Geometric Approaches to Control in the Presence of Magnitude and Rate Saturations*

Richard M. Murray†
United Technologies Research Center
East Hartford, CT. USA

February 10, 2011

Abstract

This paper gives a survey of some recent results on control of systems with magnitude and rate limits, motivated by problems in real-time trajectory generation and tracking for unmanned aerial vehicles. Two problems are considered: stabilization using "nonlinear wrappers" to rescale a given control law and real-time trajectory generation using differential flatness. For both problems, simplified versions of the general problem are studied using tools from differential geometry and nonlinear control to give insights into the limitations imposed by magnitude and rate limits and provide insights into constructive solutions to the trajectory generation and tracking problems.

1 Introduction

One of the most significant sources of nonlinearities in control systems is actuator saturation, which occurs in virtually all modern systems but is widely ignored by the existing analysis and synthesis tools. At present, there is no systematic means of analyzing and designing nonlinear control systems in the presence of magnitude and rate saturations.

One application area in which saturation nonlinearities are particularly prevalent is flight control of high agility aircraft, where actuator saturation has a significant effect on the overall stability of aircraft. A specific instance is the YF-22 crash of April, 1992, which has been blamed on a pilot-induced oscillation (PIO) caused in part by time-delay effects introduced by rate saturation of control surfaces [9]. A similar example is the Gripen JAS 39 aircraft, which crashed in August, 1993 due to pilot-induced oscillations in which saturation played a strong role. As the complexity and performance of flight systems increase, stronger theoretical understanding is required to avoid such situations and guarantee performance of the system in the face of noise and unmodeled dynamics.

At the practical level, saturation in most systems is handled in an *ad hoc* fashion. Gains are chosen and artificial saturations are inserted such that the system performs well in simulations and experimental tests. Rate saturations are sometimes modeled as equivalent time-delays to allow the use of linear control theory [4]. While these techniques work for systems of low dimension and

*Research funded in part by ONR grant N00014-96-1-0804 and AFOSR grant F49620-95-1-0419.
†On leave from the Division of Engineering and Applied Science, California Institute of Technology, Pasadena, CA.

reasonably straightforward dynamics, as systems get more complicated the difficulties in this ad hoc approach become more noticeable (as in the crash of the YF-22). Further demands for increased performance and agility will exacerbate this problem.

Many techniques are available in the linear literature for incorporating actuator saturations into the design process. One example is the use of l_1 analysis and synthesis techniques, which allow specification of the maximum output response as a function of the maximum size of noise and disturbances [6, 7]. Other techniques include the use of convex optimization to design controllers with a variety of input and performance constraints [3] and recently Lin [19] introduced a low-and-high gain linear optimal control law for semi-global stabilization. A major limitation of these approaches is that they only work for linear systems and they generate linear controllers. As a result, the designs can be very conservative (since the gain must be small enough to tolerate the worst case scenario) and it may be difficult to achieve high performance.

Several new nonlinear tools have been introduced in the last several years for analyzing and controlling linear and nonlinear systems with saturation. One of the fundamental techniques is based on the thesis work of Teel [29, 30], who showed how to stabilize a chain of integrators using nested saturation functions. This result is significant since it is known that it is not possible to stabilize a chain of three or more integrators using a linear control law followed by a saturation function. Thus even simple linear systems with simple saturations can give rise to difficult nonlinear problems. Teel's approach generates nonlinear controllers which are locally linear, but become nonlinear as the inputs grow toward the saturation limits. Another approach has been introduced by Megretski [22], who uses a gain scheduling approach to generate nonlinear stabilizers for saturated linear systems. In addition to stability, Megretski shows that for stable plants the map from plant disturbances to control inputs is L_2 bounded. A more general approach has been recently developed by Teel [31, 28], who shows how to dynamically combine a local and global controller to achieve stability and performance in the presence of input constraints.

In this paper we make use of tools from differential geometry to propose new approaches to design of control systems in the presence of magnitude and rate limits. Our thesis is that geometric tools can expose the underlying principles that describe the limits imposed by magnitude and rate constraints, and that they can be used to develop new techniques for controller synthesis for this important class of systems. Section 2 considers the stabilization problem, focusing on the class of homogeneous systems and introducing the notions of nonlinear wrappers and dynamic rescaling. Section 3 focuses on the trajectory tracking problem, making use of differential flatness techniques and presenting algorithms for real-time trajectory generation for linear systems. Finally, Section 4 gives a broad summary of the results and discusses some of the directions for future work.

2 Stablization of Homogeneous Systems

We begin with the problem of stabilization of an equilibrium in the presence of actuator constraints. We consider first the specialized problem of stabilization of homogeneous systems in the presence of magnitude constraints and make use of a "nonlinear wrapper" to convert an existing control law into a bounded control. With this as motivation, we then consider a technique for "dynamic rescaling" of control laws in the presence of magnitude and rate limits, applied to a specific flight control example.

2.1 A brief introduction to homogeneous systems

We begin with a brief introduction to homogeneous systems, following [21]. Additional information on homogeneous systems can be found in the work of Kawski [15].

Conventionally, a function is said to be homogeneous in a set of variables $x_1, \ldots, x_n$ if it is a polynomial function and each term in the polynomial is the same total order. For our purposes, we allow each variable to be weighted differently and hence we will define a *dilation* $\delta_\lambda^r : \mathbb{R}^n \to \mathbb{R}^n$ as

$$\delta_\lambda^r(x) = (\lambda^{r_1} x_1, \ldots, \lambda^{r_n} x_n) \tag{1}$$

where $\lambda > 0$ and $r = (r_1, \ldots, r_n)$ is a vector of rationals that describes the nonuniform weights for each coordinate. We write δ_λ in place of δ_λ^r for simplicity.

Given this dilation, a function $h : \mathbb{R}^n \to \mathbb{R}$ is *homogeneous of degree $l > 0$ with respect to δ_λ* if $h(\delta_\lambda(x)) = \lambda^l h(x)$. It is easily verified that this corresponds to the usual definition of a homogeneous (polynomial) function in the case that δ_λ is given by the standard dilation corresponding to $r = (1, \ldots, 1)$. We will make use of the fact that a homogeneous, degree zero function is a bounded function since $f(\delta_\lambda(x)) = f(x)$, which remains bounded as $\lambda \to \infty$.

A vector field $F : \mathbb{R}^n \to \mathbb{R}^n$ is homogeneous of degree m with respect to δ_λ if the ith component $F_i(x)$ is degree $r_i - m$ for $i = 1, \ldots, n$. Note that m is typically non-positive with this particular definition of homogeneity for vector fields. It can be easily verified that if h is a homogeneous function of degree l and F is a vector field of degree m, the Lie derivative of h with respect to F, $L_F h$, is homogeneous of degree $l + m$ (remembering that m is negative). Similarly, the Lie bracket $[F_1, F_2]$ is homogeneous order $m_1 + m_2$.

For a given dilation, we define the *Euler vector field* corresponding to δ_λ as

$$E^r(x) = \begin{bmatrix} r_1 x_1 \\ r_2 x_2 \\ \vdots \\ r_n x_n \end{bmatrix}. \tag{2}$$

A more geometric definition of homogeneity can be given by starting with an asymptotically unstable vector field E (corresponding to the Euler vector field) and defining a function α to be homogeneous order l if $L_E \alpha = l\alpha$ and a vector field F to be homogeneous order m if $[E, F] = mF$. One can easily verify that for the Euler vector field, this corresponds precisely to the coordinate definitions just given.

Example 1 (Linear vector fields)**.** All linear vector fields $F(x) = Ax$ are homogeneous order 0 with respect to the standard dilation, $r = (1, \ldots, 1)$. The flow of the Euler vector field associated with this dilation consists of radial lines emanating from the origin.

Example 2 (Chain of integrators)**.** The vector field corresponding to a chain of integrators

$$F(x) = \begin{bmatrix} x_2 \\ x_3 \\ \vdots \\ x_n \\ 0 \end{bmatrix} \tag{3}$$

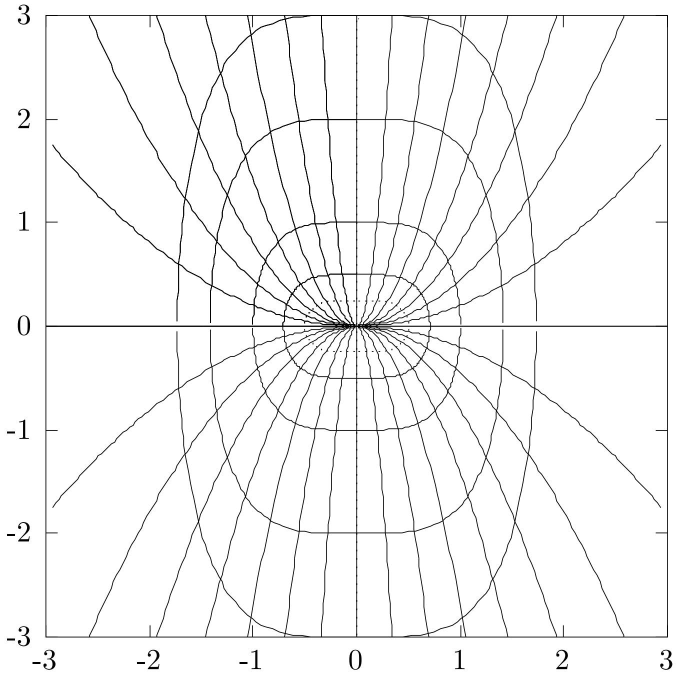

Figure 1: Flow associated with the dilation given by $r = (2, 1)$. The closed curves show the evolution of level sets under the flow of the Euler vector field.

is homogeneous order -1 with respect to the dilation given by $r = (n, n-1, \ldots, 1)$. (Since it is linear, it is also homogeneous order 0 with respect to the standard dilation.) The flow of this vector field for the case $n = 2$ is show in Figure 1. Notice that the rescaling is non-uniform in the different coordinate axes, causing an ellipse to change its principal axes under the dilation.

2.2 Nonlinear Wrappers for Homogeneous Systems with Magnitude Constraints

In this section we show how to modify control laws for homogeneous systems to produce bounded control laws. We make use of a notion that we term a "nonlinear wrapper", which takes a given control law and modifies its inputs and outputs to produced a bounded, composite control law (depicted in Figure 2). The modification of the input to the control and the output from the control is done so as to preserve the overall stability of the system while keeping the magnitude of the control action bounded. The advantage of such an approach, if successful, is that it allows the modification of a previously synthesized control law (that ignored the magnitude constraints) into a control law that respects those constraints. It is thus hoped that the original performance of the control law can be maintained when the constraints are not active and the performance can be degraded gracefully as the system pushes against the constraints.

Consider a homogeneous system

$$\dot{x} = F(x) + G(x)u \tag{4}$$

where F and G are homogeneous order $\tau < 0$ and $u = \alpha(x)$ is a control law that gives global asymptotic stability for the closed loop system. We also assume the existence of a Lyapunov function $V(x) > 0$ such that $\dot{V} < 0$ along trajectories of the closed loop system.

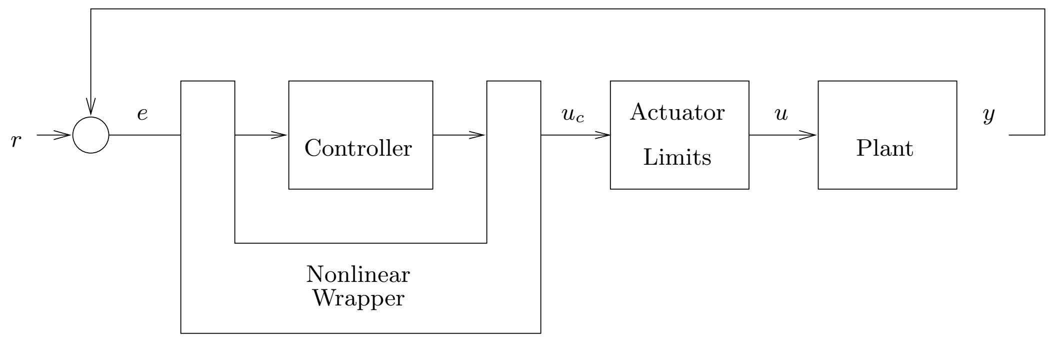

Figure 2: A nonlinear wrapper for modifying a control law to take into account actuation constraints.

Theorem 1 (Morin et al [24]). *Let $\lambda(x)$ be a scalar function given by*

$$\lambda(x) = \begin{cases} 1 & V(x) \leq 1 \\ \textit{solution of } V(\delta_\lambda(x)) = 1 & \textit{otherwise} \end{cases} \tag{5}$$

Then rescaled feedback law given by

$$\tilde{u} = \begin{cases} 0 & x = 0 \\ \frac{1}{\lambda(x)}\alpha(\delta_{\lambda(x)}(x)) & x \neq 0, \end{cases} \tag{6}$$

is homogeneous order 0 when $V(x) > 1$ (and hence $\tilde{u}$ is bounded) and the closed loop system is globally asymptotically stable.

The proof of this theorem is given in [24] and amounts to showing that $1/\lambda(x)$ is decreasing and using this to show that x is decreasing. This is illustrated in the following example.

Example 3 (Chain of two integrators). Consider the case of a double chain of integrators

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= u \end{aligned} \tag{7}$$

together with a stabilizing controller $u = -a_1x_1 - a_2x_2$ and the Lyapunov function $V(x) = a_1x_1^2 + x_2^2$. The control law given by the theorem has the form

$$\tilde{u} = -a_1\lambda^2 x_1 - a_2\lambda x_2 \tag{8}$$

and, away from $x = 0$, λ is given by

$$\lambda = \sqrt{\frac{2}{x_2^2 + \sqrt{x_2^4 + 4a_1x_1^2}}}. \tag{9}$$

We now consider the feedback given by using the scaled feedback for all x such that $V(x) > 1$ and the original feedback otherwise.

To show that this system is stable, let $\gamma(x) = 1/\lambda(x)$ and let

$$V(\gamma, x) = V(\delta_{1/\gamma}(x)) = \frac{a_1}{\gamma^4} x_1^2 + \frac{1}{\gamma^2} x_2^2. \tag{10}$$

Thus

$$\gamma = \sqrt{\frac{x_2^2 + \sqrt{x_2^4 + 4a_1 x_1^2}}{2}}. \tag{11}$$

Differentiating the equality $V(\gamma(x), x) = 1$, we obtain

$$\frac{\partial \gamma}{\partial x} \dot{x} = -\left(\frac{\partial V}{\partial \gamma}\right)^{-1} \frac{\partial V}{\partial x} \dot{x}. \tag{12}$$

Since $\frac{\partial V}{\partial \gamma} < 0$ and $\frac{\partial V}{\partial x} \dot{x} < 0$, it follows that γ is non-increasing along the trajectories of the system. From the form of γ, it is clear that this implies that $\|x\|$ is also decreasing and it follows (with some details omitted) that x is asymptotically stable.

A few remarks are in order with respect to the results of this theorem. First, the theorem requires both a control law and a Lyapunov function, which controls the rescaling process. In many cases the determination of such a Lyapunov function is easily done, but it is not clear which of many possible functions one should use or the effect of that function on the overall performance of the system. Second, although the theorem guarantees that the rescaled control law is bounded, it does not give a specific computation for what this bound is. Thus, one must compute the bound and modify the Lyapunov function (by rescaling) if the bound is too large. Finally, the theorem requires that the open loop system be homogeneous of order $\tau < 0$. Unfortunately this is a quite restricted set of systems and does not even include most linear systems.

A particularly insightful special case of the theorem is to consider the case of a chain of n integrators

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= x_3 \\ &\vdots \\ \dot{x}_{n-1} &= x_n \\ \dot{x}_n &= u \end{aligned} \tag{13}$$

with a linear state feedback law

$$u = k_1 x_1 + k_2 x_2 + \cdots + k_n x_n. \tag{14}$$

The rescaled control law in this case takes the form

$$u = k_1 \lambda^n x_1 + k_2 \lambda^{n-1} x_2 + \cdots + k_n \lambda x_n. \tag{15}$$

The form of the scaling is quite interesting: for λ small (which will occur when V and hence x is large), the control law is scaled such that the effect of the states at the top of the integrator chain are much smaller than those at the bottom. In effect, this forces the control law to react in such a way to first make the states nearest the input small. As these states decrease in magnitude, the control will pay increasing attention to states toward the top of the integrator chain.

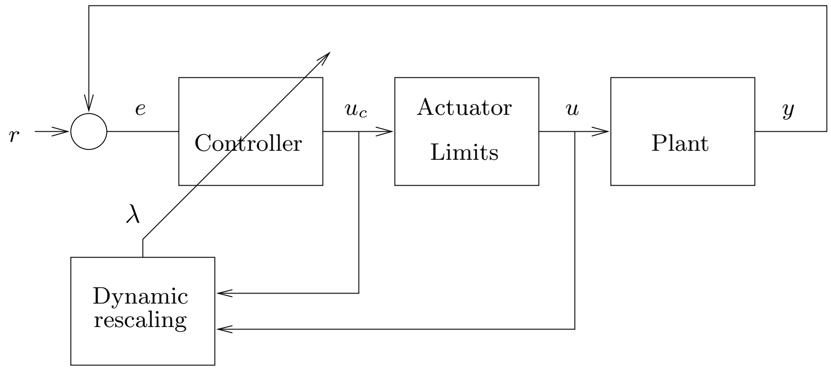

Figure 3: Dynamic rescaling for control in the presence of magnitude and rate limits.

In effect, this nonlinear scaling insures that the states at the top of the chain do not dominate the response of the system until the states at the bottom have been reduced to a reasonable magnitude. This is almost precisely the concept used in the “nested saturation” controller devised originally by Teel [30].

This approach has been extended by Morin to allow the results to be applied to a much more general class of systems, including general linear systems (which, as we noted above, are only homogeneous of order $\tau < 0$ in the chain of integrators case) [24]. Using a nonlinear rescaling theorem, Morin is able to rescale nonlinear controllers and achieve bounded controllers for a broad class of nonlinear systems as well. In order to apply the theorem, one must be given a one-parameter family of control laws and a one-parameter family of associated Lyapunov functions which satisfy certain stability and transversality properties.

2.3 Dynamic Rescaling of a Flight Control Experiment

In the previous section we described a method for rescaling a control law to allow bounded inputs. The approach consisted of two components: determine how to rescale the control law as a function of a single parameter and then determine the value of the parameter (based on the use of a Lyapunov function). In this section we demonstrate an approach for dynamically choosing the value of the parameter based on the commanded and actual inputs.

Consider a linear control system with magnitude and rate limits

$$\begin{aligned} \dot{x} &= Ax + B\sigma_M(u) \\ \dot{u} &= \sigma_N(v) \end{aligned} \tag{16}$$

where σ_M represents a magnitude saturation of magnitude M and σ_N represents a rate limit of magnitude N. Suppose we have a parameterized control law given by

$$u_c = K_\lambda(x - x_d) \tag{17}$$

where $\lambda \in \mathbb{R}$ is the rescaling parameter. As an example, for a chain of integrators, this control law could have the form derived above using homogeneous rescalings.

We now seek to choose the value of the rescaling parameter in a *dynamic* fashion, based on whether the control is saturating in either magnitude or rate. The basic idea is simple: if the control

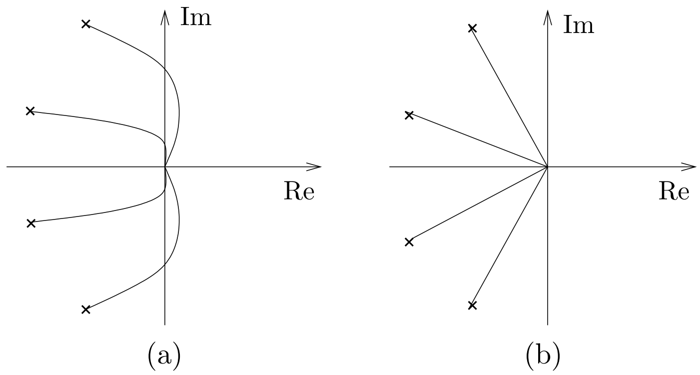

Figure 4: Root locus plots for rescaling control laws: (a) linear rescaling of gains for a chain of integrators; (b) nonlinear rescaling based on homogeneous properties.

law is saturating, reduce the "gain" of the system by decreasing λ. This is shown schematically in Figure 3. There are two obvious issues that must be addressed: (1) the one parameter family of control laws must be chosen such that for any fixed λ the system is stable and (2) the update law for λ must be chosen such that it does not just leave the system on the magnitude or rate limit (in which case there is no point to adjusting the control law).

To address these points, we restrict attention to a chain of integrators with a linear feedback control law. A naive approach to choosing the rescaling law for the gain would be to choose the control as

$$u = \lambda(k_1 x_1 + k_2 x_2 + \cdots + k_n x_n). \tag{18}$$

Note however, that for fixed λ the closed loop system is not necessarily stable, as shown in the root locus plot in Figure 4a. To remedy this, we might choose a rescaling rule that keeps the damping of the system constant, as shown in Figure 4b. This update law turns out to be precisely the homogeneous rescaling law from the previous section:

$$u = k_1 \lambda^n x_1 + k_2 \lambda^{n-1} x_2 + \cdots + k_n \lambda x_n. \tag{19}$$

To choose the value of the rescaling parameter, we use a variation of an automatic gain control algorithm originally reported in [32]. The basic idea is to rapidly decrease the gain whenever the commanded input, u_c, is not equal to the actual input (after saturation). Then, we *slowly* increase the gain when the input is no longer on one of its limits. The reason to slowly increase the gain is to insure that we do not simply sit on the magnitude and/or rate limit, in which case the automatic gain control logic has no real effect.

A specific control law of this form was derived for a chain of integrators by Lauvdal and reported in [17]. Let $h(t)$ be a piecewise continuous function of time given by

$$h(t) = 0 \quad \text{if} \quad u = u_c, \qquad \dot{h}(t) = 1 \quad \text{otherwise}. \tag{20}$$

where u and u_c are the actual and commanded input, respectively. Let T_k denote the time instance when $u = u_c$ for the kth time, $\lambda_k = \lambda(T_k)$, $\lambda_0 = 1$, N a positive number and $g(t) = (t - T_N)$.

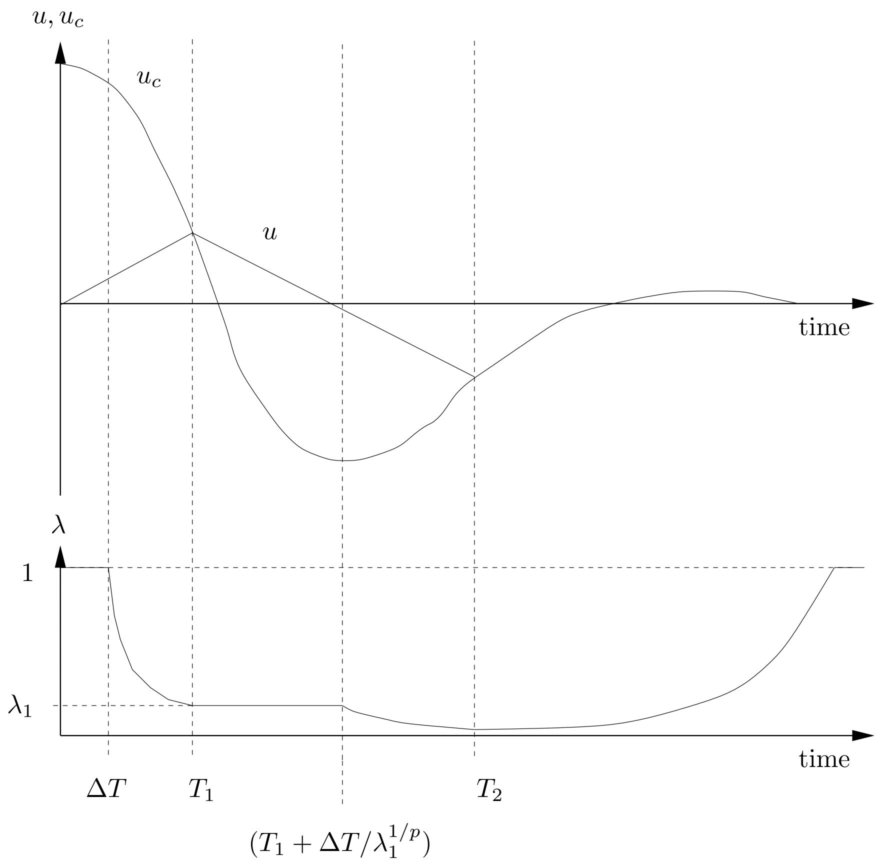

Figure 5: Sketch illustrating the evolution of the scaling factor when the input is saturating.

Define $\lambda(t)$ as

$$\lambda(t) = \begin{cases} \lambda_k & 0 < h(t) \leq \frac{\delta T}{\lambda_k^{1/p}} \\ \left(\frac{\delta T}{h(t)}\right)^p & k \leq N \quad \text{and} \quad h(t) > \frac{\delta T}{\lambda_k^{1/p}} \\ \left(\frac{\delta T}{g(t)}\right)^q & k > N \quad \text{and} \quad g(t) > \frac{\delta T}{\lambda_N^{1/q}} \end{cases} \tag{21}$$

where $p \geq 0$, $q > 1$ and $\delta T > 0$.

The complicated nature of this control law is due to the form of the proof of convergence, which can be found in [18]. The intuition behind this scaling feature is illustrated in Figure 5 and described more fully in [17]. Roughly, the control begins by waiting for a time δT, chosen to be less than the maximum amount of time the system can be saturated and still converge using a linear control action. This places an upper bound on the amount of time the controller will wait before decreasing the system gain. Once this limit is reached the control gain is rapidly decreased until time T_1, when the commanded input and the actual input are equal (and λ_1 is set to the resulting value of λ). At this point, $h(t)$ is reset to zero, and we again wait for some period of time before potentially further decreasing the control gain (at time $T_1 + \delta T/\lambda_1^{1/p}$). Then λ is decreased a second time and this process repeats until λ is small enough. It is shown in [18] that this iteration occurs a finite number of times for a step function and that the response of the system converges to the

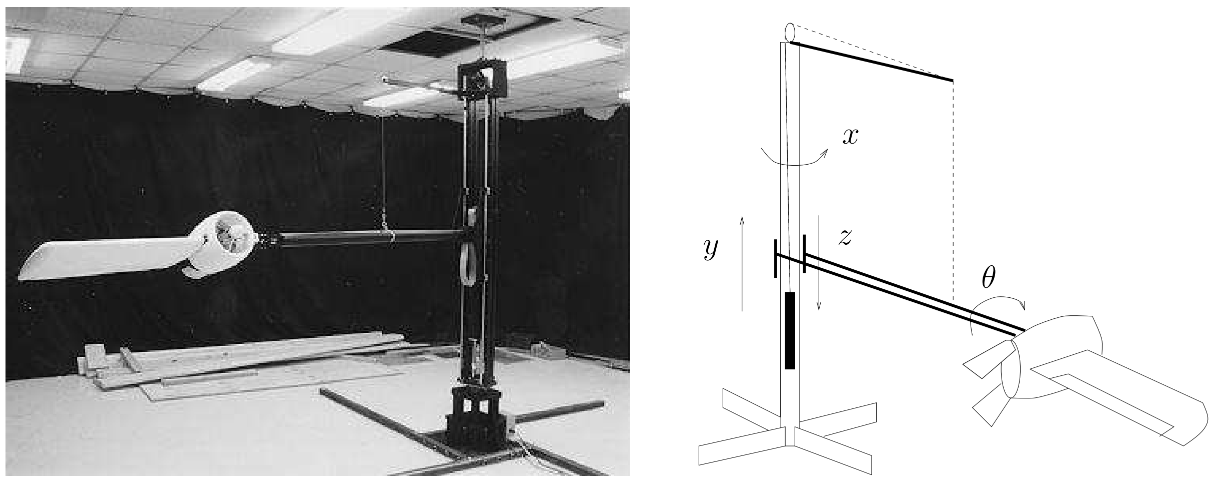

Figure 6: Thrust vectored aircraft on stand.

desired value.

Once the commanded an actual input match, λ should be slowly increased to return the system to its original performance level. It is important that λ be increased in a way that does not immediately saturate the input since this will simply cause the gain to be further decreased (according to the scaling law described above). The rule used in [17] is given by the formula

$$\lambda(t_{k+1}) = \kappa\lambda(t_k) \tag{22}$$

where t_k is the sampling time (in a discrete time implementation), $\kappa > 1$ is a free parameter and the increase is stopped when $\lambda(t) = 1$.

Although the particular form of this update law is quite messy (and apparently required in order to prove convergence), the performance of the control law is quite extraordinary, as illustrated in the following example.

Example 4 (Altitude control of the Caltech ducted fan)**.** The approach described above was implemented on a flight control experiment at Caltech, show in Figure 6. The system consists of a vectored thrust engine attached to a wing. The motor speed, engine flap angles, and wing flap angle can all be actuated and the position of the vehicle is sensed using optical encoders on the stand. The flight dynamics of the system approximate the longitudinal dynamics of an aircraft.

The dynamics of the system roughly decouple into the forward and vertical motion of the fan. In the vertical direction, the dynamics are approximated by a chain of four integrators: the wing flap generates a torque on the system, which changes the angle of attack (two integrators), and the angle of attack changes the lift coefficient, generating vertical motion (two integrators). Thus, we can roughly apply the techniques described above to the problem of stabilizing the vertical position of the system.

Artificial rate limits of 20 deg/sec were imposed on the wing flap and the system was commanded to change altitude. Figures 7–9 show the response of the system with and without rescaling of the input. Note that the responses in the 20 deg/sec rate limited case is almost as good as the case with no rate limits in place.

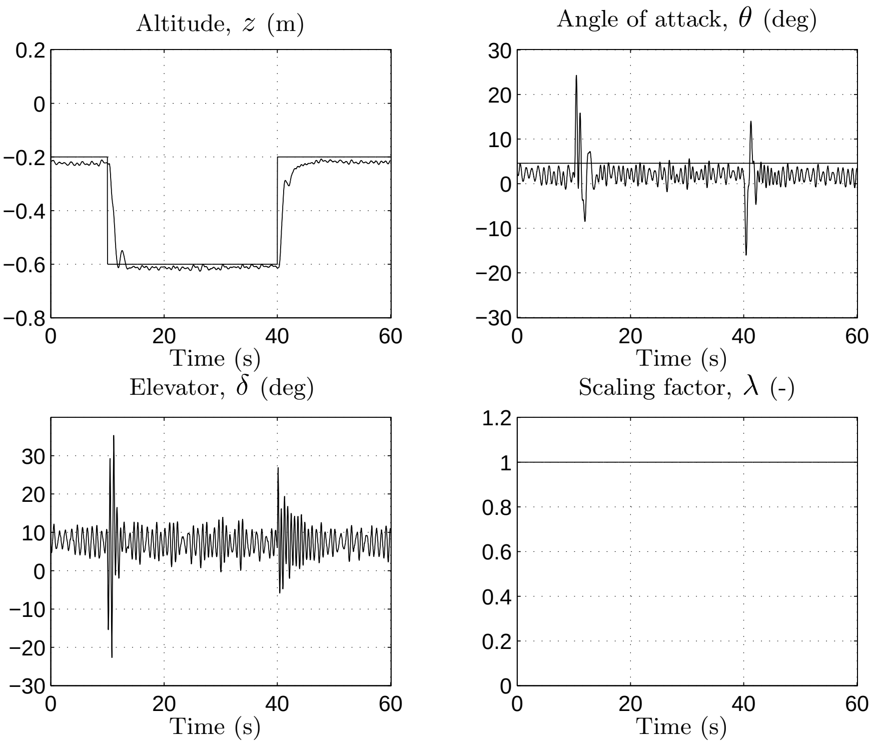

Figure 7: No rate limits.

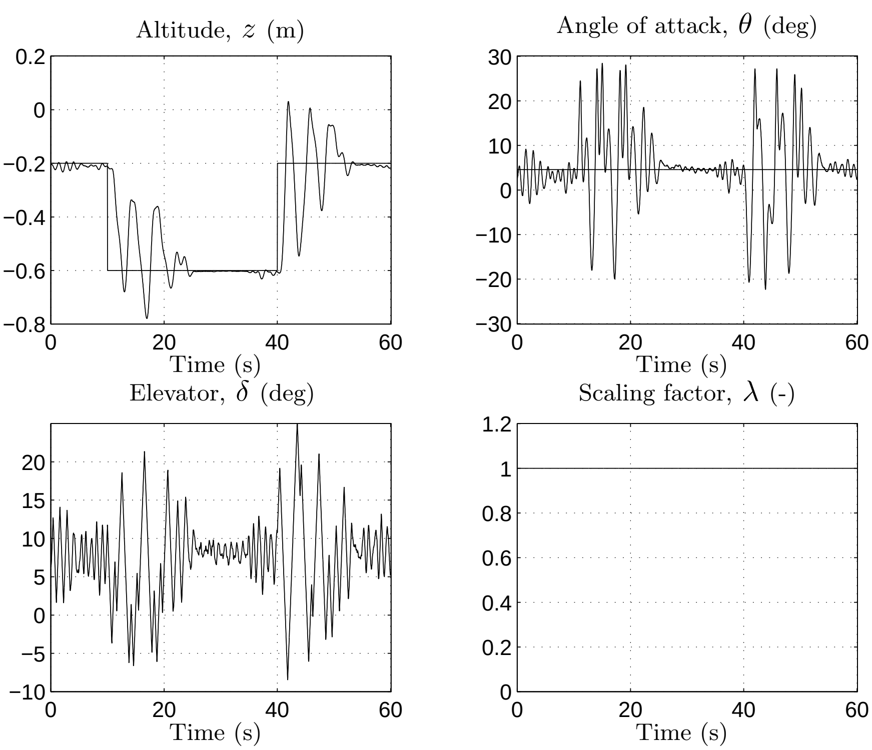

Figure 8: Rate limit; 20 (deg/s). No scaling of input.

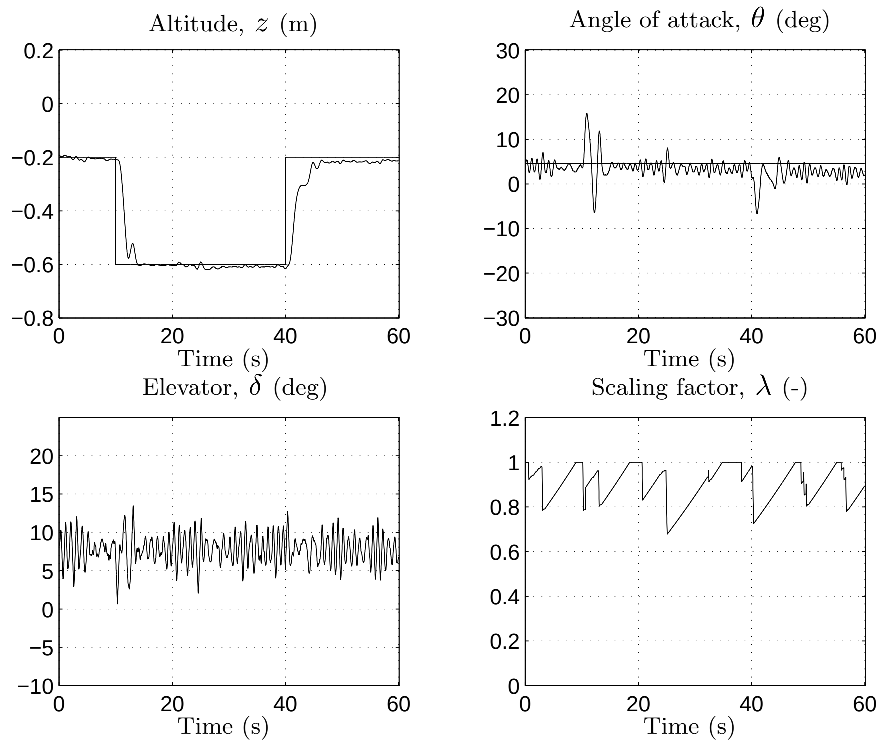

Figure 9: Rate limit; 20 (deg/s). With input scaling.

Extensions of this work have been explored in the thesis work of Lauvdal [16]. In particular, the results have been extended to general linear systems and have been applied to rudder-roll stabilization of ships.

3 Trajectory Generation Using Differential Flatness

In this section we consider the problem of generating *trajectories* for control systems in the presence of magnitude and rate saturations. We motivate the problem in terms of two degree of freedom design approaches and then make use of differential flatness to simplify the trajectory generation problem. Finally, we give some preliminary results in real-time trajectory generation in the presence of magnitude and rate constraints for linear systems.

3.1 Two degree of freedom control design

A large class of industrial and military control problems consist of planning and following a trajectory in the presence of noise and uncertainty. Examples range from unmanned and remotely piloted airplanes and submarines performing surveillance and inspection tasks, to mobile robots moving on factory floors, to multi-fingered robot hands performing inspection and manipulation tasks inside the human body under the control of a surgeon. All of these systems are highly nonlinear and demand accurate performance.

Modern geometric approaches to nonlinear control often rely on the use of feedback transformations to convert a system into a simplified form which can then be controlled with relatively

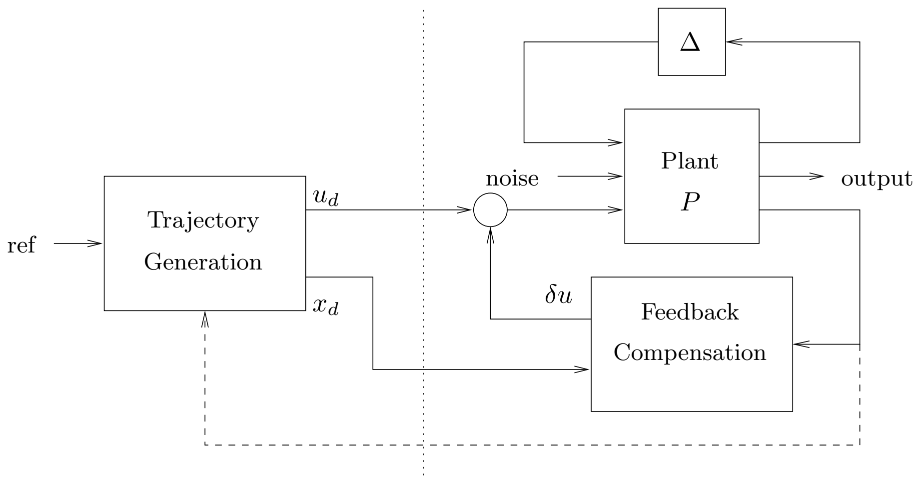

Figure 10: Two degree of freedom controller design.

standard techniques (such as linear feedback). While this approach very effectively exploits the nonlinear nature of the system, it often does so by "converting" the nonlinear system into a linear one. This can be disadvantageous if one is concerned with disturbance rejection and other performance specifications since the nonlinear transformations typically do not preserve many important properties of the system.

One way around this limitation is to make use of the notion of *two degree of freedom* controller design for nonlinear plants. Two degree of freedom controller design is a standard technique in linear control theory that separates a controller into a feedforward compensator and a feedback compensator. The feedforward compensator generates the nominal input required to track a given reference trajectory. The feedback compensator corrects for errors between the desired and actual trajectories. This is shown schematically in Figure 10.

Many modern nonlinear control methodologies can be viewed as synthesizing controllers which fall into this general framework. For example, traditional nonlinear trajectory tracking approaches, such as feedback linearization [12, 14] and nonlinear output regulation [13], are easily viewed as a feedforward piece and a feedback piece. Indeed, when the tracking error is small, the primary difference between the methods is the form of the error correction term: output regulation uses the linearization of the system about a single equilibrium point; feedback linearization uses a linear control law in an appropriate set of coordinates. It is important to note that these approaches rely on the availability of state feedback in order to generate feedforward commands. Thus, they are not traditional open loop feedforward controllers and this difference is crucial to their operation.

This two step approach can be carried one step further by completely decoupling the trajectory generation and asymptotic tracking problems. Given a desired output trajectory, we first construct a state space trajectory x_d and a nominal input u_d that satisfy the equations of motion. The error system can then be written as a time-varying control system in terms of the error, $e = x - x_d$. Under the assumption that that tracking error remains small, we can linearize this time-varying system about $e = 0$ and stabilize the $e = 0$ state.

The use of two degree of freedom techniques has been studied in a variety of applications. It is a relatively standard approach in classical robotics [26] and has also seen recent application in flexible robot systems [2, 8, 27]. Applications to flight control include the work of Meyer et al. [23]

and Martin et al. [20]. General theoretical results have been explored by Paden and Chen [5, 27] as well as ourselves [25, 35, 33]. It is also implicit in the output regulation problem studies by Isidori and Byrnes [13].

In this section we explore the problem of trajectory generation in the presence of magnitude and rate limits. We are motivated by flight control problems with a remote pilot, and hence we seek techniques that can be implemented in "real time", meaning that they are computational efficient and can be run as part of a online, closed loop implementation. The technique that we explore is the use of differential flatness, which is introduced in the next section.

3.2 Introduction to differential flatness

A system is said to be differentially flat if all of the feasible trajectories for the system can be written as functions of a flat output $z(\cdot)$ and its derivatives. In other words, given a nonlinear control system

$$\begin{aligned} \dot{x} &= f(x,u) \\ y &= h(x) \end{aligned} \tag{23}$$

we say the system is *differentially flat* if there exists a function $z(x, u, \dot{u}, \ldots, u^{(p)})$ such that all feasible solutions of the underdetermined differential equation (23) can be written as

$$\begin{aligned} x &= \alpha(z, \dot{z}, \ldots, z^{(q)}) \\ u &= \beta(z, \dot{z}, \ldots, z^{(q)}). \end{aligned} \tag{24}$$

Differentially flat systems were originally studied by Fliess et al. in the context of differential algebra [10] and later using Lie-Backlünd transformations [11]. In [33] we reinterpreted flatness in a differential geometric setting. We made extensive use of the tools offered by exterior differential systems and the ideas of Cartan. Using this framework we were able to recover most of the results currently available using the differential algebraic formulation and achieve a deeper geometric understanding of flatness. We also showed that differential flatness is more general than feedback linearization in the multi-input case. More importantly, the point of view is quite different, focusing on trajectories rather than feedback transformations. See [25] for a description of the role of flatness in control of mechanical systems and [34, 35] for more information on flatness applied to flight control systems.

Example 5 (Linear systems)**.** All linear systems are differentially flat, with the flat outputs corresponding to the top of the integrator chain(s) when the system is put into controller canonical form. It is clear that given this set of outputs, one can determine the states of the system as well as the inputs by repeated differentiation of the flat outputs. This example also makes clear that all full state, feedback linearizable systems are flat (with the linearizing output corresponding to the flat output).

Example 6 (Planar ducted fan)**.** As a nonlinear example, consider the dynamics of a planar, vectored thrust flight control system as shown in Figure 11. This system consists of a rigid body with body fixed forces and is a simplified model for the Caltech ducted fan described in Section 2.3. Let (x, y, θ) denote the position and orientation of the center of mass of the fan. We assume that the forces acting on the fan consist of a force f_1 perpendicular to the axis of the fan acting at a distance r from the center of mass, and a force f_2 parallel to the axis of the fan. Let m be the

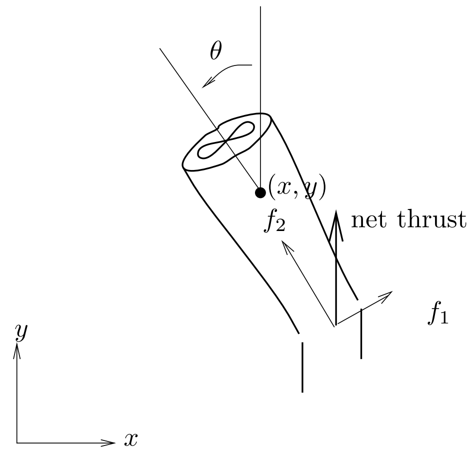

Figure 11: Planar ducted fan engine. Thrust is vectored by moving the flaps at the end of the duct.

mass of the fan, J the moment of inertia, and g the gravitational constant. We ignore aerodynamic forces for the purpose of this example.

The dynamics for the system are

$$\begin{aligned} m\ddot{x} &= f_1 \cos\theta - f_2 \sin\theta \\ m\ddot{y} &= f_1 \sin\theta + f_2 \cos\theta - mg \\ J\ddot{\theta} &= r f_1. \end{aligned} \tag{25}$$

Martin et al. [20] showed that this system is flat and that one set of flat outputs is given by

$$\begin{aligned} z_1 &= x - (J/mr)\sin\theta \\ z_2 &= y + (J/mr)\cos\theta. \end{aligned} \tag{26}$$

Using the system dynamics, it can be shown that

$$\ddot{z}_1 \cos\theta + (\ddot{z}_2 + g)\sin\theta = 0. \tag{27}$$

and thus given $z_1(t)$ and $z_2(t)$ we can find $\theta(t)$ except for an ambiguity of π and away from the singularity $\ddot{z}_1 = \ddot{z}_2 + g = 0$. The remaining states and the forces $f_1(t)$ and $f_2(t)$ can then be obtained from the dynamic equations, all in terms of z_1, z_2, and their higher order derivatives.

Having determined that a system is flat, it follows that all feasible trajectories for the system are characterized by the evolution of the flat outputs. Using this fact, we can convert the problem of point to point motion generation to one of finding a curve $z(\cdot)$ which joins an initial $z(0), \dot{z}(0), \ldots, \dot{z}^{(q)}(0)$, corresponding to the initial state, to a final set of values for the same quantities, corresponding to the final state. In this way, we reduce the problem of generating a feasible trajectory for the system to a classical algebraic problem in interpolation. Similarly, problems in trajectory generation can also be converted to problems involving curves $z(\cdot)$ and algebraic methods can be used to provide real-time solutions [34, 35].

Thus, for differentially flat systems, trajectory generation can be reduced from a dynamic problem to an algebraic one. Specifically, one can parameterize the flat outputs using basis functions

$\phi_i(t)$,

$$z = \sum a_i \phi_i(t), \tag{28}$$

and then write the feasible trajectories as functions of the coefficients a:

$$\begin{aligned} x_d &= \alpha(z, \dot{z}, \ldots, z^{(q)}) = x_d(a) \\ u_d &= \beta(z, \dot{z}, \ldots, z^{(q)}) = u_d(a). \end{aligned} \tag{29}$$

Note that no ODEs need to be integrated in order to compute the feasible trajectories (unlike optimal control methods, which involve parameterizing the *input* and then solving the ODEs). This is the defining feature of differentially flat systems. The practical implication is that nominal trajectories and inputs which satisfy the equations of motion for a differentially flat system can be computed in a computationally efficient way (solution of algebraic equations).

3.3 Real-time trajectory generation for linear systems with input constraints

If input constraints are present for a differentially flat system, these constraints can be written in terms of the coefficients of the basis functions. In particular, for a magnitude constraint $|u| < M$, the constraints have the form

$$|u(z, \dot{z}, \ldots, z^{(q)})| < M \tag{30}$$

and substituting

$$z = \sum a_i \phi_i(t), \tag{31}$$

gives a nonlinear, algebraic constraint on the coefficients a_i.

To illustrate how this problem might be attacked, we restrict ourselves to the case of a single input, linear system

$$\begin{aligned} \dot{x} &= Ax + Bu \\ z &= Fx, \end{aligned} \tag{32}$$

where z is the flat output that corresponds to the end of the integrator chain if the system were placed in controller canonical form. Since the initial and final state can be written in terms of z and it's derivatives, the endpoint constraints become

$$\begin{bmatrix} z(0) \\ \dot{z}(0) \\ \vdots \\ z^{(q)}(0) \\ z(T) \\ \dot{z}(T) \\ \vdots \\ z^{(q)}(T) \end{bmatrix} = \begin{bmatrix} \phi_1(0) & \phi_2(0) & \cdots & \phi_m(0) \\ \dot{\phi}_1(0) & \dot{\phi}_2(0) & \cdots & \dot{\phi}_m(0) \\ \vdots & & & \vdots \\ \phi_1^{(q)}(0) & \phi_2^{(q)}(0) & \cdots & \phi_m^{(q)}(0) \\ \phi_1(T) & \phi_2(T) & \cdots & \phi_m(T) \\ \dot{\phi}_1(T) & \dot{\phi}_2(T) & \cdots & \dot{\phi}_m(T) \\ \vdots & & & \vdots \\ \phi_1^{(q)}(T) & \phi_2^{(q)}(T) & \cdots & \phi_m^{(q)}(T) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_m \end{bmatrix} = Ma \tag{33}$$

The dimension of the matrix M depends on the number of basis functions and this will generally be chosen to be more than the minimum number required to make M full rank.

Similarly, the constraints on the input can be written as

$$G_0 z + G_1 \dot{z} + \cdots + G_p z^{(q)} + d < 0 \tag{34}$$

where we use the fact that u is a linear function of z and its derivatives, and rewrite the constraint $|u| < M$ as two constraints, $u < M$ and $u > -M$. Similar constraints can be written for rate limits and the matrices G_i and vector d are obtained by stacking these constraints. We now substitute the expansion z in terms of basis functions to obtain a constraint on the coefficients of the form

$$P(t)a < d. \tag{35}$$

Notice that these constraints must hold for each instant in time.

Pulling these equations together, we must find a set of coefficients that satisfy

$$\begin{aligned} Ma &= c \\ P(t)\alpha &< d. \end{aligned} \tag{36}$$

This example immediately illustrates the difficulty in solving the trajectory generation problem in the presence of input constraints. While the endpoint constraints generate a finite number of conditions on the basis function coefficients, the input limits generate constraints at each instant in time. One cannot *a priori* guarantee that these constraints can be satisfied with a finite number of basis functions for a given set of end conditions. Assuming that a solution exists, one is still left with the problem of finding a solution that satisfies the constraints.

One approach to such a solution is to attempt to place bounds on the inequality constraints in a way that allows a finite number of conditions to be checked. For example, if we choose basis functions that are monotonically increasing (eg, powers of t) then the constraint might be satisfied by choosing all coefficients to have the same sign. It is easy to see, however, that this will not work for two-sided constraints generated by limits of the form $|u| < M$.

A second approach is to approximate the constraints by evaluating them at an array of times, thus converting them to a (potentially large) set of affine constraints of the form $P(t_i)a < d$ where t_i represents the sampling time. This is the approach taking in [1], which we describe in more detail below. By choosing basis functions with sufficiently bounded derivatives, it is possible to enforce slightly more restrictive limits and give bounds on the spacing of points that is required to insure that the constraints are satisfied between sampling instants.

An additional advantage of the sampling approach is that the constraints have a particularly nice representation in the coefficient space. We first rewrite our flat solution as

$$z = \Phi_0(t) + \sum a_i \phi_i(t) \tag{37}$$

where Φ_0 is chosen to satisfy the end conditions and ϕ_i are chosen so that the function plus q derivatives is identically zero at $t = 0$ and $t = T$. Then the resulting input constraints are of the form $P(t_i)a < N(t_i)$ where $N(t_i)$ contains terms generated by Φ_0 and its derivatives. A solution to this constraint can be found by looking at the hyperplanes given by $P(t_i)a - N(t_i) = 0$ and then looking for regions which are on the proper side of each hyperplane. This is shown in Figure 12 for the case of two free parameters.

To illustrate the efficacy of this approach, it was implemented on an experimental system briefly described in the following example.

Example 7 (Pursuit-evasion experiment [1])**.** S. Agrawal and N. Faiz have implemented the method outlined above on a mass spring damper experiment (see Figure 13). The system consists of three masses, with a single motor acting on the first mass and the first and second mass connected by

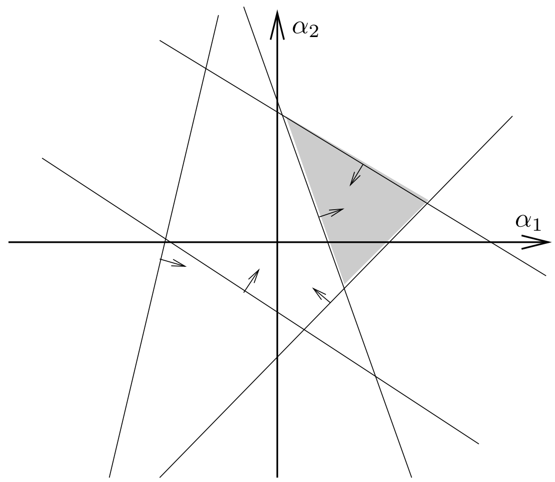

Figure 12: Feasible regions in parameter space. Each constraint on the parameters is represented by a hyperplane in parameter space and the interior of the corresponding convex set gives the feasible parameters.

springs. A third mass is disconnected as moved manually to provide a "target" for the system. Using the position of the third mass as input, a computer control system computes a feasible trajectory for the rest of the system that attempts to track the position of the third mass using the second mass. The constraints on the system are the physical positions of the second and third masses, as well as the input magnitude on the first mass. More details on the experiment are given in [1].

The collocation technique describe above was implemented, using polynomials for basis functions. Due to computational limits, 6 constraints (3 two-sided constraints) with 3 collocation points and 2 free modes were used. The position of the third mass was computed every second and a feasible trajectory was computed and used as an input to the linear control law. Figure 14 shows the results of a typical run. The six constraints are satisfied by the desired trajectory for masses 1 and 2 at all times.

4 Conclusions and Future Work

The problem of control design in the presence of magnitude and rate limits is one which has received a large amount of attention over the years but there are still very few design oriented techniques for analyzing such systems. In this paper we have focused on two specific problems, stabilization and trajectory generation, and used tools from differential geometry to study some particularly simple special cases, such as chains of integrators and linear systems. While these approaches do not cover the wide variety of practical problems that are faced in applications, they do provide interesting insights into some of the difficulties and possible approaches to control design in the presence of magnitude and rate limits.

For the problem of saturation, our approach is to design "nonlinear wrappers" that modify an existing control law to operate in the presence magnitude and/or rate limits. When the system is operating in a region that does not push the inputs against the constraints, the wrapper function performs no action and the nominal control law is used. When the system begins to saturate, the

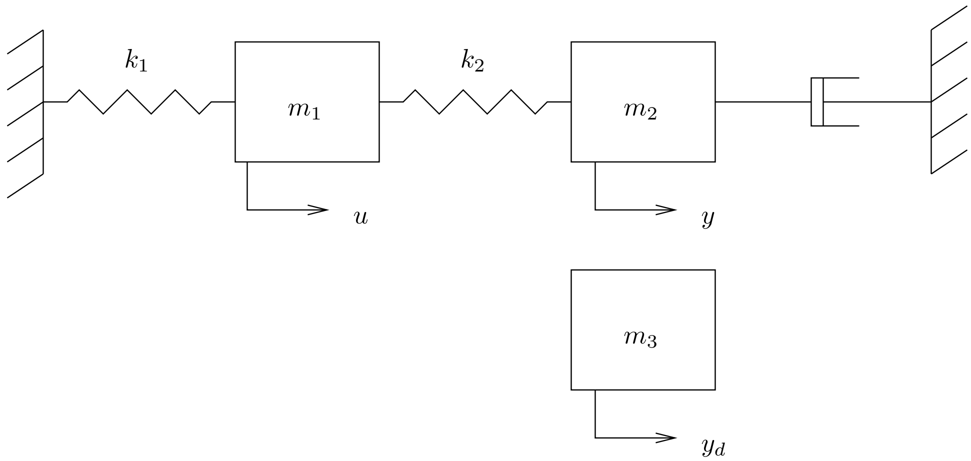

Figure 13: Pursuit-evasion experiment. Mass 2 is used to generate a reference trajectory which is tracked by mass 3.

wrapper modifies the inputs and outputs of the control law such that the desired control action is achieved without violating the constraints or causing the system to go unstable.

The form of the nonlinear wrapper is based on rescaling the control law using a nonlinear gain function. Using tools from homogeneous systems, we derived a rescaling procedure that maintained stability of the system while giving inputs with bounded controls. For the specific case of a chain of integrators, this gain function has the effect of minimizing the size of errors in states far away from the input in order to first stabilize the states near the input. This is the same type of approach as Teel's "nested saturation" framework. The approach can be generalized to non-homogeneous systems subject to finding families of control laws and Lyapunov functions that satisfy the conditions of the rescaling theorem.

With this rescaling approach as motivation, we then explored a "dynamic rescaling" technique for choosing the nonlinear gain parameter. The main idea was to select the value of the scaling parameter based on the difference between the actual and commanded input values. Once the gain had been reduced to a level that allowed the system to stop saturating, the gain was slowly increased towards its original value to allow the performance of the system to be maintained. Although the form of the control law and its proof of convergence are cumbersome, the experimental behavior of the approach is very encouraging.

Finally, we considered the problem of trajectory generation in the presence of magnitude and rate limits for differentially flat systems. For this class of systems, the trajectory generation problem can be reduced to a spline problem with the input constraints appearing as nonlinear algebraic conditions on the coefficients. We presented one specific approach to solving the constrained algebraic problem for the special case of linear systems, where the constraints become hyperplanes in the spline coefficient space.

By combining the problem of trajectory generation and stabilization, it is possible to build system that achieve high performance trajectory tracking in the presence of both magnitude and rate constraints. The real-time trajectory generation algorithm generates state and input trajectories that simultaneously satisfy the dynamics, input constraints, and output requirements. The inner loop then stabilizes along this trajectory and handles additional saturations that might occur due

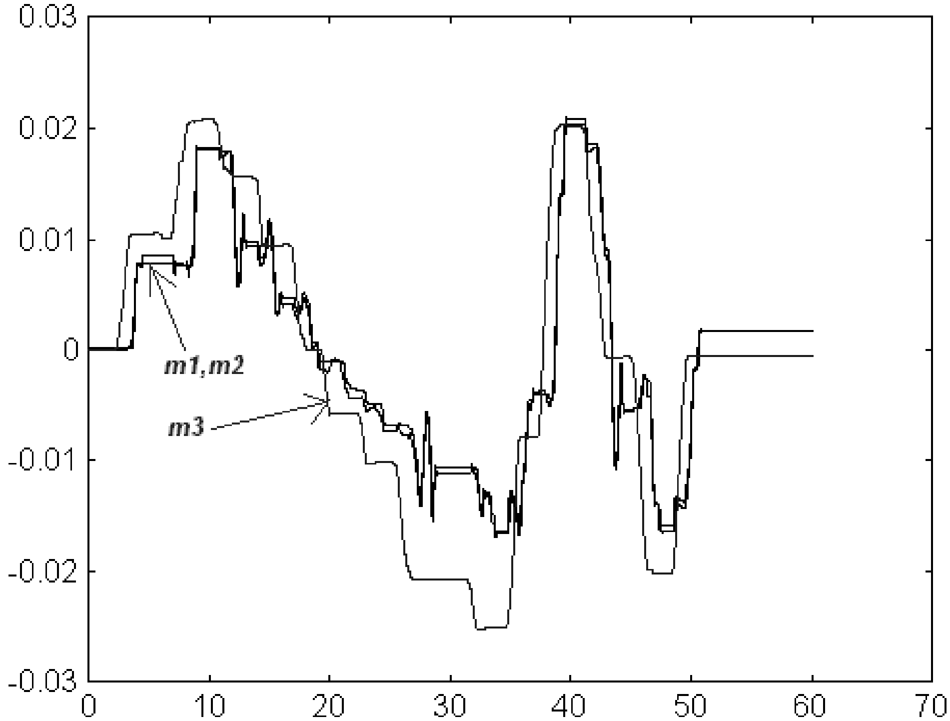

Figure 14: Real-time trajectory generation for the pursuit-evasion experiment.

to system uncertainties such as noise, parametric errors, or unmodeled dynamics.

The work presented in this paper is only a start on a more complete geometric theory for nonlinear control systems in the presence of magnitude and rate constraints. The notion of combining nonlinear wrappers with dynamic rescaling remains appealing (and performs extremely well), but stronger analytical tools are required to apply these techniques to a broader class of systems and reduce the gap between theory and practice. Our experimental results hint that this approach may be particularly effective in practical systems. Most of the current literature is based on static rescaling (similar the the results we presented for homogeneous systems) and these seem to be quite conservative compared with our approach.

Similarly, more powerful techniques for generating trajectories are needed to handle the new systems that are being planned, such as remotely operated systems operating underwater, on land, in the air, and in space. One approach which seems particularly appealing is the use of basis functions that are tuned for the specific system and task. In particular, if one can show that by choice of basis function it is possible to simplify the computations required for online implementation, this would have important implications for highly aggressive, real-time trajectory generation. Choice of basis function might also be used to provide more rigorous approaches to the surface allocation problem in flight control.

Finally, we note that although we have focused here on motion control problems, the role of magnitude and rate constraints is also of considerable important in many other classes of problems. In particular, actuation limits appear to be a primary driver in active control of fluid systems (see [37] for an example in the context of compression systems). These limits are both temporal limits (magnitude, bandwidth, and rate) as well as spatial limits (the number and spacing of actuators in a fluidic systems, for example). Preliminary work in the role of magnitude and rate constraints in control of bifurcations [36] shows some of the possible techniques for approaching this interesting class of problems.

Acknowledgements

The results described here are based on joint work with Trygve Lauvdal, Pascal Morin, Michiel van Nieuwstadt, Sunil Agrawal, Nadeem Faiz, Oliver Kaiser, Mark Milam, and Herbert Streumper. The author would also like to thank Andrew Teel for frequent interactions and discussions in this area.

References

[1] S. K. Agrawal, N. Faiz, and R. M. Murray. Feasible trajectories of linear dynamic systems with inequality constraints using higher-order representations. In *Proc. IFAC World Congress*, 1999.

[2] E. Bayo and B. Paden. On trajectory generation for flexible robots. *Journal of Robotic Systems*, 4(2):229–235, 1991.

[3] S. P. Boyd and C. H. Barratt. *Linear Controller Design: Limits of Performance*. Prentice-Hall, 1991.

[4] J. J. Buchholz. Time delay induced by control surface rate saturation. *Z. Flugwiss. Weltraumforsch*, 17:287–293, 1993.

[5] D. G. Chen and B. Paden. Stable inversion of nonlinear nonminimum-phase systems. *International Journal of Control*, 64(1):81–97, 1996.

[6] M. A. Dahleh and I. J. Diaz-Bobillo. *Control of Uncertain Systems: A Linear Programming Approach*. Prentice Hall, 1995.

[7] M. A. Dahleh and J. B. Pearson. L_1-optimal compensators for continuous-time systems. *IEEE Transactions on Automatic Control*, 32(10):889–895, 1987.

[8] S. Devasia and E. Bayo. Inverse dynamics of articulated flexible structures—Simultaneous tracking and vibration reduction. *Dynamics and Control*, 4(3):299–309, 1994.

[9] M. A. Dornheim. Report pinpoints factors leading to YF-22 crash. *Aviation Week and Space Technology*, November 1992.

[10] M. Fliess, J. Levine, P. Martin, and P. Rouchon. On differentially flat nonlinear systems. *Comptes Rendus des Séances de l'Académie des Sciences*, 315:619–624, 1992. Serie I.

[11] M. Fliess, J. Levine, P. Martin, and P. Rouchon. Linéarisation par bouclage dynamique et transformations de Lie-Bäcklund. *Comptes Rendus des Séances de l'Académie des Sciences*, 317(10):981–986, 1993. Serie I.

[12] L. R. Hunt, R. Su, and G. Meyer. Global transformations of nonlinear systems. *IEEE Transactions on Automatic Control*, AC-28(1):24–31, 1983.

[13] A. Isidori and C. I. Byrnes. Output regulation of nonlinear systems. *IEEE Transactions on Automatic Control*, 35(2):131–140, 1990.

[14] B. Jakubczyk and W. Respondek. On linearization of control systems. *Bulletin de L'Academie Polonaise des Sciences, Série des sciences mathématiques*, XXVIII:517–522, 1980.

[15] M. Kawski. Homogeneous stabilizing feedback laws. *Control—Theory and Advanced Technology*, 6(4):497–516, 1990.

[16] T. Lauvdal. *Stabilization of Linear Systems with Input Magnitude and Rate Saturations*. PhD thesis, Norwegian University of Science and Technology, Department of Engineering Cybernetics, 1998.

[17] T. Lauvdal and R. M. Murray. Stabilization of a pitch axis flight control experiment with input rate saturation. In *AIAA Conference on Guidance, Navigation, and Control*, 1997.

[18] T. Lauvdal, R. M. Murray, and T. I. Fossen. Stabilization of integrator chains in the presence of magnitude and rate saturations: A gain scheduling approach. In *Proc. IEEE Control and Decision Conference*, 1997.

[19] Z. Lin. Semi-global stabilization of linear systems with position and rate-limited actuators. *Systems and Control Letters*, 30(1):1–11, 1997.

[20] P. Martin, S. Devasia, and B. Paden. A different look at output tracking—Control of a VTOL aircraft. *Automatica*, 32(1):101–107, 1994.

[21] R. T. M'Closkey and R. M. Murray. Exponential stabilization of driftless nonlinear control systems using homogeneous feedback. *IEEE Transactions on Automatic Control*, 42(5):614–628, 1997.

[22] A. Megretski. l_2 BIBO output feedback stabilization with saturated control. In *Proc. IFAC World Congress*, 1996.

[23] G. Meyer, L. R. Hunt, and R. Su. Nonlinear system guidance in the presence of transmission zero dynamics. Technical report, NASA, 1994. Preprint.

[24] P. Morin, R. M. Murray, and L. Praly. Nonlinear rescaling of control laws with application to stablization in the presence of magnitude saturation. In *IFAC Symposium on Nonlinear Control Systems Design (NOLCOS)*, 1998.

[25] R. M. Murray. Nonlinear control of mechanical systems: A Lagrangian perspective. *Annual Reviews in Control*, 21:31–45, 1997.

[26] R. M. Murray, Z. Li, and S. S. Sastry. *A Mathematical Introduction to Robotic Manipulation*. CRC Press, 1994.

[27] B. Paden, D. Chen, R. Ledesma, and E. Bayo. Exponentially stable tracking control for multi-joint flexible-link manipulators. *ASME Journal of Dynamic Systems, Measurement and Control*, pages 1–17, 1991.

[28] A. Teel, O. Kaiser, and R. M. Murray. Uniting local and global controllers for the Caltech ducted fan. In *Proc. American Control Conference*, pages 1539–1543, 1997.

[29] A. R. Teel. *Feedback Stabilization: Nonlinear Solutions to Inherently Nonlinear Problems*. PhD thesis, University of California, Berkeley, 1992.

[30] A. R. Teel. Global stabilization and restricted tracking for multiple integrators with bounded controls. *Systems and Control Letters*, 18:165–171, 1992.

[31] A. R. Teel and N. Kapoor. Uniting local and global controllers. In *Proc. European Control Conference*, 1997.

[32] J. Van Amerongen and P. G. M. Van der Klugt añd H. R. Van Nauta Lempke. Rudder Roll Stabilization for Ships. *Automatica*, AUT-26(4):679–690, 1990.

[33] M. van Nieuwstadt, M. Rathinam, and R. M. Murray. Differential flatness and absolute equivalence. *SIAM Journal of Control and Optimization*, 36(4):1225–1239, 1998.

[34] M. J. van Nieuwstadt and R. M. Murray. Rapid hover to forward flight transitions for a thrust vectored aircraft. *Journal of Guidance, Control, and Dynamics*, 21(1):93–100, 1998.

[35] M. J. van Nieuwstadt and R. M. Murray. Real time trajectory generation for differentially flat systems. *International Journal of Robust and Nonlinear Control*, 8(11):995–1020, 1998.

[36] Y. Wang and R. M. Murray. Effects of magnitude saturation in control of bifurcations. In *Proc. IFAC World Congress*, 1999.

[37] C.-H. Yeung, Y. Wang, and R. M. Murray. Evaluation of bleed valve rate requirements in nonlinear control of rotating stall. *Journal of Propulsion and Power*, 15(5):781–791, 2000.