JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 21, No. 5, September–October 1998

Optimal Robust Tracking Subject to Disturbances, Noise, Plant Uncertainty, and Performance Constraints

Craig N. Scott* and Lincoln A. Wood†
Royal Melbourne Institute of Technology, Melbourne 3001, Australia

This paper presents a versatile approach to optimally track a known input in a real-world environment. It is related to, but different from, conventional l_1 techniques. A key feature of this approach is the separation of the tracking performance optimization from minimizing the controller's sensitivity to disturbances, sensor noise, and plant model uncertainty. These two optimizations are set up as linear programs and make use of the Youla parameterization of a two-parameter controller. The benefits of this formulation are that the solutions found are global rather than local minima and a wide range of design constraints can be incorporated into the optimization–actuator limits, rate limits, and tracking error limits being common examples. Discussion is restricted to single-input/single-output plants, and a numerical example is given.

Nomenclature

D_C	= robustness filter part of controller, feedforward path
d	= denominator polynomial (coprime factor) of P_0, $d_0 + d_1 z^{-1} + \cdots + d_m z^{-m}$
l_1 norm	= $\|h\|_1 = \sum_{i=0}^{n} \|h_i\|$
l_∞ norm	= $\|h\|_\infty = \max_i \|h_i\|$
N_{C_1}	= tracking filter part of controller, q
N_{C_2}	= robustness filter part of controller, feedback path
$N^*_{C_2}$	= digital implementation version of N_{C_2}
n	= numerator polynomial (coprime factor) of P_0, $n_0 + n_1 z^{-1} + \cdots + n_m z^{-m}$
P	= true plant that may contain nonlinearities
P_0	= linear plant model, n/d
u	= commanded plant input from the controller, $u_0 + u_1 z^{-1} + \cdots + u_k z^{-k}$
W_{2-4}	= stable filters on the respective-numbered inputs
w_1	= input reference (desired plant output); can be represented by (w_{1_N}/w_{1_D})
w_2	= disturbance input
w_3	= sensor noise input
w_4	= plant uncertainty input
X, Y, q, r	= stable, rational, linear functions
y	= plant output
z transform	= $h = \sum_{i=0}^{n} h_i z^{-i}$ (consistent with MATLAB notation)
ϕ	= tracking error $(w_1 - y)$, $\phi_0 + \phi_1 z^{-1} + \cdots + \phi_k z^{-k}$
$\phi^{+/-}$, $u^{+/-}$	= nonnegative-valued polynomial sequences

I. Introduction

RESEARCH into l_1 optimal control has been very active in recent times. The standard formulation has been dealt with at length,[1–5] deriving the requirements for zero and rank interpolation constraints. Conditions for stability in the presence of various forms of plant perturbations have been explored,[6–9] as have disturbance rejection and performance criteria.[1,6,8–10] The l_1 problem is most commonly set up as a linear program, and investigation into the solution of these formulations has been undertaken.[1,4,11,12] Optimal tracking subject to magnitude and time domain constraints has been investigated and clearly explained,[5] and inclusion of approximate H_∞ constraints has also been covered.[1,3]

In recent work,[13,14] a new form of constraint equation, valid in both the single and multivariable case, that is easier to implement in a linear program than previous approaches has been derived. The objective was also changed to add greater versatility to the method. The first experimental implementation of an l_1 controller and comparison with existing methods were achieved recently using this approach in the single-input/single-output case.[15,16]

This paper presents a unified approach to optimal tracking that is also optimally insensitive to disturbances, sensor noise, and plant uncertainty. The value of the method lies both in its ability to easily include performance constraints and in its separate treatment of the performance and robustness characteristics of the system. This is done using linear programming and the Youla parameterization for a two-parameter controller.

Plant uncertainty is incorporated into the optimization as a multiplicative perturbation, chosen because of its ease of conceptual understanding—it represents a tolerance band on the nominal transfer function. The physical meaning of other forms of uncertainty can, at times, prove difficult to define, making it hard to know exactly what plant perturbations are included in the uncertainty set. By using the multiplicative model, it is easy to account for things like neglected higher-frequency dynamics or errors in experimental measurement of the plant transfer function.

In the robustness optimization, the insensitivity to the plant uncertainty is included in the same format as the disturbance and noise inputs, whereas stability is ensured by adding a condition on one of these sensitivities. Stable filters on the disturbance, noise, and uncertainty inputs then allow the designer to specify their magnitudes and frequency contents, tailoring the optimization to the environment of the controller.

The method of controller construction also allows for the computational delay that is unavoidable in any experimental digital control implementation. Also, the controller terms (N_{C_1}, N_{C_2}, and D_C) are all directly available from the optimizations themselves, removing the need to extract them from other quantities, which in some cases is a numerically undesirable task.

II. Preliminaries

The system with controller can be represented by the block diagram shown in Fig. 1. The weights W_2 and W_3 are stable filters that represent the frequency content of w_2 and w_3, respectively, allowing w_2 and w_3 to be represented by white noise with a unity magnitude maximum. The controller consists of three filters: N_{C_1}, N_{C_2}, and D_C^{-1}. Because all three terms are taken as polynomial sequences in z^{-1}, the N_{C_1} and N_{C_2} filters will have denominators of 1, whereas the D_C^{-1} filter will have a numerator of 1. The plant P may not be exactly known and may even contain nonlinearities; however, an approximate linear model P_0 is usually obtainable, and so

Received July 18, 1997; revision received March 15, 1998; accepted for publication April 1, 1998.

*Doctoral Candidate, Department of Aerospace Engineering, Wackett Aerospace Centre, GPO Box 2476V.

†Adjunct Professor, Department of Aerospace Engineering, Wackett Aerospace Centre, GPO Box 2476V, and Research Leader, Maritime Systems and Survivability, Aeronautical and Maritime Research Laboratory, Melbourne, Australia. Member AIAA.

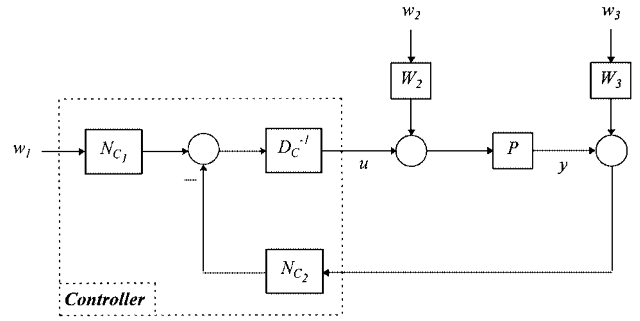


Fig. 1 Two-parameter control scheme.

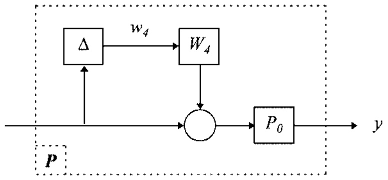


Fig. 2 Plant model.

It is good practice to have some form of limiting on the output of the controller to prevent damage to components of the system in the event of a malfunction or unforeseen event. Therefore, $\|u\|_\infty$ should be known in advance. Because $\|W_2\|_\infty$ is also known, the worst-case $\|w_4\|_\infty$ can be set to $(\|W_2\|_\infty + \|u\|_\infty)$.

Cutting the loop at the output of the Δ block moves the nonlinearity from the system to the w_4 input. Therefore, because all blocks in the modified system are linear, the effect of each input can be considered separately and the components superimposed. For example, if w_2, w_3, and w_4 are set to zero, the effect of w_1 on u can be found by tracing back along the signal path:

$$u = D_C^{-1}\left(N_{C_1} w_1 - N_{C_2} P_0 u\right) \tag{3}$$

$$u = \left(\frac{N_{C_1}}{D_C + N_{C_2} P_0}\right) w_1 \tag{4}$$

Using this approach, one can write a linear matrix equation relating the inputs of the system to the quantities of interest. Of primary interest to the designer are the tracking error ϕ and the demanded plant input u

$$\begin{Bmatrix} u \\ \phi \end{Bmatrix} = \begin{bmatrix} \dfrac{N_{C_1}}{D_C + N_{C_2} P_0} & \dfrac{-W_2 N_{C_2} P_0}{D_C + N_{C_2} P_0} & \dfrac{-W_3 N_{C_2}}{D_C + N_{C_2} P_0} & \dfrac{-W_4 N_{C_2} P_0}{D_C + N_{C_2} P_0} \\ \dfrac{D_C - N_{C_1} P_0 + N_{C_2} P_0}{D_C + N_{C_2} P_0} & \dfrac{-W_2 P_0}{D_C + N_{C_2} P_0} & \dfrac{W_3 N_{C_2} P_0}{D_C + N_{C_2} P_0} & \dfrac{-W_4 P_0}{D_C + N_{C_2} P_0} \end{bmatrix} \begin{Bmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \end{Bmatrix} \tag{5}$$

the plant can be represented as P_0 with a perturbation on its input (see Fig. 2). This is the multiplicative plant uncertainty model for which explanations are readily available.[1,6,8,10,17] It can be shown that the uncertainty will not cause the system to become unstable if the condition in Eq. (1) is satisfied:

$$\left\| \frac{W_4 N_{C_2} P_0}{D_C + N_{C_2} P_0} \right\|_{\text{PEAK GAIN}} \leq 1 \tag{1}$$

The term W_4 represents the tolerance band on the plant's nominal transfer function. For example, if P_0 represents a rigid-body model of a flexible system, it ignores the high-frequency dynamics of the plant. Therefore, $|W_4|$ would be higher at high frequency than at low frequency.

The term Δ is included in the perturbation model of Fig. 2 to account for any nonlinearities and time-varying behavior of the plant.

The coefficient matrix in Eq. (5) is sometimes referred to as the closed-loop map. In its present form, it is somewhat cumbersome to deal with as there is no guidance as to what the controller terms (N_{C_1}, N_{C_2}, and D_C) need to be to give the desired system behavior. The first thing that must be considered is the stability of the system. This means that all internal and external quantities (i.e., the outputs of every block in Fig. 1) must be stable. This can be achieved by using what is called the Youla parameterization, which makes use of a coprime factorization of the linear plant ($P_0 = n/d$) and a parameterized controller. It is commonly expressed as[17]

$$C = D_C^{-1}\begin{bmatrix} N_{C_1} & N_{C_2} \end{bmatrix} = (Y - rn)^{-1}[q \quad X + rd] \tag{6}$$

where $(Y - rn) \neq 0$ and $nX + dY = 1$.

Substituting this into Eq. (5) gives the parameterized form

$$\begin{Bmatrix} u \\ \phi \end{Bmatrix} = \left[\begin{array}{c|ccc} dq & -W_2 n(X + rd) & -W_3 d(X + rd) & -W_4 n(X + rd) \\ 1 - nq & -W_2 n(Y - rn) & W_3 n(X + rd) & -W_4 n(Y - rn) \end{array}\right] \begin{Bmatrix} w_1 \\ \hline w_2 \\ w_3 \\ w_4 \end{Bmatrix} \tag{7}$$

It is a (possibly) nonlinear, time-varying operator that satisfies the condition $\|\Delta\|_{\text{PEAK GAIN}} < 1$, but nothing else need be known about it. Its inclusion means that it is difficult to write a simple equation relating the inputs of the system to its outputs. To overcome this, the representation of the perturbation is modified slightly by cutting the connection between the Δ block and the W_4 block. The input to the W_4 block then becomes another external white noise input w_4, but unlike w_2 and w_3 its maximum magnitude is not necessarily unity. Because it is known that the gain of Δ must be less than unity, it follows that the magnitude of its output (w_4) must be less than that of the input to Δ. Therefore, a condition on w_4 can be written as in Eq. (2):

$$\begin{aligned} \|w_4\|_\infty &< \|W_2 w_2 + u\|_\infty \\ &< \|W_2 w_2\|_\infty + \|u\|_\infty \\ &< \|W_2\|_\infty + \|u\|_\infty \end{aligned} \tag{2}$$

This use of the Youla parameterization assumes that the true plant is fully known, implying that $w_4 = 0$. It has been shown[6] for a one-parameter arrangement that any controller that stabilizes the linear plant P_0 also stabilizes the true plant P when the constraint given in Eq. (1) holds. Applying this to the two-parameter case is a simple extension and is therefore omitted here.

The second consideration, apart from system stability, is system performance. Again, the Youla parameterization proves to be very convenient for this purpose. The quantities q and r may be varied freely and independently of each other (but must still remain stable and rational), hence the name "two-parameter" controller. But more than that, varying q only affects the response to the w_1 input, whereas varying r affects only the response to the w_{2-4} inputs. Thus, by changing q, optimal tracking performance can be obtained, whereas totally independent to this, by changing r, optimal insensitivity to disturbances, sensor noise, and plant uncertainty can be achieved.

Along similar lines to Ref. 18, such insensitivity shall broadly be called robustness in this paper.

III. Tracking

The tracking part of the problem reduces to the following equation:

$$\begin{Bmatrix} u \\ \phi \end{Bmatrix} = \begin{bmatrix} dq \\ 1 - nq \end{bmatrix} w_1 \tag{8}$$

There are two possible approaches to solving this problem. One method, which has been pursued with great success recently,[13–16] is to vary u and ϕ in such a way that a stable, rational q exists (i.e., the solution is feasible). Specifically, u and ϕ may be chosen subject to two constraints for the two-parameter arrangement[13]

$$d\phi + nu = dw_1 \tag{9}$$

and one of

$$\phi(z_i) = 0 \qquad \text{or} \qquad u(z_i) = 0$$

where z_i are the nonminimum-phase zeroes of w_1, if any exist. The first equation ensures that the chosen u will yield the chosen ϕ for the plant P_0. Another advantage of using this equation is that it also enforces most of the interpolation constraints needed to ensure stability. The second constraint completes the set of interpolation conditions. A possible disadvantage of this arrangement lies in the fact that the parameter q does not explicitly appear in any of the constraints (which, on the surface, appears to simplify the problem). It should be noted, however, that q is needed to form the controller, and in some cases, there may be numerical problems with extracting q from ϕ or u.

The second approach to solving the tracking problem can be thought of as varying q explicitly to obtain a favorable u and ϕ. The constraint needed for this approach is just Eq. (8), and so the only thing left to decide is how to vary q. It should be noted that no generality is lost by assuming that q is a polynomial of finite length because this still encompasses all possible stable, rational, linear functions. An objective is needed to guide the solution to some desired state, and a simple but useful goal is to minimize the l_1 norm of the u and ϕ sequences. This keeps both u and ϕ as close to zero as possible by minimizing the sum of their magnitudes. Because long nonzero sequences give larger l_1 norms, this form of objective not only keeps the tracking error and actuator usage down but also tends to yield near-minimum time maneuvers. Using this type of objective allows the tracking problem to be conveniently set up as a linear program for which many solvers are readily available. The problem can be expressed as

$$\min \left\{ \sum_{i=0}^{n} (\kappa_1 |\phi_i| + \kappa_2 |u_i|) \right\} \tag{10}$$

subject to

$$dw_1 q - u = 0$$

$$nw_1 q + \phi = w_1$$

The terms κ_1 and κ_2 allow the importance of ϕ and u in the objective to be specified. For instance, if the designer is primarily concerned with the tracking error and not so much with the actuator, then κ_1 will be set much higher than κ_2. The absolute value signs in the objective are removed by making the usual substitution[3,8] $\phi = \phi^+ - \phi^-$. For the optimal solution, this gives $|\phi| = \phi^+ + \phi^-$.

The term w_1 represents the desired plant output sequence, but it is often easier and much more compact to represent w_1 as the impulse response of a stable, linear filter. The discrete numerator and denominator of the filter can then be used directly in the constraint equations instead of the impulse response. This results in a simpler linear program.

To include the two constraint equations in the linear program, the polynomial convolution needs to be converted into a more tractable form. Convolution of two polynomials can be achieved by converting one of them to a matrix as follows:

$$ab = \begin{bmatrix} a_0 & & & \\ a_1 & a_0 & & \\ \vdots & a_1 & \ddots & \\ a_i & \vdots & \ddots & a_0 \\ & a_i & & a_1 \\ & & \ddots & \vdots \\ & & & a_i \end{bmatrix}_{(i+j+1)\times(j+1)} \begin{Bmatrix} b_0 \\ b_1 \\ \vdots \\ b_j \end{Bmatrix} \tag{11}$$

The problem with doing this for the constraint equations is that q, u, and ϕ are possibly infinite length polynomials. This means that for the example in Eq. (11) b has possibly infinite length, leading to an infinite dimensional a matrix that cannot be represented in a linear program. The authors of Refs. 1 and 12 proposed three methods to make the problem finite dimensional, namely, finitely many variables (FMV), finitely many equations (FME), and delay augmentation (DA). This paper uses the FMV approach, which restricts the length of the b vector and hence the size of the a matrix. FMV is chosen because the solutions generated are feasible (though suboptimal) and the method is intuitively simple to understand. This is equivalent to what is sometimes referred to as deadbeat control. The FME approach is also relatively simple but yields infeasible solutions. The danger of this is that it may lead to undesirable behavior when implementing the controller on a real-world structure. The DA method contains the most information about the solution, but it is more complex and therefore more difficult to incorporate into the linear program format used in this paper.

Finally, some reference inputs w_1 may require the presence of a steady-state actuator input u_{SS}. For example, if w_1 represents a setpoint for the system to attain (making w_1 a step function), plants without at least one pole at $z = 1$ will require a steady-state actuator input to keep it at that setpoint. The problem formulation can be modified[13] to account for this by splitting the actuator input into two parts: 1) a steady-state component that begins at time $t = 0$ and 2) a varying component that is determined by the optimization

$$u = \frac{u_{SS}}{1 - z^{-1}} + \bar{u} = \frac{u_{SS}}{h} + \bar{u} \tag{12}$$

The FMV approach then truncates $\bar{u}$ instead of u. If the problem was left unmodified, it would always be infeasible because the u sequence would never go to zero and would therefore require an infinite number of terms in the optimization. By making the preceding substitution, $\bar{u}$ can go to zero in a finite number of samples, and so a feasible solution can exist. The problem can now be written as

$$\min \left\{ \sum_{i=0}^{n} \left[\kappa_1 \left(\phi_i^+ + \phi_i^- \right) + \kappa_2 \left(\bar{u}_i^+ + \bar{u}_i^- \right) \right] \right\} \tag{13}$$

subject to

$$hdw_{1_N} q - hw_{1_D} (\bar{u}^+ - \bar{u}^-) = w_{1_D} u_{SS}$$

$$nw_{1_N} q + w_{1_D} (\phi^+ - \phi^-) = w_{1_N}$$

Allowing M_1, M_2, M_3, and M_4 to represent the convolution matrices for adw_{1_N}, aw_{1_D}, nw_{1_N}, and w_{1_D}, respectively, this can be put into the final matrix form

$$\min \left\{ \sum_{i=0}^{n} \left[\kappa_1 \left(\phi_i^+ + \phi_i^- \right) + \kappa_2 \left(\bar{u}_i^+ + \bar{u}_i^- \right) \right] \right\} \tag{14}$$

subject to

$$\begin{bmatrix} -M_2 & M_2 & & & M_1 \\ & & M_4 & -M_4 & M_3 \end{bmatrix} \begin{Bmatrix} \bar{u}^+ \\ \bar{u}^- \\ \phi^+ \\ \phi^- \\ q \end{Bmatrix} = \begin{Bmatrix} w_{1_D} u_{SS} \\ w_{1_N} \end{Bmatrix}$$

$$\phi^+,\ \phi^-,\ \bar{u}^+,\ \bar{u}^- \geq 0$$

This formulation is very flexible as it can incorporate any extra constraints that can be represented in the discrete domain. For instance, bounds on the actuator and tracking error can be specified in the form $\bar{u}^+, \bar{u}^- \leq U_{\text{MAX}}$, or rate limiting can be achieved using differences $\pm(\bar{u}_i^+ - \bar{u}_{i-1}^+ - \bar{u}_i^- + \bar{u}_{i-1}^-) \leq U_{\text{MAX}}$, or acceleration limiting using central differences. Indeed, any derivative of ϕ or $\bar{u}$ is easily handled in this way. Such limits need not even be a fixed value but may instead be a time-domain template, such as forcing $|\phi|$ to be less than an exponential decay curve to ensure that the tracking error decreases relatively smoothly without unwanted peaks. Dahleh and Diaz-Bobillo[1] and Elia et al.[3] have also shown how approximate H_∞ constraints could be included into an approach like this one in a natural way.

IV. Robustness

The robustness part of the problem can be written as

$$\begin{Bmatrix} u \\ \phi \end{Bmatrix} = \begin{bmatrix} -W_2 n(X+rd) & -W_3 d(X+rd) & -W_4 n(X+rd) \\ -W_2 n(Y-rn) & W_3 n(X+rd) & -W_4 n(Y-rn) \end{bmatrix} \begin{Bmatrix} w_2 \\ w_3 \\ w_4 \end{Bmatrix} \tag{15}$$

where $nX + dY = 1$, $\|W_4 n(X+rd)\|_{\text{PEAK GAIN}} \leq 1$.

But this can be simplified by observing that $N_{C_2} = (X + rd)$ and $D_C = (Y - rn)$. Again, because X, Y, and r are all stable, rational functions, no generality is lost by representing N_{C_2} and D_C by finite length polynomials in z^{-1}. These substitutions reduce the number of unknown polynomials by one and allow the problem to be written in the more compact form

$$\begin{Bmatrix} u \\ \phi \end{Bmatrix} = \begin{bmatrix} -W_2 n N_{C_2} & -W_3 d N_{C_2} & -W_4 n N_{C_2} \\ -W_2 n D_C & W_3 n N_{C_2} & -W_4 n D_C \end{bmatrix} \begin{Bmatrix} w_2 \\ w_3 \\ w_4 \end{Bmatrix} \tag{16}$$

where $nN_{C_2} + dD_C = 1$, $\|W_4 n N_{C_2}\|_{\text{PEAK GAIN}} \leq 1$.

The objective of the robustness problem is to minimize the actuator activity and tracking error in the presence of the inputs w_{2-4}. A good way to achieve this is to minimize the peak gain from each input to both u and ϕ. Conveniently, because each term in the coefficient matrix reduces to a polynomial in z^{-1}, the peak gain will be the same as the l_1 norm.[10] Therefore, the objective consists of a combination of the l_1 norms of each of the six terms in the coefficient matrix of Eq. (16). Following similar steps to those for tracking, one can express the robustness problem as the following linear program:

$$\min \begin{Bmatrix} \kappa_1 \sum\left(L_1^+ + L_1^-\right) + \kappa_2 \sum\left(L_2^+ + L_2^-\right) \\ + \kappa_3 \sum\left(L_3^+ + L_3^-\right) + \kappa_4 \sum\left(L_4^+ + L_4^-\right) \\ + \kappa_5 \sum\left(L_5^+ + L_5^-\right) + \kappa_6 \sum\left(L_6^+ + L_6^-\right) \end{Bmatrix} \tag{17}$$

subject to

$$\begin{aligned} \left(N_{C_2}\right)_0 &= 0 \\ nN_{C_2} + dD_C &= 1 \\ L_1^+ - L_1^- &= W_2 n N_{C_2} \\ L_2^+ - L_2^- &= W_3 d N_{C_2} \\ L_3^+ - L_3^- &= W_4 n N_{C_2} \\ L_4^+ - L_4^- &= W_2 n D_C \\ L_5^+ - L_5^- &= W_3 n N_{C_2} \\ L_6^+ - L_6^- &= W_4 n D_C \\ \sum\left(L_3^+ + L_3^-\right) &\leq 1 \\ L_i^+,\ L_i^- &\geq 0 \end{aligned}$$

The first constraint requires the first term in the N_{C_2} sequence to be zero. This is to account for the computational delay inherent in any digitally implemented controller. The reason for this requirement will become clear in the next section. Constraint equations (2–7) are converted to convolution matrix form in the same manner as the tracking constraints in Eq. (10), and so this step is omitted here. The summation constraint equation is the linear program equivalent of the constraint ensuring stability in the presence of plant uncertainty.

V. Controller Synthesis

The two optimizations will yield all three required controller terms (N_{C_1}, N_{C_2}, and D_C). Note, however, that if these controllers are to be experimentally implemented, there will be a one-sample delay between the measured plant output and the controller output. An easy way to account for this is to assume that N_{C_2} already includes the effect of this delay. The only way this can be so is if its first term is zero, i.e., $(N_{C_2})_0 = 0$. When implementing this controller in hardware, the delay is then stripped from N_{C_2} because it will be supplied by the system instead. Mathematically, the new controller term is given by

$$N_{C_2}^* = N_{C_2} \big/ z^{-1} \tag{18}$$

Thus, the controller effectively consists of three linear filters: N_{C_1}, N_{C_2}, and D_C^{-1} (see Fig. 1). The order of the controller can be limited to some degree by the number of terms used in the q, N_{C_2}, and D_C polynomials. High-order controllers require more careful thought in their implementation because rounding errors can lead to unexpected behavior and the required computation time between samples becomes an issue. Nevertheless, it has been shown experimentally that controllers with 100–300 terms in such polynomials can be successfully implemented on 33-MHz 386-based personal computer systems with little difficulty.

VI. Example

As an illustrative example, a scaled dynamic model of a spacecraft with deployable solar arrays has been chosen. Data for the model were derived from an experimental rig constructed for previous research.[15,16] The rig consisted of a single torque actuator at the central hub, which also had a rotational position sensor. The panels were assumed rigid, but the hinges between them were flexible. The model itself was eighth order and was discretized at 50 Hz using a zero-order hold. This model possessed a double pole on the stability boundary, which made it a difficult plant to handle. Therefore, it served as a good test for the approach presented in this paper.

Using just the one sensor and actuator (both at the central hub), a 90-deg rotation of the solar panels was to be achieved with minimal vibration of the panels. Obviously, the faster the maneuver could be completed, the better, but the actuator hardware imposed limits on both the torque that could be supplied and on the rate at which the torque was allowed to change. Additionally, the plant model inevitably contained some uncertainty. For instance, the nonlinearities of aerodynamic drag on the panels could not be included in the model (but were obviously not present in the real satellite anyway), and possible stiction of the central hub bearing, particularly at the start of the maneuver was also neglected. Because the linear plant model was found from a parameter identification procedure on the rig, there was also a tolerance on the accuracy of the resulting transfer function. Hence, the optimizations needed to consider the specified maneuver subject to a number of limitations and uncertainties.

Consider first the tracking part of the problem. For a 90-deg rotation, a step input should be used, for which the w_1 terms are easily defined (note that the plant model used radians, not degrees),

$$w_{1_N} = \pi/2, \qquad w_{1_D} = 1 - z^{-1} \tag{19}$$

It is noted that though the reference input is a step, the steady-state actuator input for the plant u_{SS} is still zero due to the presence of a pole at $z = 1$. The actuator hardware was restricted to ± 1 Nm, and so the tracking optimization was limited to ± 0.7 Nm to allow an extra ± 0.3-Nm headroom for disturbance rejection, etc. The actuator rate limit was taken to be 15 Nm/s (0.3 Nm/sample) using backward differences on $\bar{u}$. A further limit of 2.5 Nm/s^2 (0.05 Nm/sample^2) was applied to the first derivative of the actuator rate using central differences. This was done to smooth out the actuator signal somewhat, thereby reducing the severity of vibration induced in the panels during the maneuver. The final form for the linear program with these actuator limits included is

$$\min\left\{\sum_{i=0}^{n}\left[\kappa_1\left(\phi_i^+ + \phi_i^-\right) + \kappa_2\left(\bar{u}_i^+ + \bar{u}_i^-\right)\right]\right\} \tag{20}$$

subject to

$$\begin{bmatrix} -M_2 & M_2 & & & M_1 \\ & & M_4 & -M_4 & M_3 \end{bmatrix}\begin{Bmatrix} \bar{u}^+ \\ \bar{u}^- \\ \phi^+ \\ \phi^- \\ q \end{Bmatrix} = \begin{Bmatrix} 0 \\ w_{1_N} \end{Bmatrix}$$

$$\begin{bmatrix} E & -E \\ -E & E \\ \hdashline F & -F \\ -F & F \end{bmatrix}\begin{Bmatrix} \bar{u}^+ \\ \bar{u}^- \end{Bmatrix} \le \begin{Bmatrix} 0.3 \\ 0.3 \\ \hdashline 0.05 \\ 0.05 \end{Bmatrix}$$

$$\bar{u}^+,\ \bar{u}^- \le 0.7$$

$$\phi^+,\ \phi^-,\ \bar{u}^+,\ \bar{u}^- \ge 0$$

where

$$E = \begin{bmatrix} 1 & & & \\ -1 & 1 & & \\ & -1 & \ddots & \\ & & \ddots & 1 \\ & & & -1 \end{bmatrix} \qquad F = \begin{bmatrix} 1 & & & \\ -2 & 1 & & \\ 1 & -2 & \ddots & \\ & 1 & \ddots & 1 \\ & & \ddots & -2 \\ & & & 1 \end{bmatrix} \tag{21}$$

For this problem, the tracking error and actuator demands were both of interest to the designer because excessive actuator demands would result in an unacceptably high drain on the satellite's resources, whereas high tracking error would imply poor performance. Therefore, the weightings in the objective were chosen to reflect this with $\kappa_1 = \kappa_2 = 1$. The problem was solved with the u and ϕ sequences restricted to 400 terms. Somewhat shorter sequence lengths were also possible, but at the expense of increasing the objective value and hence degrading performance.

Now consider the robustness part of the problem. The aerodynamic drag was treated as a disturbance torque to the hub. The maneuver was observed to be predominantly a rigid-body rotation occurring at a much slower rate than the small panel vibrations. The aerodynamic drag would therefore act mostly to resist the low-frequency rotation, so the drag was considered to have mostly low-frequency content. This guided the choice of W_2. Very little was known about the sensor other than that it was subject to quantization by the 12-bit A/D circuitry and possibly a little noise from electrical interference, and so the frequency weighting filter W_3 was kept uniform across all frequencies at a fairly low value. An estimate of the accuracy of the plant transfer function was about 10% at low frequency and up to 50% at higher frequencies. With the maximum possible actuator demand set at ± 1 Nm and $\|W_2\|_\infty$ known, the choice for W_4 could be made. The frequency weightings on the w_{2-4} inputs were chosen as follows:

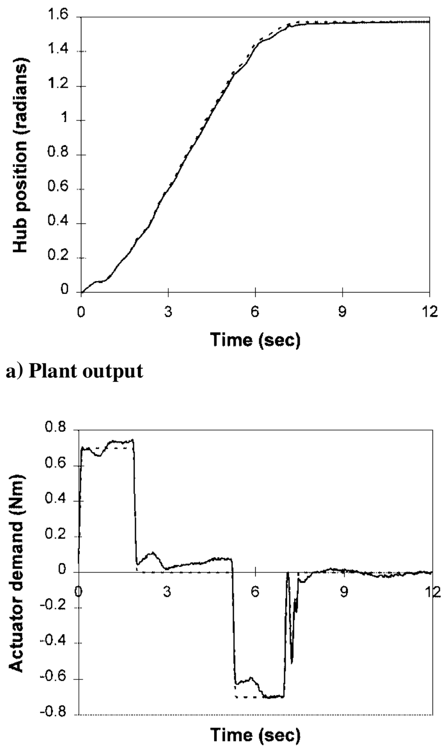
$$W_2 = 0.0075\frac{(s+100)}{(s+10)}, \qquad W_3 = 0.001$$
$$W_4 = 0.5\frac{(s+5)}{(s+25)} \tag{22}$$

Because the transfer functions from all the inputs to both the tracking error and actuator were important, the weightings on all the norms in the objective were set to the same value ($\kappa_{1-6} = 1$). The N_{C_2} and D_C polynomials were restricted to a length of 400 terms, with only a small improvement being observed beyond this length. Again, shorter sequences were possible, but at the expense of performance. The results of the optimization are given in Table 1 where the maximum contribution of each input to u and ϕ is shown. The last column gives the overall maximum contribution to u and ϕ.

It remains now to show how the resulting controller performed in a nonideal environment. Figure 3 shows the results of a simulation where an approximation to aerodynamic drag was used for the disturbance input w_2 and the sensed plant output was subjected to a small amount of white noise, then quantized. The plant model had been altered by changing the system parameters such as the damping factor, hinge stiffnesses, and panel inertias by amounts

Table 1 Robustness results

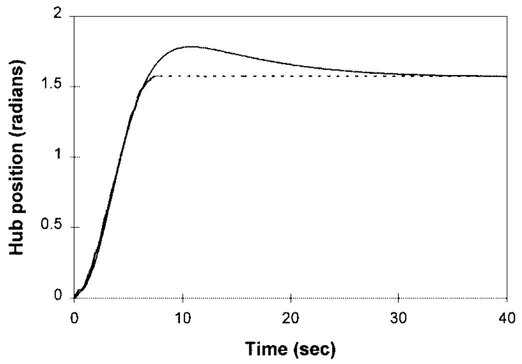
Sequence	w_2	w_3	w_4	Total
u	0.1014	0.0477	0.1411	0.2902
ϕ	0.0344	0.0014	0.0547	0.0904



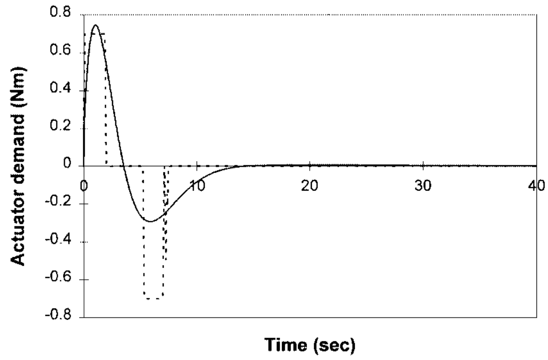
a) Plant output

b) Actuator demands

Fig. 3 l_1 **simulation results for 90-deg rotation demand: · · · · ·, optimal tracking, and ——, simulation results.**



a) Plant output



b) Actuator demands

Fig. 4 H_∞ simulation results for 90-deg rotation demand: · · · · ·, l_1 optimal tracking, and ——, H_∞ simulation results.

representative of possible errors. The dotted line in the figure shows the optimal ϕ and u as obtained from the tracking optimization. It can be seen that the controller maintained performance very close to the optimal in the nonideal environment. For the purposes of comparison, the simulation results for an H_∞ controller are also given in Fig. 4. It can be seen that the performance of the l_1-based controller was far superior to that of the H_∞ counterpart. The l_1 controller exhibited no overshoot and completed the maneuver in about 10 s, whereas the H_∞ controller showed about 14% overshoot and required about 35 s. A major reason for this was that the H_∞ optimization suffered from the need to trade off overshoot with the maximum actuator demanded, something the l_1 optimization did not require. It is noted, however, that the H_∞ controller was of a significantly lower order than was the l_1 design (10th order compared with 400th order). Preliminary investigation suggests that the l_1 controller generated does not lend itself well to order reduction techniques.

An attempt was also made to find a controller using other existing l_1 methods (see Refs. 1 and 6). Because of numerical problems in the optimization, however, no controller was able to be generated that did not significantly violate the maximum actuator limit, even in the no disturbance, noise, or plant error case. It should be noted that both the tracking and robustness optimizations of this example also proved numerically difficult to solve, but following a well-known approach, the linear programs were converted to their dual and solved without difficulty. (Many good references on linear programming give clear explanations of this dual technique. See Ref. 19, for example.) Unfortunately, even when converted to their dual, the optimizations for the other existing l_1 approaches were still unable to generate useful controllers due to numerical problems.

This example has highlighted the flexibility and simplicity of the approach presented. Research into extending the method to multiple-input/multiple-output plants is currently in progress and is showing promise.

VII. Conclusions

A versatile, straightforward method has been presented for designing controllers to optimally track a known reference in a realistic environment. The controllers are formed in two stages: tracking and robustness. The tracking performance is optimized subject to two simple constraint equations, and inclusion of further constraints such as actuator limits is shown to be relatively uncomplicated. The robustness optimization ensures stability to multiplicative plant perturbations and minimizes the system's response to disturbances, sensor noise, and variations in the plant. The more that is known about these quantities, the better the optimization can be tailored to the environment of the system. An upper bound on the magnitude of the actuator and tracking error due purely to rejecting disturbances and noise and to coping with plant variations is also found as a byproduct of the optimization method used. The usefulness of the technique presented is that it provides a complete approach to forming controllers ready for implementation in real-world applications, such as that given in the example.

Acknowledgments

This work has been associated with a research grant from the Australian Research Council. The authors also wish to acknowledge Jerome Vethecan of the Mechanical Engineering Department, Royal Melbourne Institute of Technology, and Robin Hill of the Mathematics Department, Royal Melbourne Institute of Technology, for their helpful comments and guidance.

References

[1]Dahleh, M. A., and Diaz-Bobillo, I. J., *Control of Uncertain Systems, A Linear Programming Approach*, Prentice–Hall, Englewood Cliffs, NJ, 1995, Chaps. 6, 12.

[2]Dahleh, M. A., and Pearson, J. B., "l_1-Optimal Feedback Controllers for MIMO Discrete-Time Systems," *IEEE Transactions on Automatic Control*, Vol. AC-32, No. 4, 1987, pp. 314–322.

[3]Elia, N., Dahleh, M. A., and Diaz-Bobillo, I. J., "Controller Design via Infinite-Dimensional Linear Programming," *Proceedings of the American Control Conference*, Inst. of Electrical and Electronics Engineers, New York, 1993, pp. 2165–2169.

[4]Elia, N., and Dahleh, M. A., "Controller Design with Multiple Objectives," *Proceedings of the American Control Conference*, Inst. of Electrical and Electronics Engineers, New York, 1994, pp. 1603–1607.

[5]Spillman, M., and Ridgely, D. B., "Flight Control Applications of l_1 Optimization," *Journal of Guidance, Control, and Dynamics*, Vol. 20, No. 1, 1997, pp. 49–56.

[6]Dahleh, M. A., and Pearson, J. B., "Optimal Rejection of Persistent Disturbances, Robust Stability, and Mixed Sensitivity Minimization," *IEEE Transactions on Automatic Control*, Vol. 33, No. 8, 1988, pp. 722–731.

[7]Dahleh, M. A., and Ohta, Y., "A Necessary and Sufficient Condition for Robust BIBO Stability," *Systems and Control Letters*, Vol. 11, No. 11, 1988, pp. 271–275.

[8]Dahleh, M. A., and Khammash, M. H., "Controller Design for Plants with Structured Uncertainty," *Automatica*, Vol. 29, No. 1, 1993, pp. 37–56.

[9]Khammash, M. H., "Synthesis of Globally Optimal Controllers for Robust Performance to Unstructured Uncertainty," *IEEE Transactions on Automatic Control*, Vol. 41, No. 2, 1996, pp. 189–198.

[10]Boyd, S. P., and Barratt, C. H., *Linear Controller Design, Limits of Performance*, Prentice–Hall, Englewood Cliffs, NJ, 1991, Chaps. 5, 9, and 10.

[11]Casavola, A., "A Polynomial Approach to the l_1-Mixed Sensitivity Optimal Control Problem," *IEEE Transactions on Automatic Control*, Vol. 41, No. 5, 1996, pp. 751–756.

[12]Diaz-Bobillo, I. J., and Dahleh, M. A., "Minimization of the Maximum Peak-to-Peak Gain: The General Multiblock Problem," *IEEE Transactions on Automatic Control*, Vol. 38, No. 10, 1993, pp. 1459–1482.

[13]Vethecan, J. K., "Design of Compensators for LTI Feedback Systems by Convex Optimization," Ph.D. Thesis, Dept. of Aerospace Engineering, Royal Melbourne Inst. of Technology, Melbourne, Australia, 1996, Chap. 3.

[14]Deodhare, G., Halpern, M., Hill, R., and Vethecan, J., "Simultaneous Minimization of Tracking Error and Actuator Activity in an l_1 Framework," Dept. of Mathematics, Research Rept. 15, Royal Melbourne Inst. of Technology, Melbourne, Australia, Oct. 1994.

[15]Riseborough, P., Hill, R., Vethecan, J., and Wood, L., "l_1 Norm Based Optimal Tracking and Vibration Control," *First Australasian Congress on Applied Mechanics*, Vol. 2, The Inst. of Engineers, Melbourne, Australia, 1996, pp. 819–824.

[16]Riseborough, P., "Dynamics and Control of Spacecraft with Solar Array Joint Nonlinearities," Ph.D. Thesis, Dept. of Aerospace Engineering, Royal Melbourne Inst. of Technology, Melbourne, Australia, 1993.

[17]Vidyasagar, M., *Control System Synthesis, A Factorization Approach*, MIT Press, Cambridge, MA, 1985, Chaps. 5–7.

[18]Zhao, Y., and Kimura, H., "Optimization of Control Input for Deadbeat Control Systems," *International Journal of Control*, Vol. 59, No. 4, 1994, pp. 1107–1117.

[19]Thie, P. R., *An Introduction to Linear Programming and Game Theory*, 2nd ed., Wiley, New York, 1988, Chap. 4.