Numerical Solution of Nonlinear $H_2$ and $H_\infty$ Control Problems with Application to Jet Engine Compressors

Michael Hardt, Member, IEEE, J. William Helton, Member, IEEE, and Kenneth Kreutz-Delgado, Member, IEEE

Abstract—We describe an effective numerical approach to solving nonlinear $H_2$ or $H_\infty$ optimal control problems. Our principal goal will be to use this approach to solve the important problem of jet engine compressor control. The technique is demonstrated first with the tutorial example of the control of a pendulum. We then apply the numerical approach to the problem of controlling jet engine compressor stall and surge instabilities (three-dimensional Moore–Greitzer model) while imposing saturation constraints. Standard in this model is a curve of equilibria along which one may operate the engine. Here, the instabilities are hardest to control near the highest performance equilibria. Our numerical results tell us rather dramatically which equilibrium one can optimally control to and which are unmanageable. The magnitude of the rate saturation constraint on the controller turns out to dominate this phenomenon. We choose a high-performance manageable equilibrium $E$ and compute the $H_2$ optimal law which will control the system to $E$. We then describe plots which allow one to find a neighborhood of the equilibrium within which the closed-loop system is guaranteed to remain. The technique should work with little modification in dimensions 4 and 5, at which point the “curse of dimensionality” forces restrictions.

Index Terms—$H$-infinity control, $H$-infinity optimization, jet engines, nonlinear H2 control, numerical control, optimization methods.

I. INTRODUCTION

The continued popularity of $H_2$ and $H_\infty$ control methodologies is in large part due to their very good stabilization and robustness properties. There are many types of control problems which fit naturally into the $H_2$ or $H_\infty$ framework such as stabilization problems, trajectory tracking, risk-sensitive problems, and disturbance attenuation. In the nonlinear regimes, computational methods for solving these problems are still in their early stages, and there is no generally accepted format for the best approach to solving them. We pursue two such approaches for solving nonlinear $H_2$ and $H_\infty$ control problems which we feel are well suited to handling saturation and other constraints of the problem, are efficient, and can accommodate higher dimensional systems to some extent. We use a combination of optimization and multiple shooting algorithms to solve for a good approximation of the value function of the system throughout the state space. From this, one may readily define a closed-loop control law which is the numerical solution to the nonlinear $H_2$ or $H_\infty$ control problems. Also in Section IV, for purposes of completeness, we mention some other numerical methods which serve as alternative approaches.

We illustrate these methods first in the test case of a swinging pendulum. The underlying purpose in the presentation of the pendulum test case is to clarify the use of the proposed numerical approach with a simple example. Important subtleties of numerical optimal control also become very evident even in this case. For example, there may exist initial conditions of the system for which there exist two completely different, equally optimal solutions. This can lead to situations where a small change in the initial conditions can lead to radically different solution trajectories.

Finally, we apply these numerical methods to the control of a jet engine compressor. The three-dimensional (3-D) Moore–Greitzer model is used as a basis for studying the compressor system, and we use throttle feedback control to robustly stabilize the system under the constraints and performance criteria imposed. We calculate the solution of the nonlinear $H_2$ control problem subject to rate saturation constraints and discuss considerations and different approaches to the problem in light of undesirable transient behavior which can arise. The transient behavior of special concern is a large drop in the plenum pressure rise $\Psi$, and these are punished by the $H_2$ performance criterion.

Our studies indicate that the choice of performance index or cost function to be optimized is largely secondary to the strong effects of actuator rate saturation. The strong influence of rate saturation is best displayed later with Fig. 8 in our discussion of the jet engine compressor. We describe how one may construct a globally optimal controller which meets these rate constraints. The performance of our final controller is then graphically displayed with contour plots of the resulting value function and drops in the plenum pressure in different operating regions. This work represents the first time $H_2$ optimal controllers were computed for this compressor problem.

Much research conducted lately has used throttle feedback and the 3-D Moore–Greitzer model to control the severe instabilities present in a jet engine compressor. Liaw and Abed [19] used a throttle-based linear feedback law to develop a bifurcation-based controller. This control law required sensing of only one of the state variables, namely the squared amplitude of rotating stall, and it had local stability results. Eveker et al. [9] later extended this analysis to design a controller with a much larger operating region. It required, however, measurement of the time derivative of the mass flow coefficient $\dot{\Psi}$. Krstic et al. [15] then developed a backstepping controller which guaranteed global
stability, and it had less restrictive measurement requirements by only requiring sensing of the mass flow $\Phi$ and the plenum pressure rise $\Psi$. It was even shown later [16] that this controller is inverse optimal with respect to a performance index. This result implies robustness and stability margins for the system. Additional alternative approaches to controlling the jet engine compressor may be found in [1], [2], [5], [20], and [22]. The results in this paper were first announced in [12].

The previous research mentioned did not expressly consider the effects that rate saturation might have on the problem. However, there is new research focusing primarily on this issue. In [29], a local approximation of the Moore-Greitzer dynamics is taken at the equilibrium in order to obtain a system of reduced dimension. The reduced system and controller dynamics are then studied together using phase space techniques to understand the effects of saturation constraints. Two other methods for handling rate limits in other nonlinear settings by scheduling or switching linear control laws are found in [8] and [28].

II. NONLINEAR $H_2$ AND $H_\infty$ PROBLEMS

The class of state equations that will be considered will be nonlinear smooth time-invariant having the form

$$
\dot{x} = f(x, u, w) = a(x) + b_1(x)u + b_2(x)w
$$

where $x(t) \in X$ is the state, $u(t) \in U$ is the controlled input, $w(t) \in L^2[0, T]$, $T \geq 0$, is the disturbance input, and $z$ are the to-be-controlled outputs or penalties. The system diagram is shown in [Fig. 1].

A. Nonlinear $H_2$ Control and the HJB PDE

In the nonlinear $H_2$ control problem, no disturbances enter into the model (1), $w \equiv 0$. The problem lies in minimizing a cost functional of the form

$$
V(x) = \inf_{u} \left\{ \frac{1}{2} \int_0^T \|z(t)\|^2 dt + \varphi(x(T)) \right\} (2)
$$

which is often called the value function.

For numerical purposes, it is possible to incorporate the state equations of the system into the cost functional, whereby traditional nonlinear optimization software may be applied to it. One may also solve the problem by defining a Hamiltonian function $H(x, p)$

$$
H(x, p) = p^T a(x) - \frac{1}{2} \int_0^T \|b_1(x) + \frac{1}{2} b_2(x)w \|^2 dt + \varphi(x(T)) (3)
$$

which in the $H_2$ case is convex. The solution to the $H_2$ control problem will satisfy a Hamilton–Jacobi–Bellman (HJB) inequality defined by $H(x, dV^T/dx) \leq 0$. In the numerical methods to be discussed, we will be seeking the null Hamiltonian solution where $H(x, dV^T/dx) = 0$.

The solution of the control problem is thus "simplified" to the solution of a first-order PDE.

B. Bicharacteristic ODE's

It has long been known that this class of PDE's is equivalent to a system of ordinary differential equations (ODE's) of double the dimension [3]. In the case of (3), the corresponding boundary value problem is described by the Hamiltonian vector fields $dx/dt$ and $dp/dt$. The vector $p$ is denoted as the set of adjoint variables for the system, and it is also equal to the transpose of the gradient of the value function, $p = dV^T/dx$. The boundary value problem consists of the following "bicharacteristic" equations:

$$
\dot{x} = \frac{\partial H}{\partial p}(x, p), \quad \dot{p} = -\frac{\partial H}{\partial x}(x, p)
$$

The Hamiltonian will be constant and equal to zero along an optimal trajectory, thus solving the HJB inequality.

This boundary value problem characterizes solving the finite time $H_2$ or $H_\infty$ control problem with no final state constraints. We take the approach in this paper of indirectly requiring the state $x(t)$ to converge to a desired equilibrium $x^*$ by including a penalty function of the squared error of the state from the equilibrium into the performance. This has the effect that the system will converge asymptotically to $x^*$ with speed determined by the gains chosen.

C. Optimal State Feedback Law

The vectors $x$ and $p$ may be used to define the optimal control

$$
u^* = -b_1^T(x)p (5)$$

along each optimal trajectory. The numerical values for $x$, $p$, and $u$ make up an open-loop solution as all variables will only be known as a function of time and only at a discrete set of points along the trajectory. Once one has calculated many such trajectories, one can begin to approximate the numerical function $x \mapsto u$, known only on a grid, by a function $u^*(x)$ given for all $x$ (at least all $x$ in some neighborhood) by a formula. This is done to interpolate between grid points, so one has a control law which can be quickly applied to any state. Probably many approximation methods exist. One is the easy-to-use Matlab neural-network toolbox. Later in the paper we use this to derive the optimal feedback controller for the jet engine control problem.
D. Nonlinear $\mathcal{H}_\infty$ Control

In the nonlinear $\mathcal{H}_\infty$ control problem, the model used (1) contains $L_2$-bounded disturbance inputs $w$. The objective here is to find a controller which guarantees that the closed-loop system has an $L_2$-gain less than or equal to a prespecified constant level $\gamma$ from the disturbance inputs $w$ to the output penalty functions $z$ [27].

We write this $L_2$-gain condition as

$$ \int_0^T \|z(t)\|^2 dt \leq \gamma^2 \int_0^T \|w(t)\|^2 dt + \beta(x(0)) $$

for all disturbance functions $w(.) \in L_2[0,T]$, $T \geq 0$. The constant value $\beta(x(0))$ satisfies $0 \leq \beta(x) \leq \infty$, and $\beta(x^*) = 0$, where $x^*$ is the equilibrium for the system. $z(t)$ is the response of the system for the initial condition $x(0)$.

As in (2), one may define the nonlinear $\mathcal{H}_\infty$ control problem as an optimization of a performance index. Unlike the $\mathcal{H}_2$ problem, however, the optimization is of a minimax-type which prevents the use of standard nonlinear optimization software.

One may similarly define a Hamiltonian function $H(x,p)$, which is now in general nonconvex, for the nonlinear $\mathcal{H}_\infty$ control problem

$$ H(x,p) = p^T a(x) + \frac{1}{2} p^T \left[ \frac{1}{\gamma^2} b_2(x) b_2^T(x) - b_1(x) b_1^T(x) \right] p + \frac{1}{2} h^T(x) h(x). $$

The solution to this problem, as in the nonlinear $\mathcal{H}_2$ control problem, may be obtained by solving for a $C^1$ solution $V(x)$ to the so-called Hamilton–Jacobi–Isaacs (HJI) inequality,

$$ H(x,dV/dx) \leq 0. $$

The solution may also be calculated by solving a set of bicharacteristic equations as in (4).

III. PROPOSED NUMERICAL SOLUTION OF NONLINEAR $\mathcal{H}_2$ AND $\mathcal{H}_\infty$ PROBLEMS

The numerical approach to be discussed consists of several methods which may be used independently or in combination for solving the HJB and, in some cases, the HJI equation. These methods will calculate the open-loop solution to an optimal trajectory. The first method to be discussed is limited in that it can only solve the nonlinear $\mathcal{H}_2$ optimal control problem.

A. Direct Minimization

To solve the $\mathcal{H}_2$ control problem numerically with the direct minimization approach, one discretizes the state and control variables in time over the trajectory. One may then formulate the minimization of the cost functional subject to the differential and output constraints as a finite-dimensional nonlinearly constrained optimization problem. The convexity of the $\mathcal{H}_2$ problem allows conventional Newton-based methods such as sequential quadratic programming (SQP) to be used to solve the problem. Also, the direct minimization approach will find the solution to the HJB equality, $H(x,dV/dx) = 0$. The program DIRCOL from O. von Stryk [25], [26], which uses the nonlinear optimization software NPSOL [11], takes this approach and, in addition, it provides accurate estimates of the adjoint variables $p$.

B. Multiple Shooting

The second method multiple shooting may be applied to either the $\mathcal{H}_2$ control problem or the $\mathcal{H}_\infty$ problem. The equations to be solved are the two-point or multipoint boundary value problem described by (4). An important property of this approach is that it can also handle nonconvex problems.

The prescribed boundary conditions for the problem are initial values of the state $x(0)$ and final values of the adjoint variables $p(T)$. The traditional shooting method integrates the state equations beginning at $x(0)$ and from a guess for $p(0)$ for a prespecified time $T$. When the numerical value obtained for $p(T)$ from the integration does not match the boundary condition $p(T) = \partial \varphi(x(T))/\partial x$, a zero-finding problem may be set up in terms of the constraints, and Newton's method may then be used to iterate on $p(0)$ until the boundary condition is satisfied.

The strong sensitivity of initial value problems, and the small convergence region for the Newton’s method make solving these problems via simple shooting very difficult. A multiple-shooting algorithm, however, is more stable and robust to nonlinearities. This procedure consists of dividing up the time interval $[0,T]$ into a series of grid points $[0,t_1,t_2,\ldots,T]$, and a shooting method is performed between successive grid points (see Fig. 2). A set of starting values for the state and adjoint vectors is required at each grid point in time, and continuity conditions for the solution trajectory introduce additional interior boundary conditions which are then incorporated into one large zero-finding problem to be solved.

The program MUMUS by P. Hiltmann [13] is a multiple-shooting code which can solve two-point or multipoint boundary value problems using a modified Newton method. This method is, in general, quicker than direct minimization, lending itself more readily to iterated calculations of trajectories. A drawback, however, is that the adjoint equations must be programmed as input to the code. For complicated problems, generating the equations by hand or with symbolic manipulation packages may pose a formidable task.

C. Shooting Out

While the multiple shooting method is commonly used for solving a boundary value problem, one may alternatively obtain optimal trajectories by starting near the desired equilibrium $x^*$ and at selected values $\tilde{p}$, then integrating the state and adjoint
differential equations outwards. This process is called shooting out. We are, thus, able to obtain optimal trajectories for the state without having to solve a difficult two-point boundary value problem. A set of starting values (a guess for the solution trajectory) is also not needed as is the case in direct minimization and multiple shooting. This technique was successfully used by McEneaney in [21].

The disadvantage of the shooting out method is that one has less control in reaching a specified set of states. Remaining regions of the state space, that are not able to be mapped out with optimal trajectories via shooting out, may be mapped with the previous multiple shooting approach using as starting values the optimal trajectories already obtained.

D. Combined Method

When one wishes to map the value function over a large region of the state space for purposes of approximating well the optimal closed-loop controller, it is necessary to calculate a fair number of optimal trajectories. We can use the superiority of direct minimization or shooting out methods to generate several of these trajectories when little or no prior knowledge of the solution is available in order to acquire starting values for the multiple shooting method when solving for the remainder of the trajectories. The multiple shooting method works well when many different optimal trajectories are sought after since the algorithm is fast, and one may initialize each run from the previous run. One has much more control in mapping the value function than one might have with shooting out, and the method is, in general, quicker than direct optimization methods.

E. Homotopy

In order to compute the value function $V$ and the optimal control $u^*$, we must compute a family of optimal trajectories which approximately (maybe only coarsely) sweep out most of the state space. Supposing we have computed an optimal trajectory $T_{x_1}$ from a state $x_1$ to the equilibrium $x^*$, to quickly compute trajectories $T_{x_2}$ from many initial states $x$, simply pick $x_2$ near $x_1$ and use $T_{x_2}$ to initialize a MUMUS or DIRCOL computation of $T_{x_2}$. We have found this proceeds quickly to give us many optimal trajectories (see Section VI-E). Henceforth, we refer to this as homotoping trajectories, since the point is to move them gradually.

When using the multiple shooting method, one may similarly homotopy the initial conditions of the state variables, using both the state and adjoint trajectories previously calculated to serve as starting values for the new boundary value problem to be solved. By calculating trajectories over the portion of state space of interest, one will know $u^*$ and $V$ at points over this entire area. Multivariate approximation methods may then be used to find $u^*$ approximating $u^*$ as a function of the state $x$.

After the calculations have been made, the approximation $u^*$ of the closed-loop controller may then be implemented for use in real time.

IV. Other Numerical Methods

There are many numerical approaches which have been tried on HJB and HJI equalities and inequalities and listing them all is beyond us. We shall try to list some of the main ones.

A. Finite Differences

The method of finite differences has been used successfully for solving HJB and HJI problems of low dimension. For an account of finite difference methods and their relation to Markov chains see Kushner and Dupuis [17]. Also Falcone [10] and the thesis by Dower under James [8] have used extensively finite differences in this setting. It is not clear how these would extend to high dimensions in a manageable way, since rectangular grids have a rigid geometry.

B. "Small" Basis of Functions

The idea in this approach is to select basis functions $f_j$, expand the value function $V$ as $\sum_{j=1}^{M} c_j f_j$, then some work is required to find the proper coefficients $c_j$. The main difficulty lies in selecting the basis functions $f_j$. The common approach in conventional numerical partial differential equations (PDE's) (in dimension <4) is to use basis functions which are piecewise polynomials. In principle, in order to solve a first-order PDE like HJB's, they should be continuous. The tessalating (such as tetrahedra) of space into pieces which match up is a serious problem. Doing this flexibly in three dimensions took substantial study, but of course is well developed now. The advantage is that once everything is set up, mesh refinement can be done on regions of space which may otherwise seem treacherous. However, doing this in say five dimensions may well be very difficult.

One can select basis functions in a way which is ad hoc and much less difficult. For example, Beard and McLain [4] use physical intuition coming from a particular plant model to select ten or 12 basis functions. For example, if the model contains sines and cosines and powers of $x$, then their $f_j$'s will too.

For control purposes, one need only solve HJI inequalities, provided the solutions $V$ satisfy conditions which make them what are called viscosity subsolutions. If continuity is not required for $V$ and jumps are allowed, the problem becomes easier, but the control law typically has jumps.

Once a basis of $f_j$ is selected, one can find the $c_j$'s in several ways. One is a Galerkin procedure for solving HJI equations, described for example in [4]. Another is by directly substituting $V = \sum_{j=1}^{M} c_j f_j$ into the HJI inequality to obtain inequalities of the form

$$q(\{f_j : j = 1, \ldots, M\}, x) \geq 0 \quad \forall x$$

$$\sum_{j=1}^{M} c_j f_j \geq 0.$$  

Here, $q$ is quadratic in the $f_j$, and one elects some finite set of $x$ to yield finitely many constraints (crucial areas can be heavily gridded). Various optimization schemes can be applied.

C. Partly Analytical Methods

Very intriguing is backstepping, which applies to special classes of systems and obtains analytical expressions for $V$ and
the controller \( u \). Thus, when it applies, it has the potential for beating the curse of dimensionality. One advantage of obtaining analytical expressions for \( V \) and \( u \) is that often the solution need not be recomputed with changes in the parameter values or choice of equilibrium.

Backstepping does not approach saturation directly, however, although a highly experienced practitioner may well come up with compromises. This makes it difficult to compare our efforts with backstepping, since in the jet engine compressor problem considered here, we will see that rate saturation dominates.

Similarly, feedback linearization is not able to address saturation directly.

\[ \text{D. Characteristics and Direct Minimization} \]

The method of characteristics, as they pertain to multiple shooting and shooting out, and direct minimization methods were presented in the previous section. They seem to have the most promise among methods for "fully" solving HJB equations. The shooting out or multiple shooting methods can be a bit difficult to enter into a program at this point, since the adjoint equations can be complicated. Direct methods, while slower, are comparatively easy to run. All of these methods find the value function \( V(x) \) at each point \( x \) along a curve. In delicate regions, one can pick many \( x \)'s, otherwise only a few \( x \)'s may be necessary to approximate \( V(x) \) well. A careful approach can, thus, help combat the curse of dimensionality.

The final desired result is a "simple" formula to approximate the function \( V(x) \) in order to obtain a control law. Traditional Galerkin methods reverse this in that they layout the approximation scheme first (before having any information about \( V \)). Since the Galerkin scheme must mesh very well with the numerical PDE one is solving, this is much harder to do than merely approximating \( V \) given as data on a grid.

\[ \text{E. Miscellaneous} \]

Other approaches include the technique using viability kernels as promoted by P. Saint-Pierre [24]. Bertsekas and Tsitsiklis also solve HJB and HJI equations for discrete problems in their book [6].

\[ \text{V. ILLUSTRATION: PENDULUM} \]

In order to demonstrate the basics and subtleties of applying these numerical techniques to a control problem, we present here a brief tutorial example of the optimal control of a pendulum. Already in this simple problem, one finds many numerical particularities which are important. For example, we discover multiple sheets of the pendulum. These are regions of the state space where numerically calculated optimal trajectories will stay within when iterating on the initial conditions. Yet as one progresses far enough along the sheets, it may be the case that the calculated trajectories are no longer optimal.

Later, when we deal with the more complex problem of jet engine compressor control, we must, for example, be careful to rule out multiple sheets or to find them. As we shall see in that case, extensive searching did not reveal any such anomalies.

The dynamical equations of motion of a pendulum are

\[
mI^2 \ddot{\theta} + mg \sin(\theta) = u
\]

where \( \theta \) is the position angle of the pendulum with respect to the bottom "rest" position. The pendulum rod is assumed to be massless of length \( l \) with a point mass of \( m \) at its end, and \( g \) is the gravity constant.

Let \( x_1 = \theta \) and \( x_2 = \dot{\theta} \), and for simplicity let \( I = m = 1 \).

Then we write the state and output equations as

\[
\begin{align*}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= -g \sin(x_1) + u
\end{align*}
\]

where \( x_f \) is a desired equilibrium of the pendulum and \( u_f \) is the control effort at the equilibrium. We specify a weighted quadratic cost functional to be optimized as

\[
\min_u \frac{1}{2} \int_0^T \left( ||x - x_f||_Q^2 + ||u - u_f||_R^2 \right) dt + \varphi(x(T))
\]

where \( Q, S \geq 0 \) and \( R > 0 \) are square matrices of appropriate dimension, and the norm \( ||z||_Q^2 \) satisfies \( z^T Q z \) for some vector \( z \) and matrix \( Q \). The penalty function on the final state is equal to \( \varphi(x(T)) = \frac{1}{2} ||x(T) - x_f||_S^2 \).

We wish to solve the HJB equality (3), \( H(x, \frac{\partial V}{\partial x}) = 0 \), for \( V \). From the Hamiltonian, we construct the bicharacteristic equations (4) with initial and final boundary conditions

\[
x(0) = x_0, \quad p(T) = \frac{\partial \varphi}{\partial x}(x(T)) = S(x(T) - x_f).
\]

Once these are solved, a closed-form expression for the optimal control \( u^*(x) \) along the bicharacteristic trajectory can be determined from \( p = \frac{\partial V}{\partial x}(x) \) and the extremal value for \( u^* \)

\[
u^* = -R^{-1} b_1^T(x)p.
\]

\[ \text{A. Nonlinear H}_2 \text{ Solution and Comparison of Numerical Methods} \]

We compare both the direct minimization approach (DIRCOL) and the multiple shooting method (MUMUS) in our numerical experiments. Of interest are their respective run-times, the effects of varying numbers of grid points (in time), and their convergence properties.

With the multiple shooting code MUMUS, for example, it was found that run-times are not as affected by the choice of discretization as the optimization approach of DIRCOL. Also, given a good set of starting values, MUMUS can reach the solution very quickly. The difficulty with MUMUS lies in obtaining reasonable starting values, in particular, those for the adjoint variables. For the pendulum, perhaps due to its low dimension and relatively mild nonlinearities, MUMUS was able to converge well even with the obvious choice of straight-line starting values. When iteratively calculating many trajectories with large numbers of grid points, MUMUS thus was the obvious choice.

The direct optimization package DIRCOL makes it much easier for the user to define the problem and to enter in nonlinear equality and inequality constraints when desired. The use of this approach is very useful in a preliminary analysis of an optimal control problem, even one as simple as the pendulum. Also, the
HJB equality in the nonlinear $H_2$ control problem may have more than one solution. An optimization code will converge to the stabilizing solution of the HJB equality, while it is not guaranteed that the shooting code will do the same. Since for small numbers of grid points the run-times of the two methods are very similar, a natural approach is to solve the problem initially with DIRCOL or by shooting out, then to initialize MUMUS with its output. This was the combined method previously described in Section III.D.

B. Multiple Sheets of the Value Function

The pendulum has been a heavily studied test problem. Even in our case, this example sheds light on some rather interesting properties of optimal control and of our choice of numerical methods. When applying the previously described homotopy method, we discover sheets along which the multiple shooting method will continue to produce solutions to the $H_2$ control problem, though the calculated trajectories may no longer be optimal.

In our analysis, the state $x_1$ represents the position angle which determines the configuration of the system and $x_2$ is the angle velocity. In the coordinates chosen, $x_1 = 0.0$ refers to the bottom "rest" position of the pendulum, and the desired final equilibrium is chosen as $x_f = (x_{f1}, x_{f2}) = (-1.5, 0)$.

In order to calculate $V$ and $u$, the homotopy method is used with the multiple shooting program MUMUS to sweep out a large region of the state space, repeatedly calculating trajectories with slightly varying initial conditions. We take the optimal trajectory with initial condition $(x_1(0), x_2(0))$ as an initialization for another run of MUMUS with initial condition $(x_1(0) + \delta_1, x_2(0) + \delta_2)$. The choice of $(\delta_1, \delta_2)$ will determine how accurate the resulting value function and control law will be. In our experiment, a homotopy was performed over a large region of state space, roughly $-3.9 < x_1 < 2.4, -11 < x_2 < 14$. The final time $T$, to which the trajectories are calculated, was chosen so that all of the optimal trajectories converged to within a small tolerance level of the equilibrium. Finding an appropriate $T$ required some trial and error.

Our experiment consisted of two phases. The initial phase consisted of a homotopy in $x_1(0)$ along the two primary sheets of the pendulum. One sheet traverses the pendulum in the clockwise direction iterating the initial condition $(x_1(0), x_2(0) = 0)$ in the negative $x_1$ direction. Each run uses the previous run as its initialization. The same is done in the counter-clockwise direction. Since $x_1$ is periodic, eventually the homotopy will calculate trajectories on two different sheets corresponding to the same initial condition. These two trajectories will be different since they will correspond to traversing the pendulum in opposite directions to reach the desired final endpoint and; hence, they will have different values for the value function. Recall from (2) that the value function $V(x)$ is the infimum of a cost functional over all possible trajectories. The "true" $V$ will be the lowest of all the values obtained, while the higher values are thrown out as they are not "optimal." More details on combining different calculated values for $V$ to construct the true value function are described by McEneaney [21].

Fig. 3 represents a vertical slice along the $x_1$ axis of the value function. On the vertical axis is the value function $V$ which represents the cost in reaching the desired equilibrium $x^*$. This cost is defined by the quadratic performance (9). Note that since $x_1$ is periodic in $2\pi \approx 6.3$, $x = (-3.9, 0)$ is the same point as $x = (2.4, 0)$. The value function has the same value there,
V(-3.9,0) = V(2.4,0) = 185. At this point, it is equally costly for the pendulum to travel counter-clockwise as it is to travel clockwise in order to reach the desired equilibrium. The state space is not a simply connected region, and it is therefore to be expected that there exist points from which two equally costly and optimal paths exist to the desired final state.

In the second phase, we iterate the initial values of the trajectories we wish to compute in such a manner as to keep \((x_1(0),x_2(0))\) near a level set of \(V(x)\). This way of choosing \((x_1(0),x_2(0))\)'s is not essential but easy to implement. In Fig. 4, we have graphed the value function for \(V(x) \leq 185\). Using an approximation method, we then obtain the desired closed-loop control law.

VI. JET ENGINE STALL AND SURGE CONTROL

The three state Moore-Greitzer model of a jet engine compressor has been frequently used in the literature to study the severe instabilities found in this class of systems. Surge is the name given to oscillations in the annulus averaged flow of the compression system while rotating stall is a disturbance which consists of regions of reduced or reversed flow that rotate around the annulus of the compressor [9] (See [Fig. 5]). The specific model that we will be using is [15]

\[
\dot{\Phi} = -\Psi + \Psi_c - 3\Phi R \\
\dot{\Psi} = \frac{1}{\beta^2}(\Phi + 1 - \gamma\sqrt{\Psi}) \\
\dot{R} = \sigma R(1 - \Phi^2 - R), \quad R(0) > 0
\]  

(11)

The meaning of the state variables is as follows:

- \(\Phi\) annulus—averaged mass flow coefficient;
- \(\Psi\) plenum pressure rise;
- \(R\) squared amplitude of circumferential flow asymmetry;
- \(\Psi_c\) compressor characteristic relating pressure rise \(\Psi\) in the plenum to the mass flow \(\Phi\);
- \(\gamma\) proportional to the throttle area (control);
- \(\beta, \sigma\) system-dependent parameters.

A common approach in constructing a model for the system is to represent \(\Psi_c\) as a cubic in \(\Phi\) which is only an approximation to the actual phenomenon. This is a hypothetical approximation to the true compressor characteristic since the region \((\Phi \leq 1)\) is unstable and cannot be physically observed, while in the stable region \((\Phi > 1)\) experimental data is usually used in practical settings to arrive at the compressor characteristic. A common form of the cubic approximation is

\[
\Psi_c = \Psi_{co} + 1 + \frac{3}{2}\Phi - \frac{1}{2}\Phi^3.
\]

The peak operating point of the compressor coincides with the peak of the compressor characteristic in the region \((\Phi \geq 0)\) at \(\Phi = 1\). At this point the system's linearized equations are only marginally stable when using the throttle parameter \(\gamma\) as the control input. However, it has been shown that the system may be stabilized globally at this equilibrium using a nonlinear feedback [5].

There exists a continuum of stable equilibria in the region \((\Phi > 1)\). The equilibria in this region can be parametrized by \(\Phi = \Phi_0\).

\[
\begin{bmatrix}
\Phi \\
\Psi \\
R
\end{bmatrix}_f = \begin{bmatrix}
\Phi_0 \\
\Psi_c(\Phi_0) \\
0
\end{bmatrix}.
\]

The linearized Moore-Greitzer model is uncontrollable at these equilibria, but it is stabilizable. The model at the \(R = 0\) or no-stall equilibria, prevalent in the \((\Phi > 1)\) region, cannot be stabilized by throttle feedback in the region \((\Phi \leq 1)\). There does, however, exist an unstable set of equilibria in this region, termed the stall-equilibria, for which the model can be stabilized by throttle feedback

\[
\begin{bmatrix}
\Phi \\
\Psi \\
R
\end{bmatrix}_f = \begin{bmatrix}
\Phi_0 \\
\Psi_c - 3(1 - \Phi_0^2)\Phi_0 \\
1 - \Phi_0^2
\end{bmatrix}.
\]

Here, the linearized equations are controllable, though the open-loop system is unstable.
A. Problem Statement

We shall use numerical methods to solve the nonlinear $H_2$ control problem thus obtaining an optimal control law of the form $\gamma^*(\Phi, \Psi, R)$. We began the analysis by studying a proper choice of performance index or cost function. There were two main problems also encountered at the outset which are fundamental to the system and were not able to be remedied by guessing different performances. The first main difficulty that we encounter is what we call the minimum-$\Psi$ problem. This involves a large dip in the pressure variable $\Psi$ along the closed-loop trajectory resulting from the controller $\gamma$. An objective is to design a controller which will keep the system trajectory inside a given neighborhood $\mathcal{N}$ of the chosen equilibrium. Another major difficulty is the rate saturation problem. The controller must meet specified bounds on the throttle rate $\dot{\gamma}$. Also, an absolute bound on $\dot{\gamma}$ is required, but our investigations lead us to suspect that this constraint often becomes inactive with realistic bounds on $\dot{\gamma}$. Both problems will worsen with larger initial values of $R$.

Another practical problem is to first choose a high-performance equilibrium defined by $\Phi_0$ and a control law $\gamma(\Phi, \Psi, R)$ which has the property that all states near to the equilibrium are kept close to it. We illustrate how one may compute a set $\mathcal{N}_I$ of states with the property that a closed trajectory starting at a state in $\mathcal{N}_I$ stays completely inside the specified neighborhood $\mathcal{N}$.

In our model, we chose as parameters $\alpha = 4$, $\beta = 1$, $\Phi_{in} = 0.1469$. The error coordinates of the system with respect to the no-stall equilibria ($\Phi > 1$) will be defined as $\phi = (\Phi - \Phi_0)$, $\psi = (\Psi - \Psi_e(\Phi_0))$, $r = R$. Also $\gamma_I$ will denote the equilibrium control input. We began the analysis by studying a proper choice of performance index or cost function.

B. Choice of Performance Index

A preliminary investigation did not directly impose rate saturation constraints and focused on the maximum pressure equilibrium $(\Phi_0 = 1)$. The goal was to test various quadratic and nonquadratic performance indices in order to find an optimal controller having good transient properties remedying the minimum-$\Psi$ and rate saturation problems. This approach turned out to be unsuccessful.

The general form of the quadratic performance index used was

$$\min_\gamma \int_0^T \{c_1\dot{\phi}^2 + c_2\dot{\psi}^2 + c_3\gamma^2 + c_4(\gamma - \gamma_f)^2\} \, dt \quad (12)$$

while that of one with cross-terms was represented by

$$\min_\gamma \int_0^T \{c_1\dot{\phi}^2 + c_2(\psi - c_5\phi)^2 + c_3\gamma^2 + c_4(\gamma - \gamma_f)^2\} \, dt \quad (13)$$

Placing higher powers on the penalties of the state variables actually worsened the minimum-$\Psi$ problem (see Fig. 6). Thus, in the remainder of the investigation we used a quadratic performance index with all weights set equal to one. Additionally, rate saturation constraints will always be imposed.

C. Rate Saturation Constraints

Either numerical approach (DIRCOL/MUMUS) lends itself easily to imposing rate saturation constraints. A dummy state is introduced which is equal to the control input. This then gives an additional state equation with its derivative equal to the control rate which becomes the new control for the system. In the case of the multiple shooting algorithm, a box constraint on the control rate $\dot{\gamma}$ simply translates into an if statement when the control law is defined. To simplify the problem definition, we added a quadratic term in the control rate $\dot{\gamma}$ to the performance index. This permits us to use the standard approach of setting the partial derivative with respect to the control of the Hamiltonian function to zero and thereby arrive at an expression for the optimal level of $\dot{\gamma}$. The performance index

$$\min_\gamma \int_0^T \{c_1\dot{\phi}^2 + c_2\dot{\psi}^2 + c_3\gamma^2 + c_4(\gamma - \gamma_f)^2 + c_5\dot{\gamma}^2\} \, dt \quad (14)$$

is then used for the remainder of our analysis.

Fig. 7 shows the results of imposing a constraint on the control rate $\dot{\gamma}$ with the constraints $|\dot{\gamma}| \leq 0.1$ and $|\dot{\gamma}| \leq 0.2$. The control $\gamma$ is restricted to start at its final equilibrium value since in practice the controller cannot change values discontinuously. Evidence of the constraint is seen in the slightly larger dip that the pressure $\Psi$ is forced to take. This may be considered to be the price for rate saturation. The initial conditions for this experiment were $\Phi = 1.1$, $\Psi = 2.0$, $R = 0.2$, and the desired equilibrium is $\Phi_0 = 1.2$, $\Psi_e(\Phi_0) = 2.083$, $R = 0$ where the equilibrium pressure lies at $97\%$ of its maximum value of $\Psi_e(1.0) = 2.1469$. In the remaining portion of the paper, we will continue to use a rate saturation constraint of $|\dot{\gamma}| \leq 0.1$ together with a quadratic performance index.

D. Choice of Equilibrium

In an attempt to improve the minimum-$\Psi$ problem, a tradeoff was explored by choosing different equilibria in the no-stall region $(\Phi_0 > 1)$ along the compressor characteristic. The benefit of converging to an equilibrium further down the characteristic curve is that the dip in the pressure $\Psi$ and the control rate $\dot{\gamma}$ along an optimal trajectory will be greatly reduced.

Fig. 8 pertains to the $H_2$ optimally controlled system with the rate saturation constraint $|\dot{\gamma}| \leq 1$. We study the influence of this rate saturation constraint with four different equilibria along the compressor characteristic (solid line). The figure displays the behavior of the closed-loop system after initializing at points near the chosen equilibria. The state of the closed-loop system follows a trajectory (moving in a
Fig. 6. Control $v$ versus time for the $\mathcal{H}_2$ optimal trajectory and different performance measures. The solid line is with respect to a quadratic performance index, and the dashdot and dashed lines are with respect to performance indexes with cross-terms with weights $\epsilon_k = 1$ and $3$, respectively. The circles represent the system initialized at $R = 0$ with a quadratic performance.

counter-clockwise direction) which eventually ends in that equilibrium; these are the loops we see plotted. More specifically, the four equilibria that we consider are $(\Phi_k, \Psi_k, R_k) = (\Phi_k, \Psi_k(\Phi_k), 0)$, with $\Phi_k$ equal to 1.1, 1.2, 1.3, 1.4 and with respective initialization points at $(\Phi_k, \Psi_k, 0.12)$ for $k = 1, \ldots, 4$. Thus, the system begins at $R = 0.12$ signifying a modest size stall level which must eventually reach its equilibrium value of $R = 0$.

Fig. 7. $\Psi$ versus time illustrating effects of introducing different constraints on $v$, the control rate. The solid line is the unconstrained trajectory while the dashdot and dashed line represent a constraint of $|v| \leq 0.2$ and $|v| \leq 0.1$, respectively.

Striking is that the closed-loop optimal trajectory converging to the $\Phi_2 = 1.1$ equilibrium exhibits a very large drop in pressure before convergence. In contrast, the equilibrium with $\Phi_2 = 1.2$ has a pressure which is at 97% of the maximum possible
pressure, and the trajectory initialized by our modest displacement from it has only a small pressure drop. The explanation for this dramatic change in performance of our optimal $H_2$ controller is that the rate saturation constraint becomes active for a significant time along the trajectory converging to the $\Phi_1 = 1.1$ equilibrium, and does not become active along the other trajectories.

Table I compares key features of the trajectories in Fig. 8. Listed are the values of the desired mass flow $\Phi_0$ which define the equilibria, the equilibrium pressure $\Psi_e(\Phi_0)$, the percentage of the maximal equilibrium pressure $\Psi_e(1.0)$ that the desired equilibrium lies at, the minimum value that $\Psi$ reaches along the trajectory, and the maximal value of $\gamma$ and $|\gamma|$.

E. Calculation of Value Function

For purposes of calculating a value function (2), we chose as a desired equilibrium $\Phi_0 = 1.2$, where the pressure is at 97% of the maximum equilibrium pressure. Furthermore, we assumed a rate saturation constraint of $|\gamma| \leq 0.1$. Then 381 trajectories were calculated from various initial values $(\Phi, \Psi, R)$ to the equilibrium $(\Phi_0, \Psi_e(\Phi_0), 0)$. The initial values were equally spaced with ranges $\Phi \in \{1.0, 1.4\}$, $\Psi \in \{1.6, 2.2\}$, and $R \in \{0.0, 0.2\}$. The final time $T$ to which the trajectories were calculated was set at 20.0 s, which allowed ample time for the system to converge from any state in the given region. The average time to calculate each trajectory with MUMUS was 6.3 s on an old SPARC IPX. Varying the grid size in MUMUS was at times also necessary to ensure convergence from initial values in the state space, since areas of extremely high curvature in the trajectories cause numerical troubles. Since each trajectory contains the open-loop value for the control law, the various trajectories were needed to calculate a reasonable approximation for the closed-loop control law. A slice of the value function at $\{R = 0.12\}$ is displayed in Fig. 9.

Trajectories calculated from some areas of the state-space exhibit poorer transient properties than from others. In particular, the closer the initialization is to the peak of the compressor characteristic, the more difficulty is encountered in controlling $R$ to 0. In the presence of the rate saturation constraint $|\gamma| \leq 0.1$ and starting close to the peak, the optimal trajectory would allow a substantial increase in $R$ before controlling it to zero. Such transient behavior may often be impractical or nonimplementable as the growth in $R$ signifies the growth of the stall cell. This effect becomes more pronounced with larger initial values of $R$. We see this as a consequence of the rate saturation constraint which prevents the controller from keeping the stall instability from growing.

F. State Feedback Control

After the calculation of the value function, there are a number of methods which may be used to approximate the control law in (5). The calculated trajectories provide us with numerical values of $\Phi$, $\Psi$, $R$, and $\gamma^*$ on a grid in the state space. One may then approximate the data $\gamma^*(\Phi, \Psi, R)$ with the feedback control law $\hat{\gamma}^*(\Phi, \Psi, R)$ as described in Section II-C. In [23], it was shown...
Let $x = [\Phi \Psi R]^T$ represent the state vector. Then the control law is

$$\hat{y}^*(x) = \sum_{i=1}^{10} w^{(2)}_{ij} z_i + b^{(2)}$$

(15)

where the values $z_i$ (corresponding to the ten hidden nodes) are calculated as

$$z_i = g\left(\sum_{j=1}^{3} w^{(1)}_{ij} x_j + b^{(1)}\right)$$

(16)

using the logistic sigmoid function $g(a) = \frac{1}{1+e^{-a}}$. Note that the numerical online burden for implementing this controller is at each time step ten function evaluations and about 80 flops.

### G. Evaluation of Results

Figs. 10–12 best demonstrate the effectiveness of the controller constructed in the previous section. Fig. 10 gives a good indication about the quality of minimum-$\Psi$ performance one may theoretically expect with the implementation of this feedback control law. In it we initialize $R(0)$ to 0.12 and choose as a desired equilibrium $(\Phi_0, \Psi_c(\Phi_0), R) = (1.2, 2.083, 0.0)$ which lies at 97% of the peak operating pressure. The white neighborhood in the figure, for example, contains the set of initial states with the property that the controller $\hat{y}^*$ drives these to the equilibrium $x$ along a trajectory upon which $\Psi$ remains above 2.0. Likewise, the next shaded neighborhood consist of states which are $\hat{y}^*$ controllable with pressure always remaining above 1.95.

Fig. 11 displays $\max|\Phi - \Phi_0|$ when starting from different points throughout the state space while Fig. 12 contains the maximum values of $R$ when starting from $R(0) = 0.12$. Note that in the majority of the state space, $R$ decreases monotonically with implementation of the optimal closed-loop controller resulting...
in the large region labeled by $R \leq 0.12$. Only when initializing in the upper-left corner of the $(\Phi, \Psi)$ plane near the peak of the compressor characteristic ($\Phi_0 = 1.0$, $\Psi_0(\Phi_0) = 2.117$) does the nice convergence of $R$ deteriorate. When initializing at $(\Phi_0 = 1.0, \Psi_0(\Phi_0) = 2.117, R = 0.12)$, $R$ increases to 0.88 which symbolizes almost a full stall of the compressor. Fortunately, this was the only point in the region shown where the closed-loop controller was not able to prevent an excessive growth in the value of $R$. A similar effect is shown in Fig. 11 where the deviations from the equilibrium value of $\Phi_0$ become much larger near the peak of the compressor characteristic.

The three figures allow one to determine neighborhoods of initial values $X_0$ which produce closed-loop trajectories inside a given box $\mathcal{N}$ of the specified equilibrium $x$ at $\Phi_0 = 1.2$. If a performance criterion is to keep the system state within a prespecified region around the equilibrium, then one can determine
the set of initial values \( N_f \) for which this constraint is met with implementation of the closed-loop controller. In other words, we have constructed a graphical display of the operating region of the compressor with implementation of the optimal closed-loop controller developed in this paper.

**H. Timing and Scaling to Higher Dimensions**

The numerical methods described here can easily solve optimal control problems in higher dimensions. The run-times necessary to calculate optimal trajectories grow with increased dimension, yet their growth is slow enough such that it generally does not become a limiting factor. The “curse of dimensionality” however becomes apparent when one attempts to “map out” the state space by iteratively calculating optimal trajectories. For example, assume one were to use a higher order model of the jet engine compressor with dimension 4 and calculate optimal trajectories over a grid of initial conditions in four of the state variables. Even a rough grid of eight different values for each state variable would require the calculation of \( 8^4 = 4096 \) trajectories. At 10 s/traj (SPARC IPX), this would require more than 11 h of calculation plus storage considerations.

A more intelligent selection of trajectories to be calculated would exploit the principle of optimality which tells us that intermediate points reached along a trajectory also represent initial conditions for optimal trajectories. Consequently, the value function is automatically calculated along the entire trajectory. For example in \( R^4 \), one may need only calculate \( 8^3 = 512 \) trajectories. Likewise in dimension 5, a cautious estimate on run times is 20 s/traj on a SPARC IPX for \( 8^4 \) trajectories which yields about 22 h of calculation. Thus, one can avoid much excess calculation by choosing the trajectories to more efficiently map out the state space.

**VII. CONCLUSION**

The purpose of this paper was to show that computational methods for solving nonlinear \( \mathcal{H}_2 \) control problems can be effective and practical in solving significant problems which do not lend themselves to standard linear techniques. We outline a method by which this may be done using available numerical software, then we apply this to the difficult problem of jet engine compressor control. One who has developed a valid disturbance model might use these methods for nonlinear \( \mathcal{H}_\infty \) control.

The dominating effect of rate saturation in the control of jet engine compressors is recognized, and a closed-loop controller is constructed which satisfies these constraints. Graphical contour plots highlight the system performance and transient behavior one can predict given our model.

**ACKNOWLEDGMENT**

A variation of the computational scheme proposed has already been used on aerospace problems \([7], [14]\) in the context of differential games. The authors are grateful to O. von Stryk for describing his work on robots, as well as providing his program. The authors also wish to express appreciation to P. Hiltmann and P. Gill for the use of their programs. Many thanks to M. Breitner, M. Krstić, D. Mayne, R. M. Murray, and to P. V. Kokotović and his students for their comments and suggestions.

**REFERENCES**


Michael Hardt (M’99) received the B.S. degree in mathematics and the M.A. degree in applied mathematics and statistics from the University of California, Los Angeles, in 1991 and 1993, respectively. Until 1999, he has been a doctoral student at the University of California, San Diego, studying robotics and control theory in the Department of Electrical and Computer Engineering.

He has investigated recursive symbolic robot dynamics algorithms and their uses in computer vision and the control of articulated mechanical systems. His interests extend to the use of various numerical methods for the optimal and near-optimal control of nonlinear systems, including the problem of bipedal locomotion.

J. William Helton (M’73) was born in Jacksonville, TX, in 1944. He received the bachelor's degree in mathematics from the University of Texas, Austin, in 1964, and the master's and Ph.D. degrees in mathematics from Stanford University, Stanford, CA, in 1966 and 1968, respectively.

From 1968 to 1973 he was at the State University of New York, Stony Brook, as an Assistant and then Associate Professor. In 1974 he visited the University of California, Los Angeles, for six months and subsequently moved to the University of California, San Diego, as an Associate Professor, where he is currently Professor of Mathematics.

Dr. Helton was a recipient of a Guggenheim fellowship and an Outstanding paper award from the TAC in 1986. He has delivered plenary addresses at the annual meeting of the AMS, at the EECTD, at several of the International Symposia on the Mathematical Theory of Networks and Systems, as well as at other meetings. His earlier articles concerned circuit theory, distributed systems, and aspects of the theory of operators on Hilbert space which come from circuits, systems, differential and integral equations, and noncommutative differential geometry. His theoretical studies of amplifier design then led to an interest in \(H_{\infty}\) control. Recent work focuses on extending \(H_{\infty}\) control to nonlinear systems, computer algebra aimed at linear systems research, and frequency domain optimization theory and algorithms.

Kenneth Kreutz-Delgado (S’79–M’84–SM’93) received the B.A. and M.S. degrees in physics and the Ph.D. degree in engineering systems science, all from the University of California, San Diego (UCSD).

He is a member of the AASAS and IEEE Robotics, SMC, Control and Signal Processing Societies, and is currently on the faculty of the UCSD Electrical and Computer Engineering Department as an Associate Professor of Robotics and Machine Intelligence. Before joining the faculty at UCSD, he was a Researcher at the NASA Jet Propulsion Laboratory, California Institute of Technology, where he worked on the development of intelligent telerobotic systems for “in space exploration and satellite servicing and repair. His interest in sensor-based intelligent systems that can function in unstructured and nonstationary environments is the basis for his current research activities in real-time adaptive sensory-motor control, real-time recognition/classification from multiple time-series data sets, multibody systems theory (dynamics and kinematics), and in the development of "emergent" intelligent behavior from distributed complex systems.