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

Recovery Guidance Satisfying Input and State Constraints: Rate Saturating Actuator Example

Santosh Devasia* and George Meyer†
NASA Ames Research Center, Moffett Field, California 94035-1000

When a fixed regulator is used to stabilize a desired state trajectory (guidance law), large initial tracking errors can lead to input saturation, which can result in performance deterioration. A method to modify the guidance law for recovering state-trajectory tracking (without violating input and state constraints) is presented. Although it may be feasible to find such a satisfactory recovery guidance maneuver for a given initial tracking error, online development of recovery guidance maneuvers may not be computationally tractable. A technique is developed to use precomputed recovery guidance maneuvers (precomputed for a finite set of initial tracking errors) to generate recovery guidance maneuvers for other initial errors. The technique is applied to an illustrative example with a rate-limited servoactuator, and simulation results are presented.

Nomenclature

E	= set of initial errors for which recovery guidance is precomputed
$\boldsymbol{e}(\cdot)$	= trajectory tracking error, $\boldsymbol{x}(\cdot) - \boldsymbol{x}_{\text{ref}}(\cdot)$
K	= state-trajectory regulator's gain (feedback gain)
$\boldsymbol{u}(\cdot)$	= input to the system
$\boldsymbol{u}_{\text{fb}}(\cdot)$	= feedback part of the input $\boldsymbol{u}$
$\boldsymbol{u}_{\text{ff}}(\cdot)$	= exact-tracking feedforward part of the input $\boldsymbol{u}$
$V_{(\alpha,\beta)}$	= set of acceptable initial errors, characterized by the vectors $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$
X	= set of initial conditions for which recovery guidance is precomputed
$\boldsymbol{x}$	= state of the system
$\boldsymbol{x}_{\text{mod}}$	= modification (during recovery guidance) of the reference state trajectory $\boldsymbol{x}_{\text{ref}}(\cdot)$
$\boldsymbol{x}_{\text{ref}}(\cdot)$	= exact-tracking reference state trajectory
$\boldsymbol{y}$	= system output
$\boldsymbol{y}_d(\cdot)$	= desired output trajectory
δ_f	= output of rate-limited servoactuator

I. Introduction

INVERSION of system dynamics can be used for precision output tracking control, for example, in aircraft control[1,2] and in air traffic control (ATC).[3] For example, in automated flight management systems, tracking of an accepted clearance (flight plan cleared by the ATC) can be achieved by the following. First, the clearance is converted to an executable reference flight path (desired output trajectory).[4] Second, this flight path is converted into a guidance law (exact output-tracking input-state trajectory) found, for example, by using an inversion-based approach.[5–7] Third, flight-path tracking is achieved by stabilizing the guidance law by using a trajectory regulator (see Fig. 1). However, relatively large tracking errors (or a large external disturbance that can be modeled as a tracking error) can lead to saturation of the inputs and can lead to the violation of state constraints. Such constraint violations can lead to loss of control and eventual loss of output tracking. The present work addresses the issue of modifying the guidance law to recover output tracking (from large tracking errors) without violating input and state constraints. The recovery guidance problem is addressed in the framework of general linear systems, and an example is provided to illustrate the technique.

A. Problem Description

The control scheme studied (shown in Fig. 1) is composed of a feedforward part and a feedback part. One approach to the design of such a combined feedforward/feedback controller is the stochastic optimal feedforward feedback technology (SOFFT) developed at NASA Langley Research Center,[8] where the pilot's commands are filtered by the feedforward controller and a command (guidance) is generated for the feedback part to regulate itself against. A related approach is to invert the system dynamics to find the guidance law for a desired output trajectory, see, for example, Refs. 1, 3, 5, 6, and 9. A significant problem with these approaches is that relatively large initial tracking errors can lead to violation of actuator and state constraints that can lead to performance degradation. This problem becomes significant if high gains are used in the feedback regulator to achieve tighter trajectory tracking. With high gains, the feedback regulator can only operate for a limited range of tracking errors and large tracking errors can cause saturation effects, which lead to performance deterioration. Such large tracking errors can be caused by a large external disturbance (modeled as a tracking error). Large initial tracking errors can also be caused when switching through a sequence of flight modes, for example, when transferring from a level-flight mode to an instrument-landing mode. The goal of the present paper is to study the recovery of trajectory tracking from large tracking errors for a given feedback regulator. Thus, the approach augments existing feedforward/feedback controllers, like SOFFT,[8] by enlarging the range of initial tracking errors from which trajectory tracking can be recovered.

B. Relation to Prior Work

The problem of controlling systems with actuator and state constraints has been extensively studied in the literature, see, for example, Refs. 10–12 and the references therein. The problem of output tracking for systems with constraints has also been addressed.[13] For applications to aircraft manual flight control, see Ref. 14 and the references therein. All these approaches require access to the feedback regulator (K in Fig. 1). In contrast, the present work develops an alternate approach that does not modify the regulator per se, but effectively bypasses it through recovery guidance maneuvers that modify the primary guidance law. Such an approach that does not modify the regulator is important to systems where the regulator is not easily accessible to modifications. For example, in the design of flight management systems, modifications of the regulator can require extensive testing and recertification, whereas the proposed open-loop recovery guidance can be (relatively) easily incorporated into the software.

Received Sept. 8, 1997; revision received Feb. 14, 1998; accepted for publication March 4, 1998.

*Visiting Research Associate, Human and Systems Technologies Branch; currently Assistant Professor, Mechanical Engineering Department, University of Utah, 50 S, Campus Center Drive, Room 2202, Salt Lake City, UT 84106. E-mail: santosh@me.utah.edu. Member AIAA.

†Aerospace Engineer, Human and Systems Technologies Branch, MS 262-2. E-mail: gmeyer@mail.arc.nasa.gov.

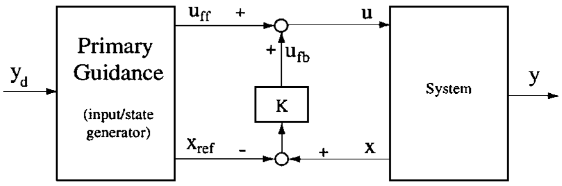

Fig. 1 Control scheme.

Other approaches prevent constraint violations by using command limiters and command rate limiters (see, for example, Ref. 15). However, these techniques can lead to large tracking errors. It is also possible to modify the guidance law by designing less-aggressive maneuvers that recover tracking without violating state and input constraints. Such recovery maneuvers tend to depend on the particular initial trajectory tracking error. Additionally, the selection (design) of a recovery maneuver tends to require extensive simulations and iterative redesigns. Further, the recovery guidance maneuver has to be redesigned for each different initial tracking error. Thus, although recovery guidance can be precomputed for a given initial error, the real-time design of a recovery maneuver may not be computationally tractable. The main result of the current paper is the development of a methodology that does not require online computation of the recovery guidance maneuvers; rather, in the spirit of the work by Mayne and Schroeder,[16] the recovery guidance maneuver is generated as the linear combination of precomputed recovery maneuvers.

C. Format of the Paper

We begin with a description of the guidance scheme in Sec. II, followed by the development of recovery guidance maneuvers from precomputed maneuvers in Sec. III. An optimal approach to develop recovery guidance maneuvers for a given initial tracking error is presented in Sec. IV, followed by an illustrative example and simulation results in Sec. V. Our conclusions are in Sec. VI.

II. Primary Guidance Scheme

Let the system be described by

$$\dot{\boldsymbol{x}}(t) = A\boldsymbol{x}(t) + B\boldsymbol{u}(t), \qquad \boldsymbol{y}(t) = C\boldsymbol{x}(t) \tag{1}$$

where $\boldsymbol{x}(t) \in \Re^n$ is the state, $\boldsymbol{u}(t) \in \Re^q$ is the input, and $\boldsymbol{y}(t) \in \Re^l$ is the output. Additionally, let a state feedback $\boldsymbol{u}(t) = K\boldsymbol{x}(t)$ be given such that the closed-loop system is stable, i.e., $A + BK$ is Hurwitz. In all of the following discussions, the feedback K is kept constant.

Given a desired output trajectory $\boldsymbol{y}_d(\cdot)$, the primary guidance scheme (see Fig. 1) finds a nominal input-state trajectory $[\boldsymbol{u}_{\text{ff}}(\cdot), \boldsymbol{x}_{\text{ref}}(\cdot)]$ that satisfies the system equations (1) and yields the desired output exactly, i.e.,

$$\dot{\boldsymbol{x}}_{\text{ref}}(t) = A\boldsymbol{x}_{\text{ref}}(t) + B\boldsymbol{u}_{\text{ff}}(t), \qquad \boldsymbol{y}_d(t) = C\boldsymbol{x}_{\text{ref}}(t) \qquad \forall t \in (-\infty, \infty) \tag{2}$$

Next, the exact output yielding state trajectory $\boldsymbol{x}_{\text{ref}}(\cdot)$ is stabilized by using state feedback (see Fig. 1). The control law is

$$\boldsymbol{u}(t) = \boldsymbol{u}_{\text{ff}}(t) + K[\boldsymbol{x}(t) - \boldsymbol{x}_{\text{ref}}(t)] \tag{3}$$

With this control law, the dynamics of the tracking error

$$\boldsymbol{e}(t) := \boldsymbol{x}(t) - \boldsymbol{x}_{\text{ref}}(t)$$

is obtained from Eqs. (1) and (2) as

$$\dot{\boldsymbol{e}}(t) = (A + BK)\boldsymbol{e}(t) \tag{4}$$

Because $A + BK$ is Hurwitz, $\boldsymbol{x}(t) \to \boldsymbol{x}_{\text{ref}}(t)$ and $\boldsymbol{y}(t) \to \boldsymbol{y}_d(t)$ as $t \to \infty$ and, therefore, output tracking is achieved. It is noted that in this output tracking scheme, the reference state trajectory $\boldsymbol{x}_{\text{ref}}(\cdot)$ and the feedforward input $\boldsymbol{u}_{\text{ff}}(\cdot)$ can be computed offline. It is also typical for the desired output trajectory to be a composition of several predetermined subtrajectories, and the primary guidance scheme may concatenate several precomputed guidance laws. These primary guidance laws could be found using inversion of the plant dynamics (for systems with same number of inputs as outputs see, e.g., Refs. 3, 5, 9, and 17, and for actuator-redundant systems see Ref. 18).

A critical problem with the preceding guidance scheme is that large errors between the reference state trajectory and the actual system state at the beginning of a maneuver can lead to actuator saturation and to substantially deteriorated performance. Note that the input-saturation problem is accentuated if a relatively high-gain regulator K is used, which may be necessary to achieve high performance for relatively small tracking errors. The goal of the recovery-guidance maneuver is to modify the primary guidance law (reference state trajectory) such that the system states can be brought close to the reference state trajectory, i.e., output tracking is recovered, without saturating the actuators and without violating state constraints. Recovery guidance from large external perturbations can also be studied under this framework provided the perturbation is modeled as an initial tracking error that triggers the recovery guidance generator.

III. Trajectory Recovery Guidance Scheme

The recovery guidance scheme aims to maneuver the system without violating input and state constraints. The approach is to modify the reference state trajectory to avoid constraint violations (see Fig. 2). This modification, $\boldsymbol{x}_{\text{mod}}(\cdot)$, is essentially open loop; however, it is assumed that the recovery guidance trajectory generator has access to the initial tracking error (when the recovery guidance is initiated).

With the modified state trajectory, the control law given by Eq. (3) becomes (see Fig. 2)

$$\begin{aligned} \boldsymbol{u}(t) &= \boldsymbol{u}_{\text{ff}}(t) + \boldsymbol{u}_{\text{fb}}(t) \\ &= \boldsymbol{u}_{\text{ff}}(t) + K(\boldsymbol{x}(t) - \boldsymbol{x}_{\text{ref}}(t)) + K\boldsymbol{x}_{\text{mod}}(t) \\ &= \boldsymbol{u}_{\text{ff}}(t) + K\boldsymbol{e}(t) + K\boldsymbol{x}_{\text{mod}}(t) \end{aligned} \tag{5}$$

where

$$\boldsymbol{u}_{\text{fb}}(t)(t) = K\boldsymbol{e}(t) + K\boldsymbol{x}_{\text{mod}}(t) \tag{6}$$

and the error equation (4) becomes

$$\dot{\boldsymbol{e}}(t) = (A + BK)\boldsymbol{e}(t) + BK\boldsymbol{x}_{\text{mod}}(t) \tag{7}$$

A. Use of Convexity to Generate Recovery Maneuvers

The key idea is to precompute the trajectory modification for a set of initial conditions (from a set X), and then use these precomputed modifications for other initial conditions. This is summarized in the following proposition that uses the convexity argument due to Mayne and Schroeder.[16] The proposition states that if satisfactory recovery guidance is known for a set of initial conditions X then satisfactory recovery guidance maneuvers can be obtained for any initial condition in the convex hull of X, provided the following assumption holds.

Assumption 1: The region of acceptable states in $\Re^n$ and the region of acceptable inputs in $\Re^q$ are both convex regions that contain the origin.

Proposition 1: For each initial condition $\boldsymbol{x}^m(0) \in X$, let $\boldsymbol{x}^m_{\text{mod}}(\cdot)$ be a trajectory modification that enables guidance recovery while

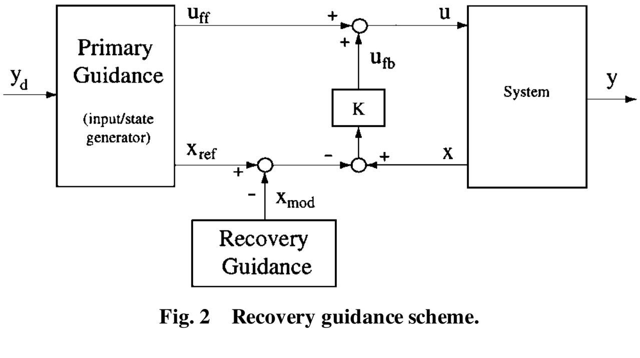

Fig. 2 Recovery guidance scheme.

satisfying the state and input constraints. Let $\boldsymbol{x}(0)$ be an initial condition in the convex hull of X and $\boldsymbol{\lambda}_{x(0)} \in \Lambda_p$ for some positive integer p, where

$$\Lambda_p := \left\{ \boldsymbol{\lambda} \in \Re^p \,|\, \lambda_m \geq 0 \;\; \forall_m \in (1, 2, \ldots, p) \;\text{ and }\; \sum_{m=1}^{p} \lambda_m = 1 \right\}$$

and where subscripts indicate vector components, such that

$$\boldsymbol{x}(0) = \sum_{m=1}^{p} \lambda_{[x(0),m]} \boldsymbol{x}^m(0)$$

where the superscript denotes elements of a sequence. Then, the trajectory modification

$$\boldsymbol{x}_{\text{mod}}(t) := \sum_{m=1}^{p} \lambda_{[x(0),m]} \boldsymbol{x}^m_{\text{mod}}(t)$$

is a valid recovery guidance law, i.e., input constraints and state constraints are satisfied.

Proof: The error dynamics [Eq. (7)] are linear in the initial condition $\boldsymbol{e}(0)$ and in the input $\boldsymbol{x}_{\text{mod}}(\cdot)$. Because $\boldsymbol{e}(0)$ and $\boldsymbol{x}_{\text{mod}}(\cdot)$ are linear combinations of $e^m(0)$ and $x^m_{\text{mod}}(\cdot)$ with the same weights for each $m \in (1, 2, \ldots, p)$, the linearity of the error dynamics yields

$$\boldsymbol{e}(t) = \sum_{m=1}^{p} \lambda_{[x(0),m]} \boldsymbol{e}^m(t)$$

where, $\boldsymbol{e}^m(\cdot)$ is the solution to the error dynamics corresponding to the trajectory modification $\boldsymbol{x}^m_{\text{mod}}(\cdot)$ and initial condition $\boldsymbol{e}^m(0)$.

Let $\boldsymbol{x}^m(\cdot)$ be the solution to the modified state equation with initial condition $\boldsymbol{x}^m(0)$ and the guidance modification $\boldsymbol{x}^m_{\text{mod}}(\cdot)$. Then, the definition of the error yields

$$\boldsymbol{x}^m(\cdot) = \boldsymbol{x}_{\text{ref}}(\cdot) + \boldsymbol{e}^m(\cdot)$$

This equality and

$$\sum_{m=1}^{p} \lambda_{[x(0),m]} = 1$$

which follows from the definition of Λ_p in proposition 1, imply that

$$\begin{aligned} \boldsymbol{x}(t) &:= \boldsymbol{x}_{\text{ref}}(t) + \boldsymbol{e}(t) \\ &= \sum_{m=1}^{p} \lambda_{[x(0),m]} \boldsymbol{x}_{\text{ref}}(t) + \sum_{m=1}^{p} \lambda_{[x(0),m]} \boldsymbol{e}^m(t) \\ &= \sum_{m=1}^{p} \lambda_{[x(0),m]} \boldsymbol{x}^m(t) \end{aligned} \tag{8}$$

Thus, the convex hull of $\boldsymbol{x}^m(t)$ forms a tube (as time varies); assumption 1 implies that the tube is in the acceptable region of the state space and, hence, $\boldsymbol{x}(\cdot)$, which lies in this tube, is also acceptable.

Similarly, linearity of the modified input in $\boldsymbol{e}(\cdot)$ and $\boldsymbol{x}_{\text{mod}}(\cdot)$ [see Eq. (5)] yields

$$\begin{aligned} \boldsymbol{u}(t) &:= \boldsymbol{u}_{\text{ff}}(t) + K\boldsymbol{e}(t) + K\boldsymbol{x}_{\text{mod}}(t) \\ &= \sum_{m=1}^{p} \lambda_{[x(0),m]} \boldsymbol{u}_{\text{ff}}(t) + K \sum_{m=1}^{p} \left[\lambda_{[x(0),m]} \boldsymbol{e}^m(t) \right] \\ &\quad + K \sum_{m=1}^{p} \left[\lambda_{[x(0),m]} \boldsymbol{x}^m_{\text{mod}}(t) \right] \\ &= \sum_{m=1}^{p} \lambda_{[x(0),m]} \boldsymbol{u}^m(t) \end{aligned} \tag{9}$$

where $\boldsymbol{u}^m(\cdot)$ is the modified input for the initial condition $\boldsymbol{x}^m(0)$. Thus, the convex hull of $\boldsymbol{u}^m(t)$ also forms a tube (as time varies); assumption 1 implies that the tube is in the convex acceptable region of the input space and, hence, $\boldsymbol{u}(\cdot)$, which lies in this tube, is also acceptable. □

Remark 1: Proposition 1 is also valid for time-varying systems.

B. Decoupling Recovery Guidance from Primary Guidance

In the described approach, the guidance modification $\boldsymbol{x}_{\text{mod}}(\cdot)$ might depend on the particular maneuver being considered; thus, for each maneuver, the $\boldsymbol{x}_{\text{mod}}(\cdot)$ to be stored may be different even if the initial tracking-error $\boldsymbol{e}^m(0)$ remains the same. We propose a method to trade off the storage requirements with performance, to remove the dependence of guidance modification $\boldsymbol{x}_{\text{mod}}(\cdot)$ from the primary guidance maneuver $[\boldsymbol{u}_{\text{ff}}(\cdot), \boldsymbol{x}_{\text{ref}}(\cdot)]$.

Note that bounds on the input components can be decoupled into separate bounds on the feedforward and feedback by using the triangle inequality

$$\begin{aligned} |u_i(t)| &\leq \left| u_{(\text{ff},i)}(t) \right| + \left| K\left(e_i(t) + x_{(\text{mod},i)}(t)\right) \right| \\ &:= \left| u_{(\text{ff},i)}(t) \right| + \left| u_{(\text{fb},i)}(t) \right|, \qquad \forall i \in (1, 2, \ldots, q) \end{aligned} \tag{10}$$

The primary guidance can be designed to ensure that the feedforward input components $|u_{(\text{ff},i)}(t)|$ are within bounds, and the recovery guidance design can be aimed at limiting the feedback components $|u_{(\text{fb},i)}(t)|$. This distribution of control authority allows the bounds on feedback input $\boldsymbol{u}_{\text{fb}}$ to be independent of the feedforward input u_{ff}. Thus, only the feedforward inputs, from the primary guidance, depend on the particular maneuver.

Similarly, bounds on the state components can also be decoupled into bounds on the components of the error $|e_i(t)|$ and bounds on maneuver-dependent reference state trajectory components $|x_{(\text{ref},i)}(t)|$

$$\begin{aligned} |x_i(t)| &\leq \left| x_{(\text{ref},i)}(t) \right| + \left| \left(x_i(t) - x_{(\text{ref},i)}(t) \right) \right| \\ &:= \left| x_{(\text{ref},i)}(t) \right| + |e_i(t)|, \qquad \forall i \in (1, 2, \ldots, n) \end{aligned} \tag{11}$$

Again, the primary guidance can be designed to ensure that $|x_{(\text{ref},i)}(t)|$ is within bounds, and the recovery guidance design can be aimed at limiting $|e_i(t)|$. This decoupling is formalized by the following proposition.

Assumption 2: The region of acceptable errors in $\Re^n$ and the region of acceptable feedback inputs in $\Re^q$ are convex regions that contain the origin.

Proposition 2: For any initial error $\boldsymbol{e}^m(0) \in E$, let $\boldsymbol{x}^m_{\text{mod}}(\cdot)$ be a trajectory modification that enables guidance recovery and satisfies the error and feedback-input constraints. Then, given an initial condition $\boldsymbol{e}(0)$ in the convex hull of E and $\boldsymbol{\lambda}_{e(0)} \in \Lambda_p$ for some p, such that

$$\boldsymbol{e}(0) = \sum_{m=1}^{p} \lambda_{[e(0),m]} \boldsymbol{e}^m(0)$$

the trajectory modification

$$\boldsymbol{x}_{\text{mod}}(t) = \sum_{m=1}^{p} \lambda_{[e(0),m]} \boldsymbol{x}^m_{\text{mod}}(t)$$

is a valid recovery guidance law, i.e., the input and state constraints are satisfied.

Proof: This follows from arguments similar to those for the proof of proposition 1 and inequalities (10) and (11). □

This proposition implies that the design of the trajectory modification x_{mod} can be done independently, of the particular primary guidance law. However, this decoupling of the recovery guidance from the primary guidance trades off performance for reduced storage.

IV. Recovery Guidance Generation

A. Recovery for a Single Initial Condition

Given a specific initial condition, any algorithm may be used to find a particular recovery guidance maneuver. The aim of this paper is not to produce a recovery guidance maneuver for a given initial condition. Rather, the goal is to use a finite set of recovery guidance maneuvers to generate new recovery guidance maneuvers for other initial conditions. However, to illustrate the method for an arbitrary initial condition, we begin by using an optimal control formulation to design recovery guidance maneuvers for a particular initial condition.

The tradeoff between the need to bound the error and the need to bound the feedback input can be posed as an optimization problem, as the minimization of [over all inputs $x_{\text{mod}}(\cdot)$ to the error dynamics (7)]

$$\begin{aligned} J &:= \int_0^\infty \left\{ \boldsymbol{e}^T(t) Q_e \boldsymbol{e}(t) + \boldsymbol{u}_{\text{fb}}^T(t) R \boldsymbol{u}_{\text{fb}}(t) + \boldsymbol{x}_{\text{mod}}^T(t) Q_{\text{mod}} \boldsymbol{x}_{\text{mod}}(t) \right\} \mathrm{d}t \\ &= \int_0^\infty \left\{ \boldsymbol{x}_{\text{mod}}^T(t) \left(K^T R K + Q_{\text{mod}} \right) \boldsymbol{x}_{\text{mod}}(t) \right. \\ &\quad \left. + 2\boldsymbol{e}^T(t)(K^T R K)\boldsymbol{x}_{\text{mod}}(t) + \boldsymbol{e}^T(t)\left(K^T R K + Q_e\right)\boldsymbol{e}(t) \right\} \mathrm{d}t \\ &:= \int_0^\infty \left\{ \boldsymbol{x}_{\text{mod}}^T(t) \hat{R} \boldsymbol{x}_{\text{mod}}(t) + 2\boldsymbol{e}^T(t)\hat{S}\boldsymbol{x}_{\text{mod}}(t) + \boldsymbol{e}^T(t)\hat{Q}\boldsymbol{e}(t) \right\} \mathrm{d}t \end{aligned} \tag{12}$$

where Q_{mod} is the weighting factor on the modification ($\boldsymbol{x}_{\text{mod}}$) of the primary guidance law, Q_e is the weighting factor on the tracking error $\boldsymbol{e}$, and R is the weighting factor on the feedback input. This optimization problem is the standard optimal control problem for the modified error dynamics equation (7)

$$\dot{\boldsymbol{e}}(t) = (A + BK)\boldsymbol{e}(t) + BK\boldsymbol{x}_{\text{mod}}(t)$$

with $\boldsymbol{x}_{\text{mod}}(\cdot)$ as the input and $\boldsymbol{e}$ as the state. The optimal trajectory modification law is then obtained as (see, for example, Ref. 19 for the development of the standard optimal control law and conditions on the weighting matrices)

$$\boldsymbol{x}_{\text{mod}}(t) = \hat{K}\boldsymbol{e}(t) \tag{13}$$

where

$$\hat{K} = -\hat{R}^{-1}((BK)^T P + \hat{S})$$

and P is the symmetric solution to the algebraic Riccati equation

$$\begin{aligned} &P(A + BK) + (A + BK)^T P - (P(BK) + \hat{S})\hat{R}^{-1} \\ &\times ((BK)^T P + \hat{S}^T) + \hat{Q} = 0 \end{aligned}$$

with $\hat{S}$, $\hat{Q}$, and $\hat{R}$ as defined in Eq. (12).

Substituting the modification law into the error dynamics Eq. (7) and solving the resulting equation yields an open-loop trajectory modification that depends only on the initial error

$$\begin{aligned} \boldsymbol{x}_{\text{mod}}(t) &= \hat{K}\exp[(A + BK + BK\hat{K})(t)]\boldsymbol{e}(0) \\ &:= \hat{K}\exp[(A + BK_e)(t)]\boldsymbol{e}(0) \end{aligned} \tag{14}$$

with $K_e = K + K\hat{K}$. For each initial error $\boldsymbol{e}(0)$, the weightings (Q_e, R, and Q_{mod}) used in the optimization cost criterion can be different; however, only the resulting K_e in Eq. (14) needs to be stored for each of the initial conditions.

Note, again, that other algorithms may be used to obtain suitable recovery guidance laws for prespecified initial conditions in E. Alternate approaches to finding the guidance modification, $\boldsymbol{x}_{\text{mod}}(\cdot)$, include the following.

1) The weighting factors in the given cost function Q_e, R, and Q_{mod} can be time varying.[19]

2) The optimization problem can also be posed in the frequency domain, and the weighting factors can also be made frequency dependent to account for bandwidth limitation of the individual actuators.[20]

3) Another approach is to achieve zero tracking error in finite time.[19]

Next, we discuss using these precomputed recovery guidance laws to generate recovery guidance for other initial conditions.

B. Recovery Guidance for Initial Error in Convex Hull of E

Let E be a set of initial errors for which the recovery guidance maneuvers are known. Any initial error in the convex hull of E may then be written as the convex combination of elements of E

$$\boldsymbol{e}(0) = \sum_m \lambda_m \boldsymbol{e}^m(0) \tag{15}$$

where $\boldsymbol{e}^m(0) \in E$. This representation may not be unique; however, each representation leads to an acceptable recovery guidance maneuver (by proposition 2).

Remark 2: The particular choice of the convex combination $\boldsymbol{e}^m(0)$ can be optimized by using an additional criterion, for example, to maximize the convergence rate of the error dynamics. Note that the resulting error dynamics $\boldsymbol{e}(\cdot)$ will converge as fast as (if not faster than) the slowest $\boldsymbol{e}^m(\cdot)$ that has nonzero coefficients in Eq. (15), which gives a lower bound to the convergence rate. Therefore, the coefficients λ_m of the convex combination in Eq. (15) may be chosen such that lower bound of the convergence rate is maximized.

We present one technique to find a recovery guidance law by appropriately defining the set of initial errors E for which the recovery guidance maneuver is precomputed. The technique is presented in the following four steps.

1) Define the convex error set $V_{(\alpha,\beta)}$. Let the set of acceptable errors $V_{(\alpha,\beta)}$ be defined by upper and lower bounds on the components of the error vector, i.e., let $\boldsymbol{\alpha} \in \Re^n$ and $\boldsymbol{\beta} \in \Re^n$ be given such that

$$\alpha_i < \beta_i, \qquad \forall i \in (1, 2, \ldots, n) \tag{16}$$

and any acceptable initial error $\boldsymbol{e}(0) \in V_{(\alpha,\beta)}$ satisfies

$$\alpha_i \le e_i(0) \le \beta_i, \qquad \forall i \in (1, 2, \ldots, n) \tag{17}$$

2) Define the set E for which recovery guidance is precomputed. We define the corner points of the set of acceptable errors $V_{(\alpha,\beta)}$ as the set of initial errors E for which the recovery guidance maneuvers are precomputed, i.e.,

$$E := \{\boldsymbol{e} \mid e_i = \alpha_i \ \text{ or } \ e_i = \beta_i, \ \ \forall i \in (1, 2, \ldots, n)\}$$

Let $\{\boldsymbol{e}^m\}_{m=1}^{m=2^n}$ be an enumeration of E. Each $\boldsymbol{e}^m$ should be written as $\boldsymbol{e}^m(0)$; however, the dependence on time ($t = 0$) is not explicitly stated for ease in notation.

3) Weight λ for an arbitrary initial error in $V_{(\alpha,\beta)}$. The weighting for an arbitrary initial error can be found using equations presented in the following claim.

Claim: Any initial error $\boldsymbol{e}(0) \in V_{(\alpha,\beta)}$ can be written as

$$\boldsymbol{e}(0) = \sum_{m=1}^{2^n} \lambda_m \boldsymbol{e}^m \tag{18}$$

where $\boldsymbol{\lambda} \in \Lambda_{2^n}$, and

$$\lambda_m = \prod_{i=1}^{n} \gamma_i\left(e_i, e_i^m\right) \tag{19}$$

with

$$\begin{aligned} \gamma_i\left(e_i, e_i^m\right) &:= \frac{\beta_i - e_i}{\beta_i - \alpha_i} \qquad \text{if} \qquad e_i^m = \alpha_i \\ &:= \frac{e_i - \alpha_i}{\beta_i - \alpha_i} \qquad \text{if} \qquad e_i^m = \beta_i \end{aligned} \tag{20}$$

Proof: This follows from an induction argument. □

4) Use $\boldsymbol{\lambda}$ to determine the recovery guidance maneuver. In this final step, the weighting $\boldsymbol{\lambda}$ found from Eq. (18) is used to generate the recovery guidance maneuver as discussed in proposition 2.

Remark 3: The performance of the recovery guidance maneuvers can be improved by decomposing the set of initial errors into smaller sets $V_{(\alpha^j,\beta^j)}$, $j \in (1, 2, \ldots, J)$, and developing recovery guidance maneuvers for each of the smaller sets. This can lead to improved performance; however, it also requires an increase in storage (to store precomputed recovery maneuvers for the corner points of each of the smaller sets).

V. Example

A. System

The recovery guidance problem for a system with a rate-limited servoactuator is considered in the following example. Let the aircraft dynamics be described by (linearized altitude dynamics of a transport aircraft with direct-lift flaps, see Example 8.3 in Ref. 21)

$$\dot{x}_1(t) = x_2(t), \qquad \dot{x}_2(t) = -1.4x_2(t) + 50\delta_f(t)$$
$$y(t) = x_1(t) \tag{21}$$

where y, $\dot{y}$, and $\ddot{y}$, i.e., x_1, x_2, and $\dot{x}_2$, are the incremental altitude (from a nominal altitude), velocity, and acceleration, respectively.

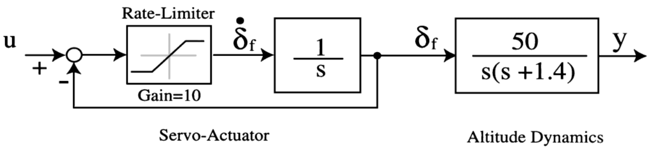

Fig. 3 Example system.

Additionally (as is typical in aircraft control), we consider the case when the input $\delta_f(t)$ is not directly accessible and is generated through a servoactuator (see Fig. 3). We consider rate boundedness of the servoactuator and represent its dynamics by[22]

$$\dot{\delta}_f(t) = 10[u(t) - \delta_f(t)] \tag{22}$$

where the commanded input is $u(t)$ and the actual input applied by the servosystem to the aircraft is $\delta_f(t)$. If saturation is avoided, then the state-space model of the entire system can be described as

$$\dot{x}_1(t) = x_2(t), \qquad \dot{x}_2(t) = -1.4x_2(t) + 50x_3(t)$$
$$\dot{x}_3(t) = -10x_3(t) + 10u(t) \tag{23}$$

where $x_3(t) = \delta_f(t)$ and x_1, x_2, and x_3 are the components of the state vector $\boldsymbol{x}$. Further, a feedback law, $u_{fb} = K(x - x_{ref})$, is assumed to be given such that the closed-loop system is stable. For all simulations, K is fixed as

$$K = [-31.6 \quad -2.4 \quad -4]$$

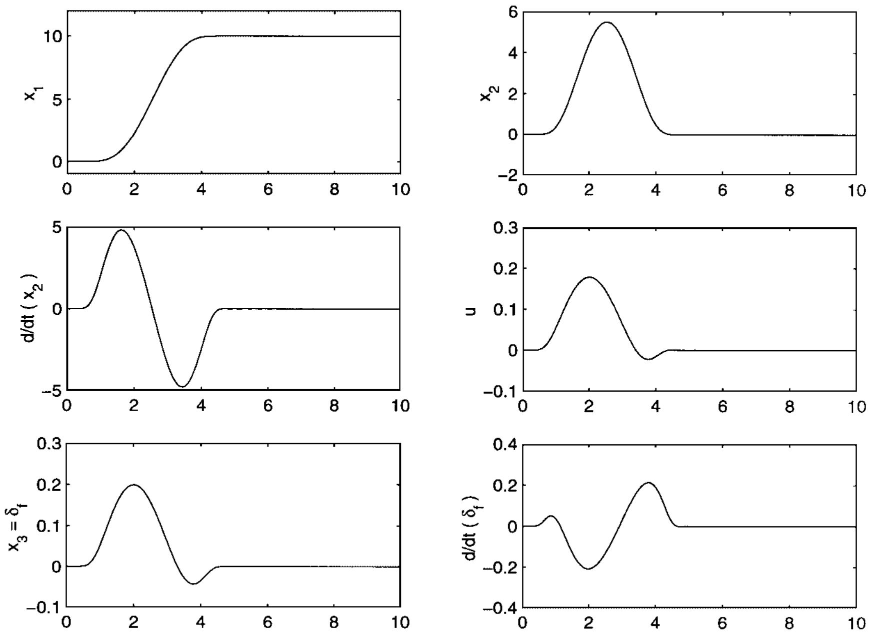

Fig. 4 Primary guidance law; horizontal axis represents time.

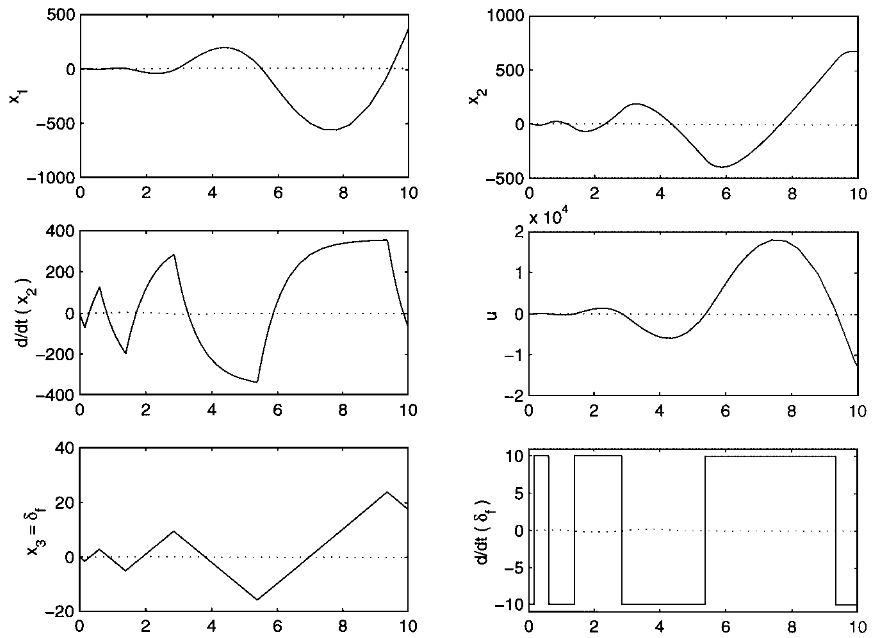

Fig. 5 Results with input saturation: initial position error = 1 and initial velocity error = 0; · · · · ·, primary guidance; horizontal axis represents time.

B. Constraints

The servorate is limited by

$$|\dot{\delta}_f(t)| = |10[u(t) - \delta_f(t)]| < 10 \tag{24}$$

This constraint can be rewritten as $|u(t) - \delta_f(t)| < 1$, which is converted to limits on the states and inputs as

$$\begin{aligned} |u(t)| &< 0.5 \\ |\delta_f(t)| = |x_3(t)| &< 0.5 \end{aligned} \tag{25}$$

Remark 4: This example illustrates a technique to convert constraints on a general variable v that is linear in the input and state

$$v(t) = F_1\boldsymbol{x}(t) + F_2\boldsymbol{u}(t)$$

into constraints on the state and input. For example, $|v(t)| < v_{\max}$ can be written as $\|\boldsymbol{x}(t)\|_\infty < (\theta/\|F_1\|_\infty)v_{\max}$ and $\|\boldsymbol{u}(t)\|_\infty < [(1-\theta)/\|F_2\|_\infty]v_{\max}$, where $\|x(t)\|_\infty := \max_i |\boldsymbol{x}_i(t)|$ is the standard ∞ norm in $\Re^n$, with $0 < \theta < 1$. It is assumed that both F_1 and F_2 have at least one nonzero term. $\|F_i\|_\infty$ is the standard induced ∞-norm for $i = 1$ and $i = 2$. $\|F_i\|_\infty := \sup_{y \neq 0, y \in \Re^n} \|F_i\boldsymbol{y}\|_\infty / \|\boldsymbol{y}\|_\infty$.

In the following, the control authority is split 50–50 between the primary guidance and the recovery guidance as follows: the primary guidance generator is to ensure that the magnitude of the servorate is less than half the maximum limit, i.e.,

$$|u_{\mathrm{ff}}(t)| < 0.25, \qquad \left|x_{3_{\mathrm{ref}}}(t)\right| < 0.25 \tag{26}$$

and the recovery guidance generator should ensure that the magnitude of the servorate does not exceed half the maximum limit, i.e.,

$$\begin{aligned} |u_{\mathrm{fb}}(t)| &< 0.25 \\ |e_3(t)| := \left|x_3(t) - x_{3_{\mathrm{ref}}}(t)\right| &< 0.25 \end{aligned} \tag{27}$$

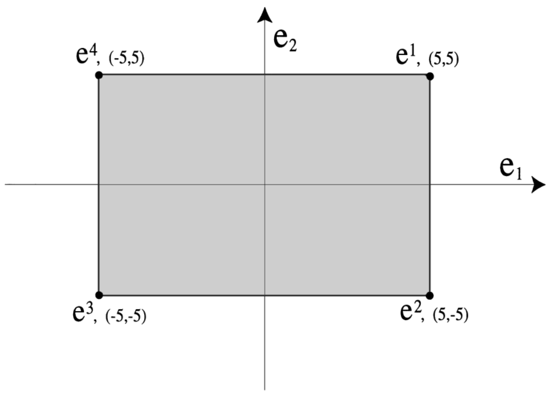

Fig. 6 Shaded area represents convex set of initial errors. Recovery maneuvers are precomputed for the corner points (e^1, e^2, e^3, and e^4). These precomputed maneuvers are then used for online generation of recovery maneuvers for any other initial error in the shaded region.

C. Design of the Primary Guidance Generator

Let a desired output profile $y_d(\cdot)$ and its first, second, and third time derivatives $[\,y_d^{(1)}\;y_d^{(2)}\;y_d^{(3)}]$ be given. Then, the exact-output tracking input state trajectory can be found from Eq. (23) as

$$\begin{bmatrix} x_{1\mathrm{ref}} \\ x_{2\mathrm{ref}} \\ x_{3\mathrm{ref}} \end{bmatrix}(t) = \begin{bmatrix} y_d \\ y_d^{(1)} \\ \frac{1}{50}\left[y_d^{(2)} + 1.4y_d^{(1)}\right] \end{bmatrix}(t) \tag{28}$$

where the superscript in parentheses is for time derivative. The existence of the time derivatives of the desired output is a necessary condition for exact tracking.[9] The exact-tracking input trajectory $u_{\mathrm{ff}}(t)$ is obtained by differentiating $x_{3\mathrm{ref}}$ with respect to time in Eq. (28) to get

$$u_{\mathrm{ff}}(t) = \frac{1}{500}\left[y_d^{(3)}(t) + 11.4y_d^{(2)}(t) + 14y_d^{(1)}(t)\right] \tag{29}$$

Note that the restrictions on the primary guidance, Eq. (26), may imply that some of the desired trajectory profiles may not be feasible, thus requiring the redesign of the desired output trajectory profiles (see Ref. 23 for an optimal approach to output-trajectory redesign for invertible systems). Figure 4 shows a primary guidance trajectory for changing the position y from 0 to 10 units; here the initial state error is zero. Figure 5 shows the system response to a nonzero initial condition when the servorate was allowed to saturate. Note the rapid degradation of tracking performance. The initial position error was 1 unit (note that the maximum change in position is 10 units), and all other state components had zero error. Note that the controller would give satisfactory performance if the initial error was relatively smaller (so that the servorate does not saturate); in such cases recovery guidance is not needed. Next, we discuss the generation of the recovery guidance maneuvers for large initial errors.

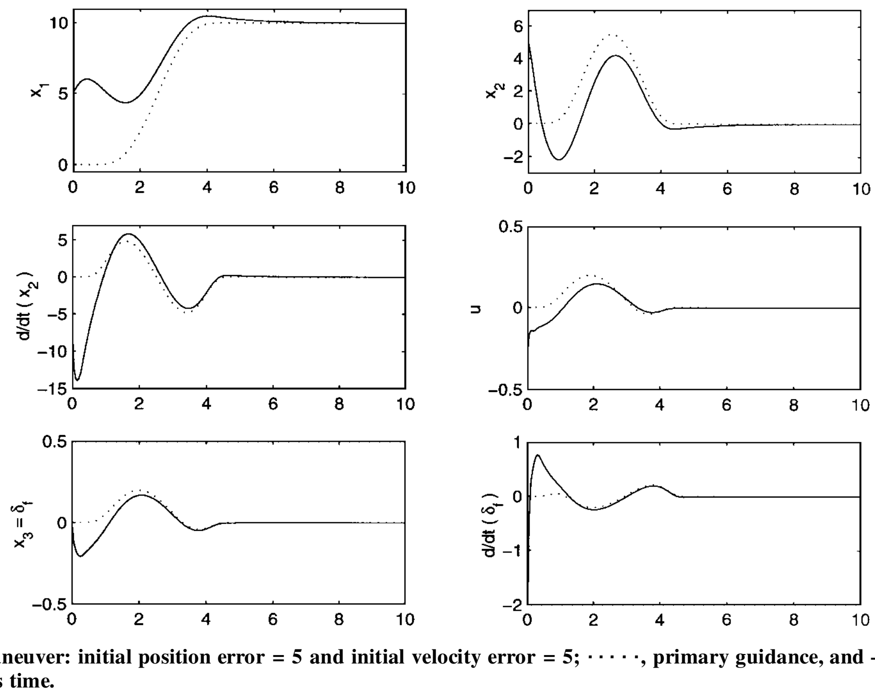

Fig. 7 Precomputed maneuver: initial position error = 5 and initial velocity error = 5; · · · · ·, primary guidance, and ——, modified maneuver; horizontal axis represents time.

D. Development of Recovery Guidance

To reduce the number of simulations, in the following discussion it is assumed that the initial condition of the actuator state x_3 is zero, i.e., initial errors have the form $[e_1 \;\; e_2 \;\; 0]^T$. If the initial condition of the actuator state is nonzero, then the only change is that the number of initial errors (corner points of the error set) for which recovery guidance will have to be precomputed increases from 2^2 to 2^3 (and 4 more simulations will have to be presented). In the simulations, the initial tracking errors in the position and velocity are assumed to be in the following range (see Fig. 6) $|e_1| \leq 5$ and $|e_2| \leq 5$. This initial error of 5 units is large; it is 50% of the maximum change in position (10 units) during the entire maneuver (see Fig. 4).

For this particular example, we compute recovery guidance laws for four extreme initial errors (e^1, e^2, e^3, and e^4, which are corner points of the set of acceptable initial errors in Fig. 6), and the results of the recovery guidance maneuvers are shown in Figs. 7–10. Even with substantially large initial tracking errors the resulting recovery guidance maneuvers lead to recovery of trajectory tracking without violating input and state constraints (compare Figs. 7–10 with the case without recovery guidance in Fig. 5). However, the design of such recovery guidance requires the manipulation of the weighting matrices in the cost criterion and requires repeated simulations to check performance. This design is not computationally feasible online for an arbitrary initial condition. The idea is to generate recovery guidance laws that satisfy the system and input constraints for a few initial conditions (e^1, e^2, e^3, and e^4) and then use these to generate online recovery guidance laws for other initial conditions.

For the present problem, the other acceptable initial conditions are in the shaded area defined by the four extreme (corner) points in Fig. 6. An acceptable initial error $e(0)$ can then be written as a convex combination of the corner points (e^1, e^2, e^3, and e^4) as (see steps 1–3 in Sec. IV.B)

$$\boldsymbol{e}(0) = \sum_{m=1}^{4} \lambda_m \boldsymbol{e}^m$$

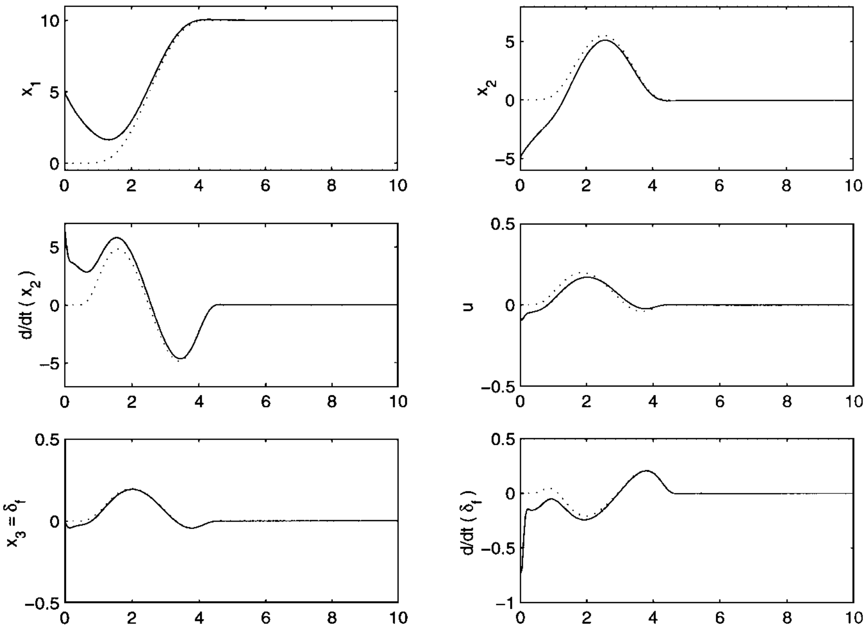

Fig. 8 Precomputed maneuver: initial position error = 5 and initial velocity error = − 5; · · · · ·, primary guidance, and ——, modified maneuver; horizontal axis represents time.

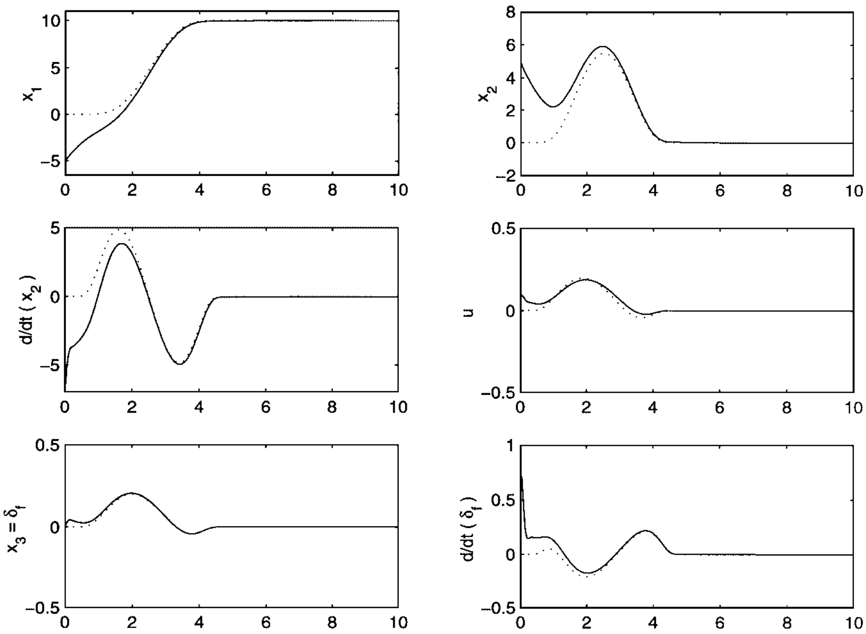

Fig. 9 Precomputed maneuver: initial position error = − 5 and initial velocity error = 5; · · · · ·, primary guidance, and ——, modified maneuver; horizontal axis represents time.

with

$$\lambda_m = \prod_{i=1}^{2} \gamma_i\left[e_i(0), e_i^m\right] = \gamma_1\left[e_1(0), e_1^m\right]\gamma_2\left[e_2(0), e_2^m\right]$$

where

$$\begin{aligned} \gamma_1\left[e_1(0), e_1^m\right] &:= \tfrac{1}{10}[e_1(0) + 5] \quad \text{if} \quad m \in \{1, 2\} \\ &:= \tfrac{1}{10}[5 - e_1(0)] \quad \text{if} \quad m \in \{3, 4\} \\ \gamma_2\left[e_2(0), e_2^m\right] &:= \tfrac{1}{10}[e_2(0) + 5] \quad \text{if} \quad m \in \{1, 4\} \\ &:= \tfrac{1}{10}[5 - e_2(0)] \quad \text{if} \quad m \in \{2, 3\} \end{aligned} \tag{30}$$

The recovery guidance law can then be written as a convex combination of the precomputed recovery guidance laws (for the corner points) as in proposition 2.

E. Results

Simulation results are presented in Fig. 11, which show the results of the recovery guidance maneuver: Output tracking was achieved without violating the constraints. Note that these results (Fig. 11) are for the same initial condition error that 1) led to rate saturation and 2) led to loss of tracking when recovery guidance was not used (compare Figs. 5 and 11). In contrast, the recovery guidance maneuver (Fig. 11) guaranteed the satisfaction of the state and input constraints. A second case was considered to represent a larger error in initial conditions; the magnitude of initial position error was increased from 1 to 2.5 units, and initial velocity error was increased from 0 to 5 units. Note that the new recovery guidance maneuver was not designed online, but was found as a combination of precomputed maneuvers. As seen in Fig. 12, output tracking was still recovered by the guidance generator without saturating the servoinput rate.

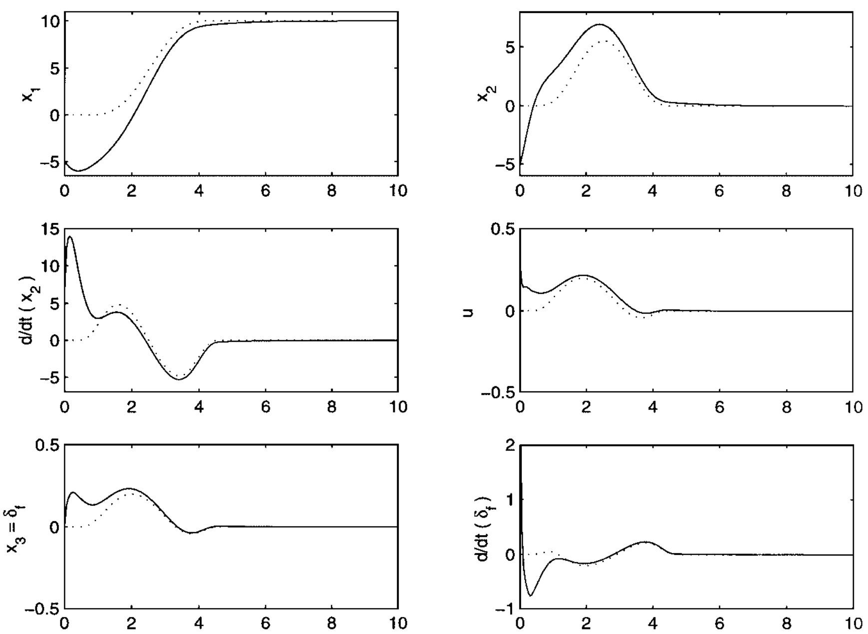

Fig. 10 Precomputed maneuver: initial position error = −5 and initial velocity error = −5; · · · · ·, primary guidance, and ——, modified maneuver; horizontal axis represents time.

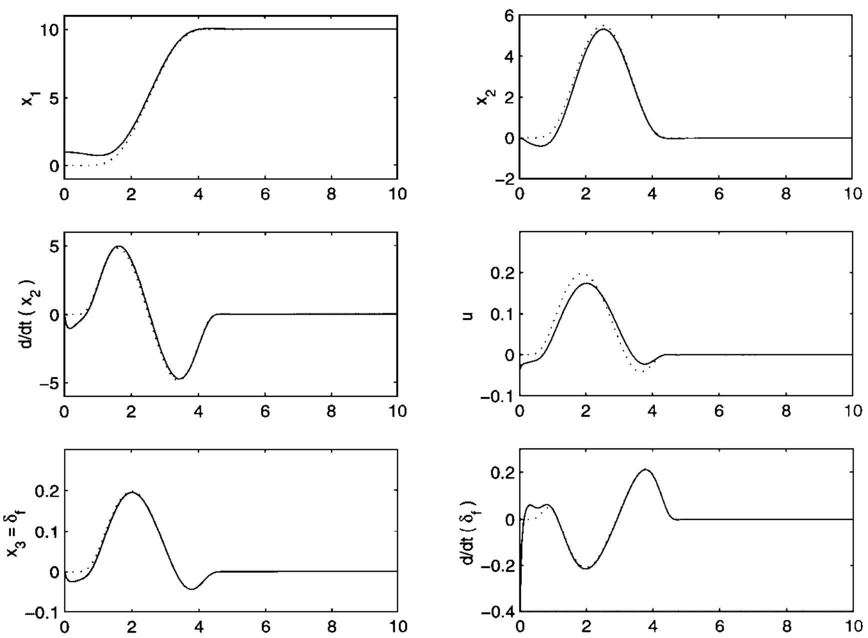

Fig. 11 Results of recovery guidance using precomputed trajectories: initial position error = 1 and initial velocity error = 0; · · · · ·, primary guidance, and ——, modified maneuver; horizontal axis represents time.

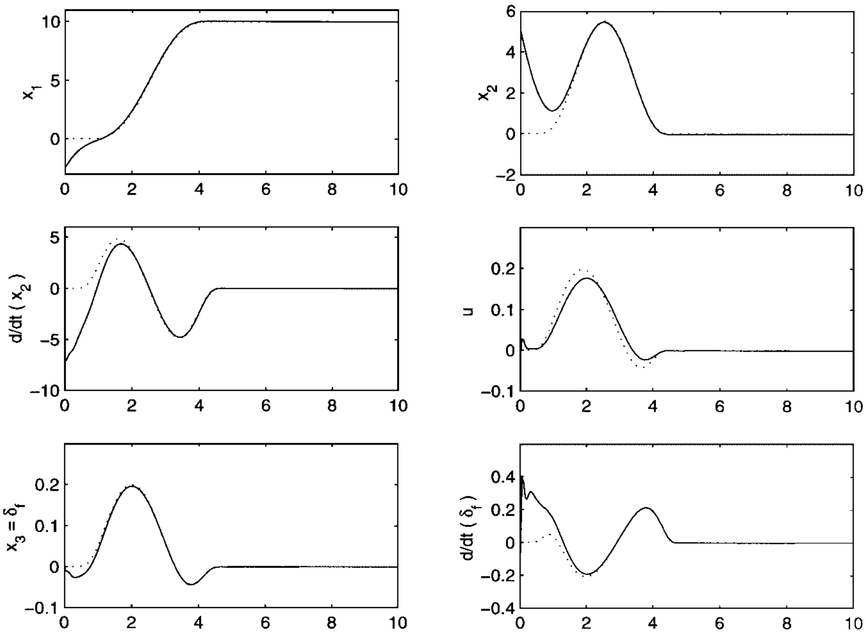

Fig. 12 Results of recovery guidance using precomputed trajectories: initial position error = − 2.5 and initial velocity error = 5; · · · · ·, primary guidance, and ——, modified maneuver; horizontal axis represents time.

VI. Conclusions

A method to modify guidance laws to recover trajectory tracking without violating input and state constraints was presented. The technique uses recovery guidance laws that are precomputed for a few initial conditions to generate satisfactory recovery guidance maneuvers for other initial conditions. The technique was applied to an illustrative example with a rate-limited actuator, and simulation results were presented. Simulations showed that recovery guidance can successfully achieve recovery of trajectory tracking even with relatively large initial tracking errors. In contrast, for the same initial conditions, if such recovery guidance maneuvers were not used, then actuator rate saturation led to eventual loss of tracking.

Acknowledgments

This work was supported in part by NASA Ames Research Center Grant NAG-2-1042. S. Devasia was visiting NASA Ames Research Center under IPA Agreement NCC 2-8018. An earlier version of this paper was submitted to the 1998 American Control Conference.

References

[1]Martin, P., Devasia, S., and Paden, B., "A Different Look at Output Tracking: Control of a VTOL Aircraft," *Automatica*, Vol. 32, No. 1, 1996, pp. 101–107.

[2]Tomlin, C., Lygeros, J., and Sastri, S., "Output Tracking for a Non-minimum Phase Dynamic CTOL Aircraft Model," *Controls and Decision Conference*, Inst. of Electrical and Electronics Engineers Control System Society, 1995, pp. 1867–1872.

[3]Meyer, G., Hunt, L. R., and Su, R., "Nonlinear System Guidance in the Presence of Transmission Zero Dynamics," NASA TM 4661, Jan. 1995.

[4]Slattery, R., and Zhao, Y., "Trajectory Synthesis for Air Traffic Automation," *Journal of Guidance, Control, and Dynamics*, Vol. 20, No. 2, 1997, pp. 232–238.

[5]Devasia, S., Chen, D., and Paden, B., "Nonlinear Inversion-Based Output Tracking," *IEEE Transactions on Automatic Control*, Vol. 41, No. 7, 1996, pp. 930–943.

[6]Meyer, G., and Smith, G. A., "Dynamic Forms Part ii: Applications to Aircraft Guidance," NASA TP 3695, Oct. 1997.

[7]Devasia, S., and Paden, B., "Stable Inversion for Nonlinear Non-minimum-Phase Time-Varying Systems," *IEEE Transactions on Automatic Control*, Vol. 43, No. 2, 1998, pp. 283–288.

[8]Ostroff, A. J., and Proffitt, M. S., "Design and Evaluation of a Stochastic Optimal Feed-Forward and Feedback Technology (SOFFT) Flight Control Architecture," NASA TP 3419, June 1994.

[9]Isidori, A., *Nonlinear Control Systems: An Introduction*, Springer–Verlag, Berlin, 1989, Chap. 4.

[10]Gilbert, E. C., and Tan, K. C., "Linear Systems with State and Control Constraints: The Theory and Application of Maximal Output Admissible Sets," *IEEE Transactions on Automatic Control*, Vol. 36, No. 9, 1991, pp. 1008–1020.

[11]Saberi, A., Lin, Z., and Teel, A. R., "Control of Linear Systems with Saturating Actuators," *IEEE Transactions on Automatic Control*, Vol. 41, No. 3, 1996, pp. 368–378.

[12]Teel, A. R., "Nonlinear Small Gain Theorem for the Analysis of Control Systems with Saturation," *IEEE Transactions on Automatic Control*, Vol. 41, No. 9, 1996, pp. 1256–1270.

[13]Bemporad, A., Casavola, A., and Mosca, E., "Nonlinear Control of Constrained Linear Systems via Predictive Reference Management," *IEEE Transactions on Automatic Control*, Vol. 42, No. 3, 1997, pp. 340–349.

[14]Miller, R. B., and Pachter, M., "Maneuvering Flight Control with Actuator Constraints," *Journal of Guidance, Control, and Dynamics*, Vol. 20, No. 4, 1997, pp. 729–734.

[15]Hess, R. A., and Snell, S. A., "Flight Control System Design with Rate Saturating Actuators," *Journal of Guidance, Control, and Dynamics*, Vol. 20, No. 1, 1997, pp. 90–96.

[16]Mayne, D. Q., and Schroeder, W. R., "Nonlinear Control of Constrained Linear Systems," *International Journal of Control*, Vol. 60, No. 5, 1994, pp. 1035–1043.

[17]Silverman, L. M., "Inversion of Multivariable Linear Systems," *IEEE Transactions on Automatic Control*, Vol. 14, No. 3, 1969, pp. 270–276.

[18]Devasia, S., and Bayo, E., "Redundant Actuators to Achieve Minimal Vibration Trajectory Tracking of Flexible Multibodies: Theory and Application," *Journal of Nonlinear Dynamics*, Vol. 6, No. 4, 1994, pp. 419–431.

[19]Anderson, B. D. O., and Moore, J. B., *Optimal Control—Linear Quadratic Methods*, Prentice–Hall, Englewood Cliffs, NJ, 1990, Chap. 2.

[20]Gupta, N. K., "Frequency Shaped Cost Functionals: Extension of Linear-Quadratic-Gaussian Design Methods," *Journal of Guidance and Control*, Vol. 3, No. 6, 1980, pp. 529–535.

[21]Nelson, R. C., *Flight Stability and Automatic Control*, McGraw–Hill, Boston, 1998, pp. 305, 306.

[22]Klyde, D. H., McRuer, D. T., and Myers, T. T., "Pilot Induced Oscillation Analysis and Prediction with Actuator Rate Limiting," *Journal of Guidance, Control, and Dynamics*, Vol. 20, No. 1, 1997, pp. 81–89.

[23]Devasia, S., "Optimal Output-Trajectory Redesign for Invertible Systems," *Journal of Guidance, Control, and Dynamics*, Vol. 19, No. 5, 1996, pp. 1189–1191.