uses a cylindrical algebraic decomposition (CAD) approach. However, this algorithm was not capable of effectively handling non-trivial problems. More recently (Hong, 1990; Collins and Hong, 1991; Hong, 1992) have introduced a significantly more efficient partial CAD QE algorithm.

The Cylindrical Algebraic Decomposition (CAD) algorithm, has been developed (Collins and Hong, 1991) for the computer elimination of quantifiers on polynomial-function inequalities. This algorithm requires a *finite number* of "algebraic" operations. However the number of operations is still doubly exponential in the number of variables, so that only problems of modest complexity can actually be computed. See (Basu, Pollack, and Roy, 1994) for a discussion of computational complexity in quantifier elimination. A software package called QEPCAD (Quantifier Elimination by Partial Cylindrical Algebraic Decomposition) has been developed for the solution of quantifier elimination problems (H. Hong, Institute for Symbolic Computation, Linz, Austria). An excellent introduction to quantifier elimination theory and its applications to control system design may be found in the monograph of (Jirstrand, 1996).

3. Reduction to a Quantifier Elimination Problem

From the discussions above it follows that frequency-domain robust multiobjective feedback design problem can be reduced to the satisfaction of inequality constraints of the form given in (1) with logic quantifiers of the form 'for all ω" and "for all $\mathbf{p}$" over given ranges of ω and $\mathbf{p}$. Typically the variables in the polynomials are real and are related to plant (controlled system) and compensator (controller) parameters. The final design objective is to obtain quantifier-free formulas for the compensator parameters or, for the existence problem, to obtain a "true" or "false" output. For example, given a plant transfer function $G(s,\mathbf{p})$, and a controller with transfer function $C(s,\mathbf{q})$, the requirement that the transfer function between reference input and control input be constrained to have a magnitude less than a given value, α_U, may be written

$$\left|\frac{C(j\omega,\mathbf{q})}{1+C(j\omega,\mathbf{q})G(j\omega,\mathbf{p})}\right| < \alpha_U \tag{3}$$

By squaring the magnitude and clearing fractions, the expression above takes on the form $F_1(\omega,\mathbf{p},\mathbf{q}) > 0$, where the function F_1 is polynomial in its arguments as long as the components of the vector $\mathbf{p}$ and $\mathbf{q}$ enter the coefficients of the polynomials in the transfer function G and C polynomially. Similarly tracking error can be reduced to an inequality of the form $F_2(\omega,\mathbf{p},\mathbf{q}) > 0$. Finally stability of the closed-loop system is guaranteed, via the Routh Hurwitz test, by the satisfaction of further inequalities of the form $F_i(\mathbf{p},\mathbf{q}) > 0$.

4. Algorithms for solving RMOD problems via QE theory

In this section, we discuss the ways of directly using QEPCAD to solve RMOD problems. Particularly, two algorithms are given here, which are based on the observations: (i) $\mathbf{p}$ and ω are quantified variables; (ii) $\mathbf{q}$ are free (unquantified) variables. The task of the algorithms for RMOD is to eliminate $\mathbf{p}$ and ω to obtain the solutions on $\mathbf{q}$, i.e. *quantifier-free formula* on $\mathbf{q}$.

Algorithm 1

step 1: Use QE formula,

$$(\exists\mathbf{q})(\forall\mathbf{p}\in\mathcal{P})(\forall\omega\in\Omega)[F_i > 0,\ \forall i], \tag{4}$$

to determine if a solution exists.

step 2: If "yes", use the QE formula,

$$(\forall\mathbf{p}\in\mathcal{P})(\forall\omega\in\Omega)[F_i > 0,\ \forall i], \tag{5}$$

to obtain a quantifier-free formula $\Psi(\mathbf{q})$ which is then used to determine a set of admissible vector values.

Comments: If there are more than one design parameter, the given *quantifier-free formula* is rather complicated such that one cannot figure out the solution regions (see example 1). So, the following algorithm is proposed, by which the solutions can cleanly be displayed in figures.

Algorithm 2

step 1: The same as that in algorithm 1;

step 2: If "yes", use the QE formula,

$$(\exists q_k)(\forall\omega\in\Omega)(\forall\mathbf{p}\in\mathcal{P})[F_i > 0,\ \forall i], \tag{6}$$

for all k, except $k = j$, to obtain a quantifier-free formula in the single unquantified variable q_j. This formula in q_j involves only polynomials in one variable, for which inequality can easily be checked by finding the roots of the respective polynomials. In this way one can compute the intervals that include admissible values of q_j. This can be repeated for other components of $\mathbf{q}$ to obtain intervals for each component of the design vector $\mathbf{q}$, defining boxes in which admissible variables may lie.

step 3: Within the boxes obtained above, use QE formula,

$$(\forall\mathbf{q}\in\mathcal{Q})(\forall\omega\in\Omega)(\forall\mathbf{p}\in\mathcal{P})[F_i > 0,\ \forall i], \tag{7}$$

to check if some interesting point or subbox, denoted $\mathcal{Q}$, is a solution.

5. QE-discretization Approach

Although the QE theory discussed above appears very attractive for robust multiobjective design, it breaks down computationally even for very simple control problem (see example 2 in next section). Since the complexity of QE algorithm is double exponential in the number of variables and expressions, the successful application of the QE algorithm requires one to reduce the number of variables and number of expressions as much as possibly. QE discretization is one way to reduce the number of variables, and is described next.

step 1: Pick a particular value of $\mathbf{p}$, which is generally the nominal value $\mathbf{p_0}$ of $\mathbf{p}$;

step 2: The QE formula,

$$(\forall\omega \in \Omega)[F_i(\omega, \mathbf{p}_0, \mathbf{q}) > 0,\ \forall i], \qquad (8)$$

is used to determine the range $R_{\mathbf{q}}^{\mathbf{p_0}}$ of design parameters $\mathbf{q}$ for $\mathbf{p} = \mathbf{p}_0$. It should be pointed out that the space $R_{\mathbf{q}}^{\mathbf{p_0}} \supset R_{\mathbf{q}}^{\mathbf{p}}$, i.e. $R_{\mathbf{q}}^{\mathbf{p_0}}$ is a necessary solution region, where $R_{\mathbf{q}}^{\mathbf{p}}$ denotes the solution regions for all $\mathbf{p} \in \mathcal{P}$;

step 3: Discretize the space $R_{\mathbf{q}}^{\mathbf{p_0}}$. Then the QE formula,

$$(\forall\omega \in \Omega)(\forall \mathbf{p} \in \mathcal{P})[F_i(\omega, \mathbf{p}, \mathbf{q}_0) > 0,\ \forall i], \qquad (9)$$

is used to determine if some discretized point $\mathbf{q_0}$ is a solution point. After we search the $R_{\mathbf{q}}^{\mathbf{p_0}}$, we can find a solution region in the $\mathbf{q}$ space.

Comments:

- Note that this approach is different from the pure discretization approach mentioned in the introduction. In particular, in this approach, only the design parameters $\mathbf{q}$ are discretized and QE takes care of the plant parameters $\mathbf{p}$ and ω, while, in the pure discretization approach, ω, $\mathbf{q}$, and $\mathbf{p}$ must <u>all</u> be discretized.
- In the step 2 for determining $R_{\mathbf{q}}^{\mathbf{p_0}}$, the idea of algorithm 2 (step 2) should be used if the number of design parameters is more than one.
- In the step 3, some stochastic idea (e.g. Monte-Carlo) can be used to choose the discretized points instead of fixed-discretization.
- For the step 3 of the QE-discretization approach given above, one may suggest that subdivision strategy be used, i.e. check if each subbox $\mathcal{Q}$ subdivided within $R_{\mathbf{q}}^{\mathbf{p_0}}$ is a solution region by using QE formula,

$$(\forall\omega \in \Omega)(\forall \mathbf{p} \in \mathcal{P})(\forall \mathbf{q} \in \mathcal{Q})[F_i(\omega, \mathbf{p}, \mathbf{q}) > 0,\ \forall i]. \qquad (10)$$

It should be noticed that doing so is generally inappropriate since the number of variables is increased.

6. Examples

In this section, two examples are given to illustrate the application of QE theory, algorithm 1 & 2 (example 1) and mixed QE-discretization approach (example 2).

Example 1: (Dorato, Yang, and Abdallah, 1996). The plant is $G(s, p_1) = \frac{1}{s+p_1}$, $p_1 = \pm 1$. The problem is to find a optimal PI compensator ($C(s, q_1, q_2) = q_1 + q_2/s$) such that we have

- robust stability;
- steady-state tracking error;
- as small as possible control effort (i.e. small value of $\alpha_U = n/d$, where n and d are integers).

Solutions: First we formulate the problem into a system of Boolean formulas, which are $F_1(\omega, p_1, q_1, q_2)$, $F_2(\omega, p_1, q_1, q_2)$, and $F_3(\omega, p_1, q_1, q_2)$ (omitted here due to limited space). Then algorithm 2 is used, which produces the step-by-step results shown below:

step 1: The answer to the existence question is "yes" with "minimum" control effort $n^*/d^* = 41/10$ in the sense that the answer is "no" for $n/d = 40/10$;

step 2: $1.9758 \le q_1 \le 2.0248$ and $0 < q_2 \le 0.0253$;

step 3: Since this problem is an optimization problem in the sense that the control effort was made as small as possibly, the compensator set is basically shrinker to single point. To find optimal compensator in the sense that control effort is as small as possible, pick the mean values of q_1 and q_2, which are $q_1^* = 2$ and $q_2^* = 0.0126$. QE formula $(\forall\omega \in [0, \infty])(\forall p_1 = \pm 1)[F_1^{(q_1^*, q_2^*)}(\omega, p_1) > 0 \wedge F_2^{(q_1^*, q_2^*)}(\omega, p_1) > 0 \wedge F_3^{(q_1^*, q_2^*)}(\omega, p_1) > 0]$ was put to check if $(q_1^*,\ q_2^*)$ is a solution. It turns out that "true" was returned, which means that optimal PI compensator is $C^*(s) = 2 + 0.0126/s$.

Note: If algorithm 1 is used, then it turns out that, after step 2, QEPCAD gives

$$\begin{aligned}&[961q_1^4 - 5084q_1^3 + 620q_2^2q_1^2 - 5084q_2q_1^2 + 9266q_1^2\\ &\quad 1640q_2^2q_1 + 13448q_2q_1 - 6724q_1 + 100q_2^2\\ &\quad 1640q_2^3 + 820q_2^2 - 6724q_2 + 1681 \le 0 \vee\\ &(10q_1^2 - 41 \le 0 \wedge 31q_1^2 - 82q_1 - 10q_2^2 - 82q_2\\ &\quad +41 \ge 0)]\\ &\wedge q_1 - 1 > 0 \wedge q_2 > 0 \wedge q_1 + 1 > 0.\end{aligned}$$

We can see that this quantifier-free formula is rather complicated.

Example 2: (Fiorio, Malan, and Milanese, 1993), where the robust tuning of a proportional plus integral compensator is considered. The plant is $G(s, p_1, p_2) = \frac{p_1}{1 - s/p_2}$, where $0.8 \le p_{1,2} \le 1.25$. The controller is a PI compensator $C(s, q_1, q_2) = \frac{q_1(1+s/q_2)}{s}$. The design aim is to find a family of compensators which satisfy the following performance measures:

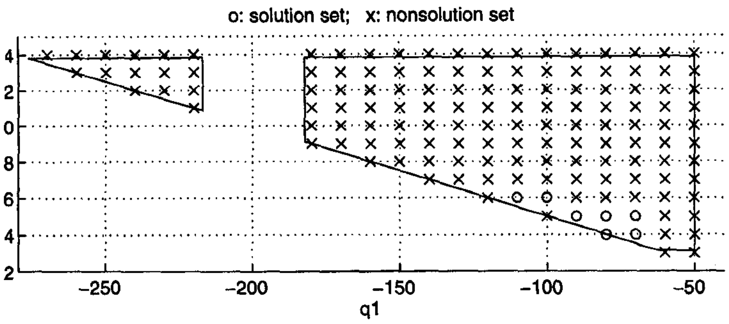

Figure 1: Admissible design parameter values for example 2.

- robust stability;
- unitary ramp steady state error $|e_r| < 0.02s$;
- bandwidth $\omega_b \geq 10rad/s$;
- resonance peak of the complementary sensitivity function $T_p < 1.4$;
- control effort $|R(j\omega)| < 20, \forall \omega \in [0, \infty]$;

Solutions: In this example, the QE-discretization approach was used. The results are:

step 1: pick $\mathbf{p_0} = [p_1 \ \ p_2] = [1 \ \ 1]$;

step 2: $R_{q_1}^{\mathbf{p_0}} = [(q_1 < -217) \vee (-182 < q_1 < -50)]$ and $R_{q_2}^{\mathbf{p_0}} = [3.13 < q_2 < 13.83]$. The necessary solution region $R_{\mathbf{q}}^{\mathbf{p_0}}$ is shown in figure 1, which is bounded by solid lines;

step 3: Discretize the $R_{\mathbf{q}}^{\mathbf{p_0}}$. Also see figure 1 for discretized points. Then apply QEPCAD for each of these points. We obtain the solution regions which is shown in figure 1, where symbol 'o' show the solution points and symbol "x" show the non solution points.

Note: If either algorithm 1 or algorithm 2 is used, then QEPCAD gives the message below.

```
qe16M: 1418 Memory fault - core dumped
      17156.8 real  16496.9 user  156.7 sys
16496.9u 156.7s 4:45:56 97% 0+-4416k
7+1975io 44pf+0w
```

From this we see that QEPCAD alone cannot solve even relatively simple problems.

7. Conclusions

We have explored the relations between QE and RMOD, from which it can be seen that QE theory fits control problems, especially for the problems where there are no analytical design approaches, quite well. Two algorithms for RMOD are given in this paper based on the QE theory. Because of high computational cost of QE algorithm, which greatly limits its application, a new method, called QE-discretization approach, is proposed. It turns out that this new approach for RMOD can solve some interesting problems, e.g. example 2 in this paper, for which QEPCAD alone breaks down.

References

[1] B.D.O. Anderson, N.K. Bose, and E.I. Jury, Output Feedback Stabilization and Related Problems-Solutions via Decision Methods. *IEEE Transactions on Automatic Control.* AC-20, 53-65, 1975.

[2] S. Basu, R. Pollack, and M.F. Roy, *On the combinatorial and algebraic complexity of quantifier elimination,* Proc. 35th Symposium on Foundations of Computer Science, Santa Fe, NM, pp. 632-641, 1994.

[3] G.E. Collins, Quantifier Elimination in the Elementary Theory of Real Closed Fields by Cylindrical Algebraic Decomposition. *Lecture Notes in Computer Science.* Santa Fe, NM, 632-641, 1975.

[4] G.E. Collins and H. Hong, Partial Cylindrical Algebraic Decomposition for Quantifier Elimination. *Journal of Symbolic Computation.* 12, 299-328, 1991.

[5] P. Dorato, W. Yang, and C.T. Abdallah, Robust Multi-Objective Feedback Design by Quantifier Elimination. *Journal of Symbol Computation* Accepted for publication, 1996.

[6] H. Hong, *Improvements in CAD-based Quantifier Elimination,* Ph.D Thesis, The Ohio State University, 1990.

[7] H. Hong, Simple Solutions Formula Construction in Cylindrical Algebraic Decomposition based Quantifier Elimination. *International Symposium on Symbolic and Algebraic Computation.* Berkeley, CA, 1992.

[8] G. Fiorio, S. Malan, M. Milanese, and M. Taragna, Robust Performance Design of Fixed Structure Controllers with Uncertain Parameters. *32nd IEEE Conf. Decision and Control,* San Antonio, TX, 3029-3031, 1993.

[9] M. Jirstrand, *Algebraic methods for modeling and design in control,* Linkõping Studies in Science and Technology, Thesis no. 540, Linkõping University, 1996.

[10] A. Seidenberg, *A New Decision Method for Elementary Algebra,* Annals of Math., **60**, pp. 365-374, 1954.

[11] A. Tarski, *A Decision Method for Elementary Algebra and Geometry,* 2nd Ed., Berkeley, University of California Press, 1951.