JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 30, No. 1, January–February 2007

Model-Predictive Dynamic Control Allocation Scheme for Reentry Vehicles

Yu Luo,* Andrea Serrani,† and Stephen Yurkovich‡
The Ohio State University, Columbus, Ohio 43210
and
Michael W. Oppenheimer§ and David B. Doman¶
U.S. Air Force Research Laboratory, Wright–Patterson Air Force Base, Ohio 45433

DOI: 10.2514/1.25473

Allocation of control authority among redundant control effectors, under hard constraints, is an important component of the inner loop of a reentry vehicle guidance and control system. Whereas existing control allocation schemes generally neglect actuator dynamics, thereby assuming a static relationship between control surface deflections and moments about a three-body axis, in this work a dynamic control allocation scheme is developed that implements a form of model-predictive control. In the approach proposed here, control allocation is posed as a sequential quadratic programming problem with constraints, which can also be cast into a linear complementarity problem and therefore solved in a finite number of iterations. Accounting directly for nonnegligible dynamics of the actuators with hard constraints, the scheme extends existing algorithms by providing asymptotic tracking of time-varying input commands for this class of applications. To illustrate the effectiveness of the proposed scheme, a high-fidelity simulation for an experimental reusable launch vehicle is used, in which results are compared with those of static control allocation schemes in situations of actuator failures.

I. Introduction

CONTROL laws for hypersonic reentry vehicles typically consist of multiple loops employing linear and nonlinear methodologies, ultimately specifying moments or angular accelerations in pitch, roll, and yaw. Translating these specifications into control commands to control surface effectors (actuators) is complicated by the fact that the physical architecture is generally characterized by the presence of more control effectors than controlled variables. The implication of this characteristic is that the control system possesses a certain degree of redundancy; that is, there exist several combinations of effector positions to produce similar results. Thus, the physical architecture can, in principle, achieve multiple control objectives and maintain certain performance when control authority is limited by failures. With this increased capability comes the requirement for control allocation schemes that optimize, in real time, some objective criteria, such as minimizing control deflections or drag. Moreover, it is typical for the inner-loop control allocation laws to undergo a separate design from the outer-loop closed-loop control, to simplify the control design and ensure the stability and robustness of the overall system. In this work, we focus on the inner-loop control allocation design, within an architecture based on nonlinear dynamic inversion.

Received 27 May 2006; revision received 8 September 2006; accepted for publication 11 September 2006.

*Graduate Student, Department of Electrical and Computer Engineering, 2015 Neil Avenue.

†Assistant Professor, Department of Electrical and Computer Engineering, 2015 Neil Avenue. Member AIAA.

‡Professor, Department of Electrical and Computer Engineering, 2015 Neil Avenue.

§Aerospace Engineer, Control Design and Analysis Branch, 2210 Eighth Street, Suite 21. Senior Member AIAA.

¶Senior Aerospace Engineer, Control Design and Analysis Branch, 2210 Eighth Street, Suite 21. Associate Fellow AIAA.

Generally speaking, the objective of control allocation is to generate appropriate commands to the actuators to produce the desired control at the plant input. In the presence of hard performance constraints, the control allocation scheme must distribute the available control authority among redundant control effectors to meet the control objectives, and simultaneously satisfy the constraints. To make the problem tractable, and yet to still obtain useful results, research in the area has tended to focus on control design solutions that make two simplifying assumptions: linear relationships between moments and effector surface deflections, and static modeling for the actuators.

Following the seminal work of Durham [1–3], several algorithms for linear control allocation and linear control mixing have been developed over the last decade [4–12], and a few survey papers exist [13,14] that point out the advantages and disadvantages of various control allocation schemes. For the most part, existing linear control allocation algorithms are capable of dealing with systems under nominal conditions, where moments are linearly related to control effector positions, and have the ability to account for position constraints. Indeed, it is for this reason that most existing algorithms assume that a linear relationship exists between controlled variables and effector variables. In cases in which this assumption fails, errors in the control allocation schemes must be mitigated by the robustness resulting from feedback control laws. Some research has attempted to free some of the burden on the feedback portion of the control law, especially when failures occur, by including nonlinear effects and thereby increasing the accuracy of the control allocator. For example, in several works [15–18] the concept of nonlinear control allocation is explored to deal with systems where moments are nonlinearly related to effector deflections.

Within the context of linear control allocation techniques for these applications, an alternative to adding nonlinear effects is to account for actuator dynamics in the control design. To date, most algorithms that have appeared (cited above) neglect the effect of actuator dynamics, or deal with the actuator dynamics separately. However, the presence of actuator dynamics can decrease the overall effective bandwidth of the control system [19], and can even accentuate the effect of unmodeled nonlinearities. The method proposed by Härkegård [20] capitalizes on the effector redundancy aspect to allow different actuators to produce control effort over various ranges of

frequencies; in that work, the process was referred to as "dynamic" control allocation. In the work by Hodel et al. [21,22] the same terminology is adopted to denote a control allocation method that, building upon existing algorithms, allows frequency-dependent scaling in the cost function or in the constraints of the optimization problem to shape the frequency content of the commands to suppress chattering of the control effectors.

In this paper, we propose a scheme using the terminology "dynamic control allocation," but here it is to refer to the explicit inclusion of actuator dynamics into the inner control loop design, thereby setting it apart from most existing control allocation schemes where a static relationship between control surface deflections (actuator outputs) and moments about a three-body axis (plant inputs) is assumed. In so doing, we build on previous results [23,24], where a dynamic control allocation scheme (one that included effects of linear actuator dynamics) was introduced for nominal performance studies. In that work the effectiveness of a simple model-predictive control (MPC) architecture was demonstrated. The present paper extends our previous work for cases away from the nominal, when an actuator failure occurs. Actuator failures include lockup failure (where the failed actuator remains in a fixed position, irrespective of the command to the device), surface loss (the actuator is still functioning properly, but the vehicle control surface is partially or completely damaged), and floating surface failure (actuator position floats with the angle of attack). Under these circumstances, control allocation schemes must be able to accommodate the increased demands on the remaining effectors to produce desired and attainable moments.

MPC-based control schemes generally compute the control inputs by optimizing an open-loop control objective over a future time interval at each control step. The focus of this paper is not specifically on the MPC methodology; rather, it offers a novel use of MPC for dynamic control allocation. The proposed scheme does, however, extend schemes found in typical MPC applications (see, for example, the book by Camacho and Bordons [25], and references therein) in that it provides asymptotically stable tracking of nonconstant reference trajectories. In our earlier work [23], for a reentry vehicle with four actuators, we showed that tracking of constant reference trajectories is guaranteed if the control signal remains constant across the prediction horizon. Following this, an MPC-based approach for a reentry vehicle with six control surfaces was introduced [24], for the nominal case of no actuator failure. Building on those results, the dynamic MPC control allocation problem for this paper is posed as a sequential two-step quadratic programming problem with dynamic constraints, which can be cast into a linear complementarity problem, and therefore solved by linear programming approaches in a finite number of iterations. A high-fidelity simulation for an experimental reusable launch vehicle is used, where results are compared with those of static control allocation schemes in situations of actuator failures.

The paper is organized as follows: in Sec. II, the specific control architecture for reentry vehicles adopted in this paper is introduced, alongside with a baseline static control allocation scheme. The dynamic control allocation problem is formalized in Sec. III, where the proposed algorithm is introduced. In Sec. IV, the results of comparative simulation studies with a static control allocation scheme are presented and discussed.

II. Control Architecture for a Reentry Vehicle Model

The guidance and control architecture for reentry vehicles considered in this paper, shown in Fig. 1, is based on the one originally proposed in Doman and Ngo [26], and adopted in subsequent papers [15,17,18,23]. Assuming that a desired angular rate command $\omega_{\text{cmd}}(t) = [p_{\text{cmd}}(t) q_{\text{cmd}}(t) r_{\text{cmd}}(t)]^T$ is made available by the guidance system, the control input required for the vehicle angular velocity $\omega(t)$ to track $\omega_{\text{cmd}}(t)$ is computed in the three-dimensional angular acceleration space by means of a dynamic inversion-based controller. The resulting desired angular acceleration profile is then allocated within a set of redundant actuators, given, in the case considered in this paper, by six aerodynamic control surfaces. Specifically, let the aircraft rotational dynamics be given as

$$\dot{\omega} = f(\omega, \theta) + g(\delta, \theta) \tag{1}$$

where $\omega = [pqr]^T$ is the vector of angular rates, computed with respect to a body-fixed coordinate frame, and the vector $\theta \in \mathbb{R}^p$ contains measurable or estimated time-varying parameters associated with the relevant operating point and trim conditions of the vehicle (i.e., Mach number, angle of attack). The components of the $\delta \in \mathbb{R}^m$ denote the angular deflection of control surfaces of the vehicle playing the role of control input to the system in Eq. (1). The term $f(\omega, \theta)$ includes accelerations due to the base aerodynamics and engine system, whereas $g(\delta, \theta)$ represents control dependent accelerations. Assuming that the inertia matrix $I \in \mathbb{R}^{3\times 3}$ remains constant during reentry, the vector fields $f(\omega, \theta)$ and $g(\delta, \theta)$ read, respectively, as

$$f(\omega, \theta) = I^{-1}[G_{\text{wb}}(\omega, \theta) - \omega \times I\omega], \qquad g(\delta, \theta) = I^{-1} G_{\delta}(\delta, \theta)$$

where $G_{\text{wb}}(\omega, \theta) = [LMN]^T_{\text{wb}}$ is the moment generated by the wing-body system, and $G_{\delta}(\delta, \theta) = [LMN]^T_{\delta}$ is the moment about the body-fixed axis generated by the control surfaces. Real-time information about moment coefficient data is provided to the control system by a specific aerodynamic database integrated with an online system identification algorithm, as described in Doman and Ngo [26]. The online identification scheme is also capable of detecting the occurrence of failures or damages to the control effectors, providing an estimate of the mapping $g(\delta, \theta)$ to be used for control reconfiguration.

The control system is based on a multiloop model-reference scheme [27], and is briefly described here. The reference model is given by the first-order prefilter

$$\omega_m(s) = W_m(s)\omega_{\text{cmd}}(s)$$

where $W_m(s)$ is the multivariable transfer function

$$W_m(s) = (sI + K_m)^{-1} K_m$$

and $K_m \in \mathbb{R}^{3\times 3}$ is a positive definite diagonal matrix. The outer-loop module of the dynamic inversion-based controller in Fig. 1 computes the desired acceleration $\dot{\omega}_{\text{des}}$ as a correction term that is a function of the error between the states of the prefilter and the actual body rates, whereas the inner-loop module provides cancellation of acceleration due to the base aerodynamic system and rotational dynamics. To take

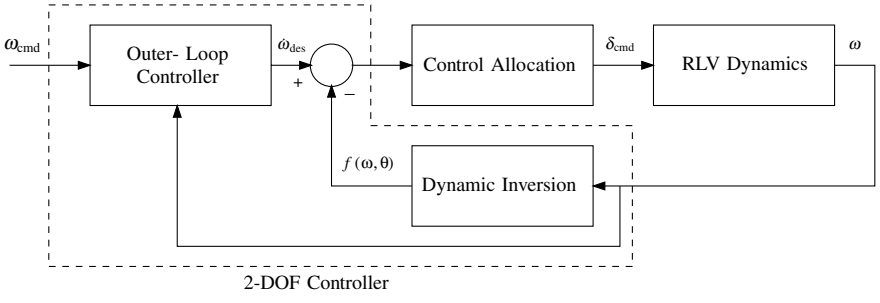

Fig. 1 Model-reference control architecture with dynamic inversion and control allocation.

into account the effect of inversion error, the outer-loop controller provides a proportional, integral, and derivative control action on the tracking error $\tilde{\omega} = \omega_m - \omega$. Specifically, the outer-loop controller is given by the following controller with unitary feedforward action

$$\omega_{\text{des}}(s) = \frac{1}{s^2}(K_d s^2 + K_p s + K_i)\tilde{\omega}(s) + \omega_m(s)$$

where K_p, K_i, and K_d are positive definite diagonal matrices. A simple manipulation shows that, provided that the control allocation module assigns to the control surface deflections δ the desired command δ_{cmd} such that

$$g(\delta_{\text{cmd}}, \theta) = \dot{\omega}_{\text{des}} - f(\omega, \theta) \tag{2}$$

and thus $s\omega(s) = s\omega_{\text{des}}(s)$, the resulting transfer function in closed-loop between the command input and the angular velocity is given precisely by $\omega(s) = W_m(s)\omega_{\text{cmd}}(s)$, yielding asymptotic tracking of the desired angular velocity profile.

In the above formulation, the *control allocation problem* is that of inverting the mapping $u = g(\delta, \theta)$ with respect to δ, and distributing efficiently the control law among the available actuators. Because solving for δ_{cmd} in Eq. (2) amounts to solving an underdetermined equation, additional constraints on the amplitude of control surface deflections can be accommodated in principle by the allocation algorithm. In an attempt to reduce the complexity of the algorithm design, most modern control allocation schemes assume that the control effective mapping is linear, that is,

$$g(\delta, \theta) = G(\theta)\delta$$

Typically, this assumption loses validity when some control surfaces operate in a range close to their position and rate limits, where nonlinear behavior is likely to occur, resulting in an inaccurate description of the relationship between surface deflections and moments along the body axis. Although there have been a few attempts to retain a fully nonlinear mapping $g(\delta, \theta)$ in the formulation of the control allocation problem [18], in this paper a time-varying affine model of the form

$$g(\delta, \theta) = G(\theta)\delta + \varepsilon(\theta) \tag{3}$$

will be adopted [15]. The formulation in Eq. (3), providing an intercept correction to the standard linear model, results in an improved approximation of the control effective mapping, while at the same time retaining most of the computational advantages of the linear formulation. Denoting by y_{des} the quantity

$$y_{\text{des}} = \dot{\omega}_{\text{des}} - f(\omega, \theta) - \varepsilon(\theta) \tag{4}$$

in accordance with Eq. (2), the *static control allocation problem* for the model in Eqs. (1–3) is defined as that of finding δ_{cmd} such that

$$G(\theta)\delta_{\text{cmd}} = y_{\text{des}}$$

subject to a set of constraints of the form

$$\delta_{\min} \le \delta_{\text{cmd}} \le \delta_{\max}$$

where the inequalities are intended to hold componentwise.

As flight control systems are implemented in digital computers, control allocation is implemented as a discrete time algorithm. As a result, using a first-order Euler approximation of the derivative of the commanded deflection, it is possible to incorporate constraints

$$|\dot{\delta}_{\text{cmd}}| \le \dot{\delta}_{\max}$$

on the rate of change of $\delta_{\text{cmd}}(t)$ in the above formulation. This is accomplished by requiring that at each time step t_k the commanded deflection satisfies time-varying constraints of the form

$$\bar{\delta}_{\min}(k) \le \delta_{\text{cmd}}(t_k) \le \bar{\delta}_{\max}(k), \qquad k = 1, 2, \ldots$$

where

$$\bar{\delta}_{\min}(k) = \max[\delta_{\min}, \delta_{\text{cmd}}(t_{k-1}) - \dot{\delta}_{\max} T_s]$$
$$\bar{\delta}_{\max}(k) = \min[\delta_{\max}, \delta_{\text{cmd}}(t_{k-1}) + \dot{\delta}_{\max} T_s]$$

and T_s is the sampling time of the algorithm.

For the solution of the static control allocation problem, a wide variety of constrained optimization techniques is available (see the excellent survey by Bodson [13], and references therein). As a representative of these techniques, we adopt the mixed optimization scheme with intercept correction (MOIC) [15,17] for a comparative study with the novel model-predictive algorithm developed in Sec. III. The algorithm described in the cited works, building upon the work of Buffington [28] and Bodson [13], poses the control allocation problem as the 1-norm mixed optimization problem

$$\min_{\delta} J_M = \min_{\delta}\{||y_{\text{des}} - G(\theta)\delta||_1 + \lambda||W_p(\delta - \delta_p)||_1\} \tag{5}$$

In Eq. (5), the parameter λ is used as a relative weight between the primary objective of the control allocation problem, and a secondary objective, consisting in driving the commanded deflection toward a "preferred" vector δ_p whenever there is sufficient control authority left.

III. Model-Predictive Dynamic Control Allocation

In this section, we devote our attention to the main problem considered in this paper, that is, the development of an algorithm for the allocation of the desired control vector y_{des} in the presence of nonnegligible actuator dynamics. The proposed approach replaces the static mixed optimization formulation given in the previous section with a two-stage moving-horizon optimization process. In the first stage, referred to as *target calculation*, a feasible inverse of the actuator dynamics is computed by means of a mixed quadratic optimization over a finite horizon, and a reference trajectory in the state and input spaces of the actuator dynamics is made available to the controller. Then, the control allocation problem is posed as a receding-horizon optimal trajectory tracking problem (the reference trajectory being the one calculated at the previous stage), and solved by means of model-predictive control techniques. The overall dynamic control allocation architecture, shown in Fig. 2, is composed of a prediction module that has the role of computing a prediction of the desired control reference trajectory y_{des} over the given finite horizon, a target calculation module that computes a feasible inverse of the actuator dynamics, and a model-predictive tracking controller that solves the receding-horizon optimal control problem.

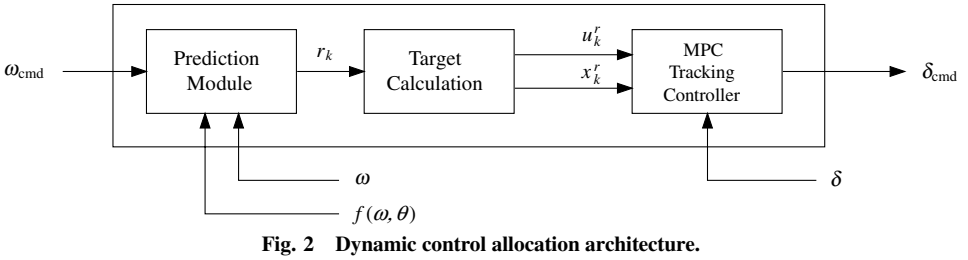

Fig. 2 Dynamic control allocation architecture.

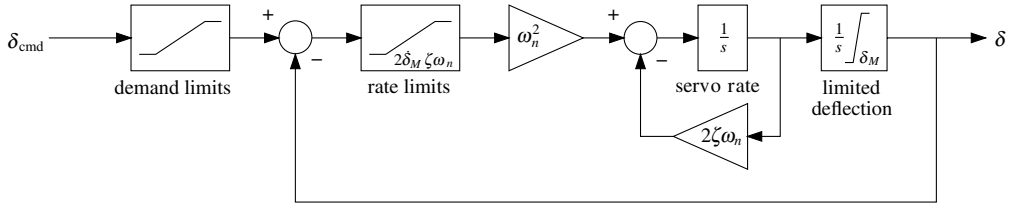

Fig. 3 Constrained actuator dynamics.

A. Actuator Dynamics

A constrained second-order linear system of the form shown in Fig. 3 is assumed for the nonnegligible dynamics of each actuator governing the deflection of the corresponding control surface. The overall actuator dynamics, including the mapping between actuator position and resulting control angular acceleration about the body axis, can be represented as a system of the form

$$\dot{x} = A_{act}x + B_{act}\delta_{cmd}, \qquad \delta = C_{act}x, \qquad y = G(\theta)C_{act}x$$
$$x_{min} \le x \le x_{max}, \qquad \delta_{min} \le \delta_{cmd} \le \delta_{max} \tag{6}$$

with state $x = \begin{bmatrix} \delta^T & \dot{\delta}^T \end{bmatrix} \in \mathbb{R}^{2m}$, input $\delta_{cmd} \in \mathbb{R}^m$, auxiliary output $\delta \in \mathbb{R}^m$, and regulated output $y \in \mathbb{R}^3$ given by the angular acceleration produced by the control surfaces. Limits in the position and rate of change of control surfaces are included as a set of constraints on the state, together with constraints on commanded deflections. The specific control allocation problem resulting from augmentation with actuator dynamics is referred to as the *dynamic control allocation problem*, stated as follows:

1. Dynamic Control Allocation Problem

Given the reference trajectory y_{des}, find the input command δ_{cmd} such that the output $y \in \mathbb{R}^3$ of the constrained system in Eq. (6) tracks y_{des} as closely as possible.

To proceed with the development of the algorithm, we first discretize the actuator dynamics, and write their sampled-data equivalent (with a minor abuse of notation) as

$$x_{k+1} = Ax_k + Bu_k \qquad y_k = C_k x_k, \qquad x_{min} \le x_k \le x_{max}$$
$$u_{min} \le u_k \le u_{max} \tag{7}$$

where $x_k = x(t_k) \in \mathbb{R}^{2m}$ is the sampled state, $u_k = \delta_{cmd}(t_k) \in \mathbb{R}^m$ is the control input, $y_k \in \mathbb{R}^3$ is the regulated output, and $C_k = G(\theta_k)C_{act}$. Note that, although it is reasonable to assume that the actuators dynamics are time invariant, and so the pair (A_{act}, B_{act}) is fixed, the parameter θ_k should be considered time varying to allow for variations in control effective mapping over the entire envelope of flight conditions. For ease of notation, the constraints in Eq. (7) will be written as set-membership constraints of the form $x_k \in \mathcal{X} \subset \mathbb{R}^{2m}$, $u_k \in \mathcal{U} \subset \mathbb{R}^m$, where $\mathcal{X}$ and $\mathcal{U}$ are polytopes. To keep the notation streamlined, the reference to be tracked will henceforth be denoted by $r_k = y_{des}(t_k) \in \mathcal{Y}$, where the polytope $\mathcal{Y} \subset \mathbb{R}^3$ accounts for limitations in the angular acceleration provided by the actuators.

The following assumptions ensure solvability of the problem and well-posedness of the receding-horizon optimization problem:

Assumption 1: (A, B) is controllable, and (A, C_k) is observable for any $k \ge 0$

Assumption 2: At each time t_k the reference r_k is known over the horizon $\mathcal{I}_k = [k, k+N]$, that is, given a fixed $N \in \mathbb{N}$, the sequence $\{r_{k+i}\}_{i=0}^N$ is available to the controller for all $k \in \mathbb{N}$.

B. Target Calculation

Because the reference trajectory is time varying, the first step in solving the dynamic control allocation problem is to compute, over a given finite horizon of length $N \in \mathbb{N}$, an input sequence and a corresponding feasible state trajectory yielding as output the given reference sequence $\{r_k\}$. In other words, as is the case for any tracking problem, the computation of a feasible inverse of Eq. (7) is required. Borrowing terminology from the literature on model-predictive control [29,30], the *target calculation problem* is stated as follows:

1. Target Calculation Problem

For any $k \ge 1$, given a sequence $\{r_{k+i}\}_{i=0}^N \subset \mathcal{Y}$ and an initial state x_k^r such that $C_k x_k^r = r_k$, find a sequence $\{u_{k+i}^r\}_{i=0}^{N-1} \subset \mathcal{U}$ such that the corresponding solution of

$$x_{k+i+1}^r = Ax_{k+i}^r + Bu_{k+i}^r \qquad y_{k+i}^r = C_{k+i}x_{k+i}^r \tag{8}$$

yields $y_{k+i}^r = r_{k+i}$ for all $i = 1, \ldots, N$, and satisfies $\{x_{k+i}^r\}_{i=1}^N \subset \mathcal{X}$.

Once a solution of the target calculation problem is available, then by means of the simple coordinate transformation $\tilde{x}_k = x_k - x_k^r$, $\tilde{u}_k = u_k - u_k^r$, and $\tilde{y}_k = y_k - r_k$, the tracking problem for the system in Eq. (7) can be cast as the problem of *stabilizing the equilibrium* $\tilde{x} = 0$ *of the error system*

$$\tilde{x}_{k+1} = A\tilde{x}_k + B\tilde{u}_k \qquad \tilde{y}_k = C_k\tilde{x}_k \tag{9}$$

subject to the time-varying constraints

$$x_{min} - x_k^r \le \tilde{x}_k \le x_{max} - x_k^r \qquad u_{min} - u_k^r \le \tilde{u}_k \le u_{max} - u_k^r$$

To ensure that the origin is in the interior of the feasible set for all feasible sequences $\{x_k^r, u_k^r\}$, the constraints of the target calculation problem are tightened by means of positive offsets, that is, the original set of constraints are replaced by

$$x_k^r \in \mathcal{X} = \{x\colon x_{min} + \Delta x_{min} \le x \le x_{max} - \Delta x_{max}\}$$
$$u_k^r \in \mathcal{U} = \{u\colon u_{min} + \Delta u_{min} \le u \le u_{max} - \Delta u_{max}\}$$

where the components of the vectors $\Delta u_{min}, \ldots, \Delta x_{max}$ are strictly positive.

Several issues need to be addressed regarding the feasibility of both the target calculation problem and the stabilization of the constrained system in Eq. (9). As $\dim(u_k) > \dim(y_k)$, the required feasible input/state reference trajectory, if it exists, may not be uniquely defined. Moreover, the target calculation problem may not have a solution $\{x_k^r, u_k^r\}$ that does not temporarily violate the given constraints, even if the reference satisfies $r_k \in \mathcal{Y}$ for all $k \ge 0$. As a matter of fact, when implemented in the flight control system, the reference trajectory generated by the dynamic inversion module is not even guaranteed to satisfy $r_k \in \mathcal{Y}$ for all $k \ge 0$, as the controller may at times require unattainable control efforts. It is precisely the role of the target calculation algorithm to compute a "steady-state solution" of actuator dynamics that achieves a compromise between fidelity of response and exact fulfillment of constraints. For these reasons, it is convenient to cast the target calculation as an optimization problem with mixed constraints, where the requirements that $\{x_k^r\} \subset \mathcal{U}$ and $y_k^r = r_k$ are relaxed by means of the introduction of appropriate slack variables.

We begin by noticing that, due to the zero-order hold introduced with the sampling of the continuous-time model, the system in Eq. (7) has relative degree one, that is, the matrix $C_{act}B$ is nonsingular. This, in turn, implies that the matrix $C_kB = G(\theta_k)C_{act}B$ has full rank for any $k \ge 0$. To find the state/input target sequence in each moving interval $\mathcal{I}_k = [k, k+N]$, $k \ge 0$, we employ the following recursive algorithm: let the vectors $\boldsymbol{x}_k^r \in \mathbb{R}^{2mN}$, $\boldsymbol{u}_k^r \in \mathbb{R}^{mN}$, and $\boldsymbol{y}_k^r \in \mathbb{R}^{3N}$ be defined, respectively, as

$$\boldsymbol{x}_k^r = \begin{pmatrix} x_{k+N}^r \\ x_{k+N-1}^r \\ \vdots \\ x_{k+1}^r \end{pmatrix}, \qquad \boldsymbol{u}_k^r = \begin{pmatrix} u_{k+N-1}^r \\ u_{k+N-2}^r \\ \vdots \\ u_k^r \end{pmatrix}, \qquad \boldsymbol{y}_k^r = \begin{pmatrix} y_{k+N}^r \\ y_{k+N-1}^r \\ \vdots \\ y_{k+1}^r \end{pmatrix}$$

for all $k \geq 0$. Assuming that the initial state x_k^r, is known, the solution of Eq. (8) satisfies an equation of the form

$$\boldsymbol{x}_k^r = \boldsymbol{F}x_k^r + \boldsymbol{G}\boldsymbol{u}_k^r \qquad \boldsymbol{y}_k^r = \boldsymbol{H}_k x_k^r + \boldsymbol{K}_k \boldsymbol{u}_k^r \tag{10}$$

Because the matrix $\boldsymbol{K}_k$ can be easily shown to have full rank, the target trajectory over the moving interval $\mathcal{I}_k$ is computed solving the underdetermined system of equations

$$\boldsymbol{K}_k \boldsymbol{u}_k^r = \boldsymbol{r}_k \text{ subject to } \boldsymbol{u}_k^r \in \boldsymbol{U} \quad \text{and} \quad \boldsymbol{G}\boldsymbol{u}_k^r + \boldsymbol{F}x_k^r \in X \tag{11}$$

where $\boldsymbol{r}_k \in \mathbb{R}^{3N}$ is given by

$$\boldsymbol{r}_k = \begin{pmatrix} r_{k+N} \\ r_{k+N-1} \\ \vdots \\ r_k \end{pmatrix} - \boldsymbol{H}_k x_k^r$$

and $\boldsymbol{U} \subset \mathbb{R}^{mN}$ and $\boldsymbol{X} \subset \mathbb{R}^{2mN}$ denote the corresponding polytopes for $\boldsymbol{u}_k^r$ and $\boldsymbol{x}_k^r$, respectively. The state trajectory $\boldsymbol{x}_k^r$, yielding the initial state x_k^r for the moving interval $\mathcal{I}_{k+1}$ at the next step, is computed using the first equation in Eq. (10). For the first interval $\mathcal{I}_0$, the initial state x_0^r is not assigned, and it can be determined by the optimization algorithm to help generate a feasible solution.

To overcome the problem associated with unfeasible solutions of Eq. (11), it is convenient to relax some constraints by means of the introduction of an appropriate number of slack variables. Assuming that preference is given to satisfying the constraints on the commanded input, the target optimization can be cast as a quadratic optimization problem with mixed constraints [29,30], with cost function

$$\begin{aligned} J_k^{\text{tar}} = {} & \boldsymbol{r}_k^{sT}\boldsymbol{P}_r\boldsymbol{r}_k^s + \boldsymbol{x}_k^{sT}\boldsymbol{Q}_r\boldsymbol{x}_k^s + (\boldsymbol{u}_k^r - \boldsymbol{u}_k^p)^T\boldsymbol{R}_r(\boldsymbol{u}_k^r - \boldsymbol{u}_k^p) \\ & + \boldsymbol{w}_1^T\boldsymbol{r}_k^s + \boldsymbol{w}_2^T\boldsymbol{x}_k^s \end{aligned} \tag{12}$$

where $\boldsymbol{r}^s \in \mathbb{R}^{3N}$ and $\boldsymbol{x}_k^s \in \mathbb{R}^{2mN}$ are vectors of nonnegative slack variables, $\boldsymbol{P}_r$, $\boldsymbol{Q}_r$, and $\boldsymbol{R}_r$ are symmetric and positive definite matrices of appropriate dimension, and $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$ are vectors of positive weights. The target calculation problem is solved by minimizing J_k^{tar}, subject to

$$\begin{pmatrix} -\boldsymbol{r}_k^s \\ -\boldsymbol{x}_k^s \\ \boldsymbol{u}_k^r \\ -\boldsymbol{u}_k^r \\ \boldsymbol{K}_k\boldsymbol{u}_k^r - \boldsymbol{r}_k^s \\ -\boldsymbol{K}_k\boldsymbol{u}_k^r - \boldsymbol{r}_k^s \\ \boldsymbol{G}\boldsymbol{u}_k^r + \boldsymbol{F}x_k^r - \boldsymbol{x}_k^s \\ -\boldsymbol{G}\boldsymbol{u}_k^r + \boldsymbol{F}x_k^r - \boldsymbol{x}_k^s \end{pmatrix} \leq \begin{pmatrix} \boldsymbol{0} \\ \boldsymbol{0} \\ (u_{\max} - \Delta u_{\max})\boldsymbol{1} \\ -(u_{\min} + \Delta u_{\min})\boldsymbol{1} \\ \boldsymbol{r}_k \\ -\boldsymbol{r}_k \\ (y_{\max} - \Delta y_{\max})\boldsymbol{1} \\ -(y_{\min} + \Delta y_{\min})\boldsymbol{1} \end{pmatrix} \tag{13}$$

where $\boldsymbol{0}$ and $\boldsymbol{1}$ denote, respectively, vectors

$$\boldsymbol{0} = \begin{pmatrix} 0 & 0 & \cdots & 0 \end{pmatrix}^T, \qquad \boldsymbol{1} = \begin{pmatrix} 1 & 1 & \cdots & 1 \end{pmatrix}^T$$

of appropriate dimension. The vector $\boldsymbol{u}_k^p$ represents a preferred value of the control reference that offers the possibility to optimize additional performance criteria whenever there is sufficient control authority to do so. A possible choice for $\boldsymbol{u}_k^p$, apart from the obvious $\boldsymbol{u}_k^p = 0$, is given by the solution of the unconstrained weighted minimum energy problem

$$\boldsymbol{u}_k^p = W^{-1}\boldsymbol{K}_k^T(\boldsymbol{K}_k W^{-1}\boldsymbol{K}_k^T)^{-1}\boldsymbol{r}_k$$

where $\boldsymbol{W} > 0$ is a weighting matrix. Alternatively, the target calculation problem can be posed as the linear programming (LP) problem

$$\min_{\boldsymbol{u}^r, \boldsymbol{x}^s, \boldsymbol{r}^s} J_k^{\text{tar}} = \begin{pmatrix} w_1^T & w_2^T \end{pmatrix} \begin{pmatrix} \boldsymbol{r}_k^s \\ \boldsymbol{x}_k^s \end{pmatrix} \tag{14}$$

subject to Eq. (13), which corresponds to the constrained minimization of the functional

$$\min_{\boldsymbol{u}^r} J_k^{\text{tar}} = ||\boldsymbol{w}_1^T(\boldsymbol{K}_k\boldsymbol{u}_{k-1}^r - \boldsymbol{r}_k)||_1$$

In case $J_k^{\text{tar}} > 0$, the problem is not feasible, and the solution yields the best approximation in terms of the 1-norm of the error that does not violate the limits on the commanded input. If $J_k^{\text{tar}} = 0$, the problem is feasible, and nonuniqueness of the solution can be exploited to drive the commanded deflections $\boldsymbol{u}_k^r$ toward $\boldsymbol{u}_k^p$. In that case, the subobjective function is chosen as the weighted 1-norm problem

$$\begin{aligned} \min_{\boldsymbol{u}_k^r} J_k^p = ||\boldsymbol{w}_3^T(\boldsymbol{u}_k^r - \boldsymbol{u}_k^p)||_1 \text{ subject to } \boldsymbol{K}_k\boldsymbol{u}_k^r = \boldsymbol{r}_k \\ \boldsymbol{u}_k^r \in \boldsymbol{U} \quad \text{and} \quad \boldsymbol{G}\boldsymbol{u}_k^r + \boldsymbol{F}x_{(k-1)N}^r \in \boldsymbol{X} \end{aligned} \tag{15}$$

where $\boldsymbol{w}_3$ is a vector of positive weights. The subobjective in Eq. (15) can also be formulated as an LP problem with structure similar to Eq. (14).

The solution of the target optimization provides reference states, reference inputs, and reference output $y_k^r = C_k x_k^r$ for the actuator dynamics. The optimization guarantees that y_k^r remains close to the actual reference r_k (in the 1-norm or the 2-norm sense), when the constraints are active. The reference trajectory (x_k^r, u_k^r) is then employed in the second stage of the dynamic control allocation algorithm, where a control that asymptotically tracks y_k^r is computed using a standard MPC algorithm.

C. Prediction of the Control Reference Trajectory

In order for the target calculation module to generate $\{x_{k+i}^r\}_{i=1}^N$ and $\{u_{k+i}^r\}_{i=0}^{N-1}$, the reference sequence $\{r_{k+i}\}_{i=0}^N$ must be available over any moving horizon, that is, the sampled-data (in this section, to avoid introducing additional cumbersome notation, we have denoted by $\{s(t_k)\}$ the sequence obtained sampling a continuous-time signal $s(t)$ at times t_k, $k \in \mathbb{N}$, where $T_s = t_{k+1} - t_k$ is the sampling period) sequence $\{y_{\text{des}}(t_i)\}$ must be known in advance over $[t_k, t_{k+N}]$ for any $k \geq 0$. Note, however, that at time $t = t_k$, the sequence $\{y_{\text{des}}(t_{k+i})\}_{i=1}^N$ is not available a priori, because $y_{\text{des}}(t)$ is the output of the 2-DOF controller in closed loop with the vehicle dynamics, under the assumption that the actuator dynamics are negligible. As a result, at any $k \geq 0$ a prediction $\{\hat{r}_{k+i|k}\}_{i=1}^N$ must be employed over the receding horizon $\mathcal{I}_k$ in place of the sequence $\{r_k\}_{i=1}^N$. To find the desired prediction $\hat{r}_{k+i|k}$, we begin by recalling from Eq. (4) that

$$y_{\text{des}}(t_i) = \dot{\omega}_{\text{des}}(t_i) - f[\omega(t_i), \theta(t_i)] - \varepsilon[\theta(t_i)], \qquad i = k, \ldots, k+N \tag{16}$$

At the initial time t_k of the moving horizon $\mathcal{I}_k$, the values of $\omega(t_k)$ and $\theta(t_k)$ are known, and thus $y_{\text{des}}(t_k)$ is available. As a result, it is possible to set $\hat{r}_{k|k} = r_k$. The commanded angular velocity vector $\omega_{\text{cmd}}(t)$ is assumed to be available from the trajectory-planning database of the flight control system for the entire duration of the specific flight mode (i.e., ascent, reentry, terminal area energy management, and so on). Starting with the knowledge of $\omega_{\text{cmd}}(t_{k+i})$ over $i \in [0, N]$, the prediction $\{\hat{r}_{k+i|k}\}_{i=1}^N$ is computed as follows: first, $\theta(t_k)$ is kept constant over the prediction horizon, that is, $\theta(t_{k+i}) = \theta(t_k)$, $i = 1, \ldots, N$. This, in turn, determines the intercept correction $\varepsilon_{k+i} = \varepsilon[\theta(t_k)]$, $i = 1, \ldots, N$, in Eq. (16). A prediction of the output $\dot{\omega}_{\text{des}}$ of the outer-loop controller is computed assuming that the dynamic inversion algorithm achieves exact cancellation of the

nonlinear term $f(\omega, \theta)$. In this case, $\omega_{\text{des}} = \omega$, and thus the transfer function between ω_{cmd} and $\dot{\omega}_{\text{des}}$ is given by $sW_m(s)$. This yields a prediction $\hat{\dot{\omega}}_{\text{des}}$ of $\dot{\omega}_{\text{des}}$ as the output of the linear system

$$\hat{\omega}(t_{k+i+1}) = \hat{\omega}(t_{k+i}) + \frac{1}{T_s} K_m[\omega_{\text{cmd}}(t_{k+i}) - \hat{\omega}(t_{k+i})]$$

$$\hat{\dot{\omega}}_{\text{des}}(t_{k+i}) = K_m[\omega_{\text{cmd}}(t_{k+i}) - \hat{\omega}(t_{k+i})], \qquad i = 1, \ldots, N$$

$$\hat{\omega}(t_k) = \omega(t_k)$$

where $\hat{\omega}(t_{k+i})$ is a prediction of $\omega(t_{k+i})$, and $T_s = t_{k+1} - t_k$. Finally, the contribution of the dynamic inversion module is computed replacing ω with prediction obtained above, letting

$$\hat{f}[\omega(t_{k+i}), \theta(t_{k+i})] = f[\hat{\omega}(t_{k+i}), \theta(t_k)], \qquad i = 1, \ldots, N$$

which is a valid approximation as long as the vehicle trajectory remains close to the reference.

D. Model-Predictive Tracking

After the target input and state sequences have been computed, the control allocation problem at time k is posed as a sequential model-predictive tracking problem. Specifically, we consider the minimization of the cost function

$$J_k^{\text{tr}} = \sum_{i=0}^{N-1} \{(y_{k+i|k} - y_{k+i}^r)^T Q (y_{k+i|k} - y_{k+i}^r) + (u_{k+i} - u_{k+i}^r)^T \times R(u_{k+i} - u_{k+i}^r)\} + (x_{k+N|k} - x_{k+N}^r)^T \Psi_{k+N} (x_{k+N|k} - x_{k+N}^r)$$

where $Q > 0$, $R > 0$, and $\Psi_i > 0$ for all $i \geq 0$. The variable $x_{k+i|k}$ is the predicted state of the system in Eq. (7) at the step k, with $x_{k|k} = x_k$. Defining the error variables

$$\bar{x}_{k+i} = x_{k+i|k} - x_{k+i}^r, \qquad \bar{u}_{k+i} = u_{k+i} - u_{k+i}^r$$

the cost function is rewritten as

$$J_k^{\text{tr}} = \sum_{i=0}^{N-1} (\bar{x}_{k+i}^T Q_{k+i} \bar{x}_{k+i} + \bar{u}_{k+i}^T R \bar{u}_{k+i}) + \bar{x}_{k+N}^T \Psi_{k+N} \bar{x}_{k+N} \tag{17}$$

where $Q_i = C_i^T Q C_i$, and the problem is cast as the minimization of J_k^{tr}, subject to

$$\bar{x}_{k+i+1} = A\bar{x}_{k+i} + B\bar{u}_{k+i} \qquad D\bar{u}_{k+i} \leq d_{k+i}, \qquad E\bar{x}_{k+i} \leq e_{k+i}$$
$$i = 0, \cdots, N-1 \tag{18}$$

where

$$D = \begin{pmatrix} I \\ -I \end{pmatrix}, \qquad E = \begin{pmatrix} C_k \\ -C_k \end{pmatrix}, \qquad d_{k+i} = \begin{pmatrix} u_{\max} - u_{k+i}^r \\ -u_{\min} + u_{k+i}^r \end{pmatrix}$$
$$e_{k+i} = \begin{pmatrix} y_{\max} - y_{k+i}^r \\ -y_{\min} + y_{k+i}^r \end{pmatrix}$$

The optimal control sequence $u_{k+i|k}^* = \bar{u}_{k+i} + u_{k+i}^r$ yields the implicit model-predictive control allocation policy $\delta_{\text{cmd}}(t_k) = u_{k|k}^*$. The optimization problem given by Eqs. (17) and (18) may be infeasible or difficult to solve due to the given constraints on the input and output trajectories. The problem can be alleviated by relaxing the hard constraint for the state, replacing Eq. (18) with the mixed constraints

$$D\bar{u}_{k+i} \leq d_{k+i} \qquad E\bar{x}_{k+i} \leq e_{k+i} + \varepsilon_k, \qquad i = 0, \ldots, N-1 \tag{19}$$

where it is assumed that $\bar{u}_k = 0$ and $\bar{x}_k = 0$ is feasible. The cost function of the optimization problem is augmented with a penalty on the slack variable ε_k as

$$J_k^{\text{mixed}} = J_k^{\text{tr}} + \varepsilon_k^T S \varepsilon_k \tag{20}$$

where $S > 0$, and the mixed-constrained model-predictive control allocation (MPCA) problem is posed as the minimization of J_k^{mixed}, subject to the constraints given by Eq. (19), and the first equality in Eq. (18). Because the tracking problem is now posed as a model-predictive control with soft constraints on the state, and $(\bar{x}, \bar{u}) = (0, 0)$ lies in the interior of the feasible set, asymptotic stability of the actuator dynamics suffices to guarantee feasibility of the optimization problem, and asymptotic convergence to the origin, as shown in the works of Rawlings et al. [31,32]. In any case, a careful selection of the weights of the objective function is instrumental in improving the controller performance, and several methods exist to enforce asymptotic convergence through an appropriate selection of the terminal constraint (see the excellent survey [33] and references therein). The main result of the section is summarized as follows:

Proposition III.1: For the system given by Eq. (7), the solution of the moving-horizon sequential optimization problems with cost functions and constraints given by Eqs. (12), (13), (19), and (20), respectively, yields a control sequence $\{u_k\}$ such that $u_k \in \mathcal{U}$ for all $k \geq 0$, and $\lim_{k\to\infty} ||y_k - y_k^r|| = 0$.

E. Linear Complementarity Problem Formulation

In implementation, linear programming methods may present an advantage over sequential quadratic optimization methods, in that LP problems can be solved within a finite number of iterations [34]. Model-predictive control problems with mixed constraints can be cast into linear complementarity problems (LCP) of the following form [25,35]: Given a vector $q \in \mathbb{R}^n$ and matrix $M \in \mathbb{R}^{n\times n}$, find two vectors s and z satisfying

$$s - Mz = q, \qquad s, z \geq 0, \qquad s^T z = 0$$

To transform our specific MPCA problem to the LCP formulation, we first express the predicted state error as a function of the initial condition and the input error as follows:

$$\bar{x}_{k+i|k} = A^i \bar{x}_k + \sum_{j=0}^{i-1} B\bar{u}_{k+j}$$

and collect the sequence $\{\bar{u}_{k+i}\}_{i=0}^{N-1}$ into the vector $\boldsymbol{u}_k$. Accordingly, the cost function in Eq. (17) can then be expressed as the quadratic form

$$J(\boldsymbol{u}_k) = \tfrac{1}{2}\boldsymbol{u}_k^T L_k \boldsymbol{u}_k + b_k \boldsymbol{u}_k + f_k \tag{21}$$

for some matrix L_k and some vectors b_k and f_k which depend on the initial state error $\bar{x}_k$. An equivalent set of constraints for $\boldsymbol{u}_k$ is given in the form

$$\boldsymbol{u}_k \geq 0, \qquad R_k \boldsymbol{u}_k \leq c_k \tag{22}$$

where R_k and c_k are, respectively, an appropriate matrix and a column vector related to the specific original constraints for $\bar{u}_k$ and $\bar{x}_k$. Let v_k and w_k be the vectors of Lagrange multipliers associated, respectively, with the first and the second set of constraints in (22), and let ε_k be the vector of slack variables. The Karush–Kuhn–Tucker (KKT) conditions [36] for Eqs. (21) and (22) read as follows:

$$R_k \boldsymbol{u}_k + \varepsilon_k = c_k, \qquad -L_k \boldsymbol{u}_k - R_k^T v_k + w_k = b_k, \qquad \boldsymbol{u}_k^T w_k = 0$$
$$v_k^T \varepsilon_k = 0, \qquad \boldsymbol{u}_k, v_k, w_k, \varepsilon_k \geq 0$$

The KKT conditions can be further expressed as the linear complementarity problem

$$s_k - M_k z_k = q_k, \qquad s_k^T z_k = 0, \qquad s_k, z_k \geq 0$$

where

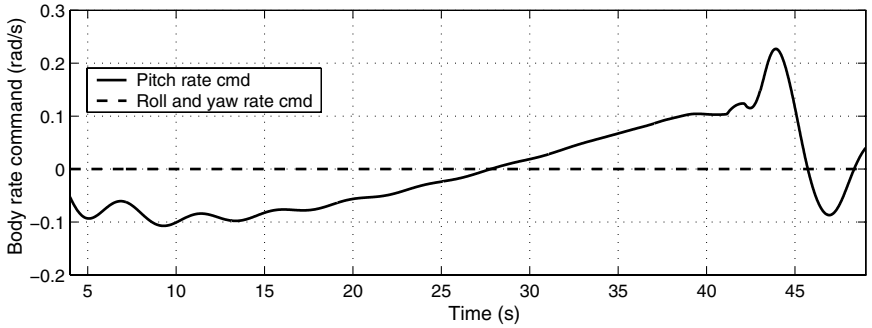

Fig. 4 Body rate commands: pitch rate command (solid); roll/yaw rate command (dashed).

$$M_k = \begin{pmatrix} 0 & -R_k \\ R_k^T & L_k \end{pmatrix}, \qquad q_k = \begin{pmatrix} c_k \\ b_k \end{pmatrix}, \qquad s_k = \begin{pmatrix} \varepsilon_k \\ w_k \end{pmatrix}$$

$$z_k = \begin{pmatrix} v_k \\ \boldsymbol{u}_k \end{pmatrix}$$

If the matrix L_k is sign definite, convergence to the optimal solution in a finite number of iterations is guaranteed [36]. A carefully chosen initial solution can greatly reduce computational requirements. A standard choice is to let the solution of the unconstrained MPCA problem, expressed in a quadratic programming form, be the starting point. In this way, the initial point is close to the solution of an LCP problem derived from a constrained model-predictive control problem.

IV. Case Study

In this section, we present and discuss simulation results for a case study regarding the reentry vehicle model considered in Doman and Ngo [36] and Schierman et al. [27]. The specific vehicle model considered here is equipped with six control surfaces: right flap, left flap, right tail, left tail, body flap, and speed brake, with upper deflection limit 30 deg, lower deflection limit −30 deg for the first four control surfaces, and rate limit 60 deg/s for all control surfaces under nominal conditions. The speed brake has an upper limit of 70 deg and a lower limit of 0 deg, and the body flap has an upper limit of 25 deg and a lower limit of −20 deg. The length of the prediction horizon has been selected as $N = 10$, with sampling time $T_s = 0.02$ s. As a baseline for comparison, we also present results obtained with the *static* control allocation method based on the MOIC scheme of Doman et al. [15,17], briefly introduced in Sec. II. The reentry vehicle model, the model-reference control architecture, and the MPCA and MOIC control allocation schemes were implemented in Simulink®, with a fixed-step 4th-order Runge–Kutta integration algorithm. The feasible reference trajectory chosen for the comparative study exhibits change in the pitch motion only, and corresponds to an approach and landing maneuver. The segment of command trajectory employed in the simulations has duration equal to 50 sec, as shown in Fig. 4.

A. Comparative Simulation Studies Under Nominal Situations

Typical tests performed for evaluating the model-predictive control allocation scheme include a wide variation on the dynamical characteristics of the six actuators in terms of damping ratio and natural frequency. Each actuator is assumed to behave as a second-order system with amplitude and rate limits, and damping ratio ranging from 0.5 to 0.7. Different natural frequencies, ranging from 20 to 5 rad/s have been independently assigned and tested for each pair of tail and wing effectors, whereas a range from 10 and 5 rad/s has been considered for the two body effectors. Table 1 displays results from several tests in which the damping and natural frequencies are varied for the tail effectors (right and left flaps), wing effectors (right and left flaps), and the body effectors (flap and speed brake). As performance metrics in comparing large numbers of tests, we typically calculate the mean square error (MSE) and the maximum error with respect to the pitch rate reference trajectory over the entire test interval. The sample results given in Table 1 indicate that the error metrics for the model-predictive control allocation scheme are approximately 1 order of magnitude smaller than those of the MOIC scheme, depending on the test conditions. It is also worth noting that as the natural frequency ω_n decreases, the performance of MPCA varies little, whereas that of the MOIC scheme rapidly degenerates.

To illustrate performance for a typical test, we first show in Fig. 5 the pitch rate tracking error and the deflection of the tail control surfaces for a relatively benign situation, corresponding to test case 3 in Table 1. In this case, the tail effectors have sufficient authority to control the pitch motion in the presence of relatively fast and well-damped actuator dynamics, whereas the remaining control surfaces show very little activity (not shown). It can be observed that the MOIC scheme maintains the capability of achieving good tracking of the reference command, although, as expected, the MPCA algorithm yields tighter tracking. The roll and yaw rate errors are negligible with both MPCA and MOIC allocation, and thus their plot has been omitted.

Simulation results for test case 5 are presented in Figs. 6 and 7, showing, respectively, pitch and roll rate error and the deflection of control surfaces. Recall that for the specific case 4 the natural frequency of the tail, wing, and body effectors are 12, 8, and 5 rad/s, respectively, whereas the damping ratio is $\zeta = 0.5$. As seen in Fig. 6,

Table 1 Performance comparison between MPCA and MOIC in nominal conditions. W: wing effectors; T: tail effectors; B: body effectors; ζ: damping ratio; ω_n: natural frequency

Test case no.	ζ	ω_n			Max. error, pitch rate		MSE, pitch rate	
		W	T	B	MPCA	MOIC	MPCA	MOIC
1	0.7	20	12	10	1.81e − 2	6.26e − 2	3.07e − 3	1.32e − 2
2	0.7	12	8	5	2.51e − 2	7.46e − 2	5.03e − 3	1.53e − 2
3	0.7	10	8	5	3.14e − 2	7.46e − 2	5.63e − 3	2.53e − 2
4	0.5	20	12	10	1.80e − 2	5.64e − 2	3.20e − 3	1.22e − 2
5	0.5	12	8	5	3.43e − 2	7.44e − 2	5.66e − 3	2.47e − 2
6	0.5	10	8	5	3.52e − 2	7.99e − 2	5.70e − 3	2.92e − 2

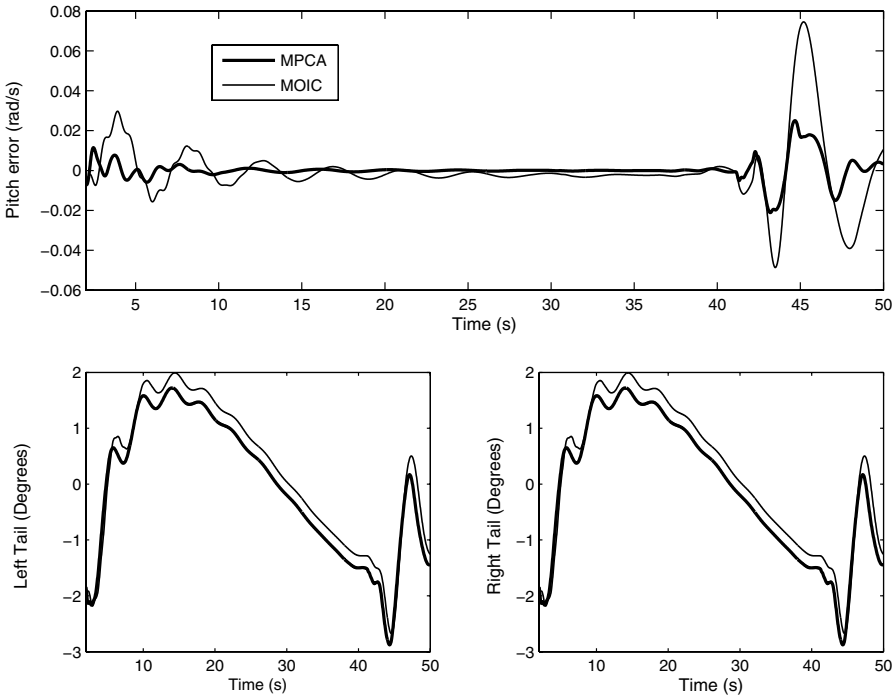

Fig. 5 Pitch rate error (upper plot) and tail control surface deflections (lower plots) for case 3 in Table 1. Thick line: MPCA; thin line: MOIC.

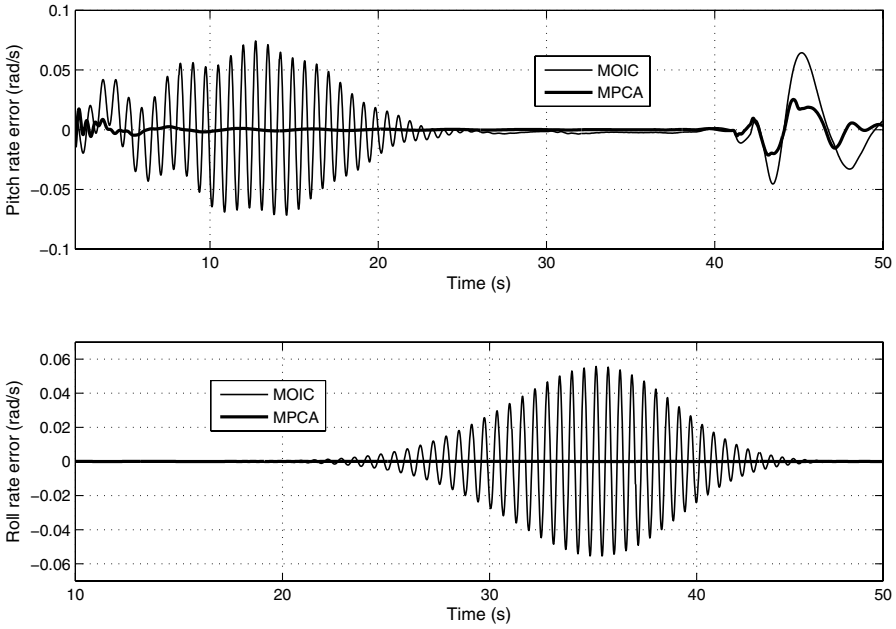

Fig. 6 Pitch rate error (upper plot) and roll rate error (lower plots) for test case 5 in Table 1. Thick line: MPCA; thin line: MOIC.

the effect of the limited bandwidth of the tail and wing effector dynamics, paired with a relatively small damping ratio, produces a sizable tracking error for the pitch rate when the static control allocation method is employed. The MPCA scheme, on the other hand, produces a behavior of the tracking error comparable to the one obtained in the previous case. The reference trajectory specifies zero roll and yaw motions, but the MOIC scheme cannot track that request for the entire duration of the test (the yaw rate error has a behavior similar to the roll rate, and is therefore omitted). For this same test, Fig. 7 shows the control effort, where distinctive differences in the action of the two schemes is apparent (because the wing flaps and tail effectors are symmetric, only the left actuators are shown.)

As a final test, we choose to present the rather stringent test condition given by case 6 in Table 1, to give an example of severe loss of performance that may occur in extreme situations. For this case study, Fig. 8 shows that the MOIC-based control loop is pushed to the verge of instability, while at the same time the tracking capability of the MPCA-based scheme is confirmed. Figure 9 shows the responses of control surfaces, where a significantly larger control action than in the previous test is apparent. Note, in particular, that whereas the MPCA allocates effectively the control effort among the actuators, MOIC produces large excursions of the left and right flap deflections related to the extremely high values of the roll rate error visible in the bottom plot in Fig. 8.

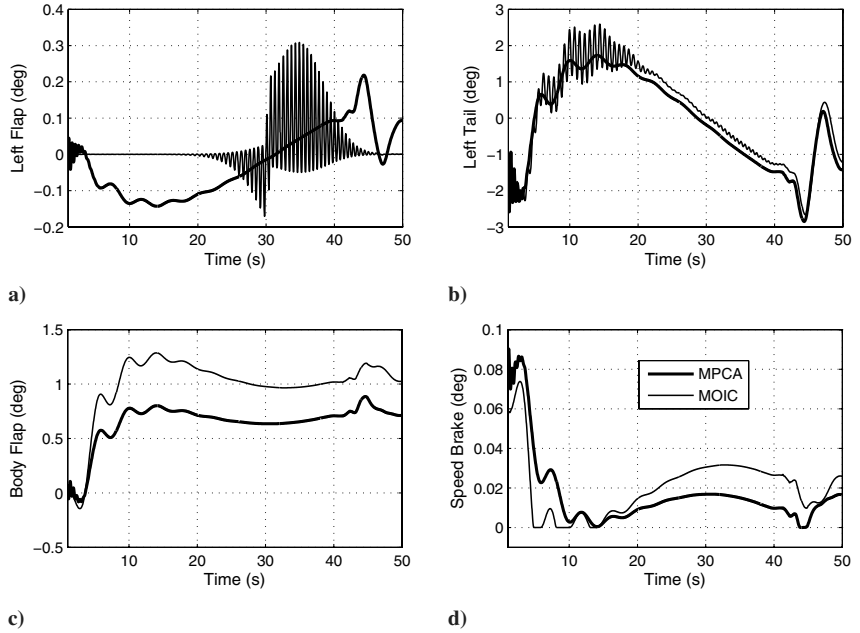

Fig. 7 Control surface deflections for case 5 in Table 1: a) left wing effector; b) left tail effector; c) body flap; d) speed brake. Thick line: MPCA; thin line: MOIC.

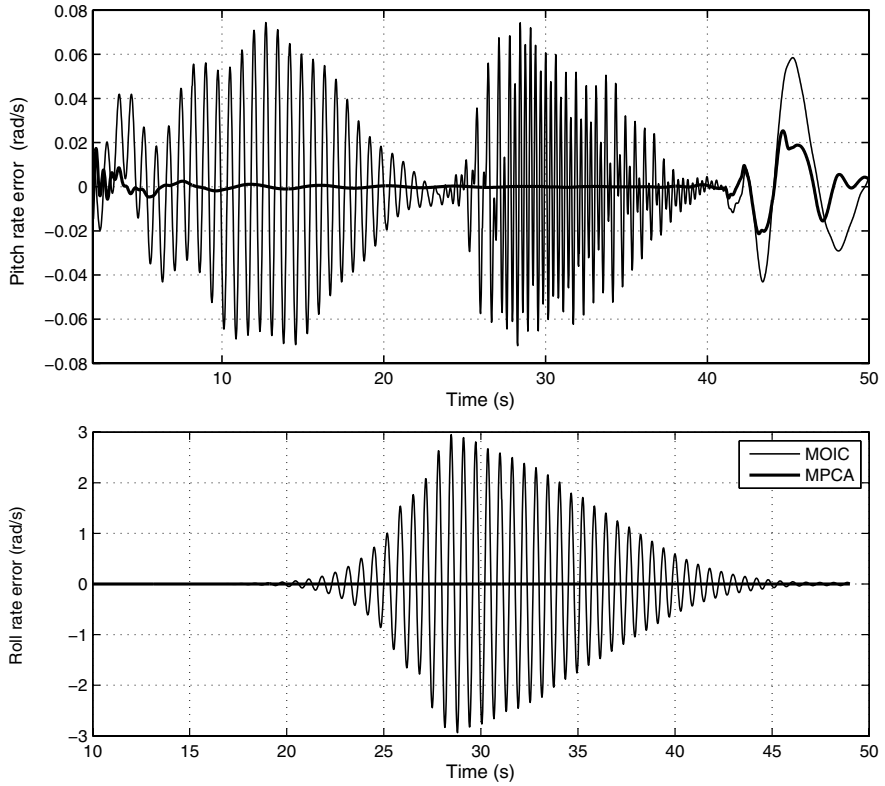

Fig. 8 Pitch rate error (upper plot) and roll rate error (lower plots) for test case 6 in Table 1. Thick line: MPCA; thin line: MOIC.

B. Comparative Simulation Studies Under Failure Situations

A similar set of damping ratios and natural frequencies as the nominal conditions are used to illustrate and compare performance of the two control allocation methods under the occurrence of failures in the presence of nonnegligible actuator dynamics. As representative of a typical failure, we have considered here the case in which a control surface remains locked at a given position, therefore providing both a loss of control effectiveness and a possible disturbance moment about all three body axes. It is assumed that the occurrence of the failure is detected by the online system identification algorithm, and the relative information passed to the control allocation module, in such a way that real-time control reconfiguration is possible. Table 2 illustrates comparative performance indices for failure situations with similar configuration as given in Table 1. Comparing the results between the two tables, notice that failures at the wing effector have less effect on the relative performance of the two algorithms, whereas failures at the tail effectors produce a significant deterioration in performance for the

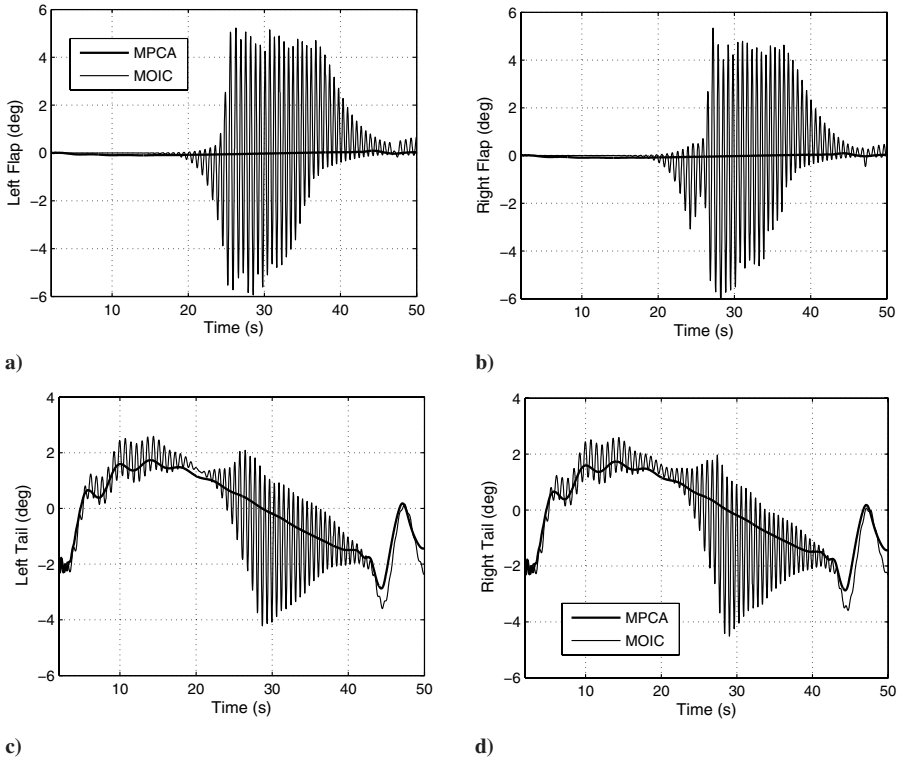

Fig. 9 Control surface deflections for case 6 in Table 1: a) left wing effector; b) right wing effector; c) left tail effector; d) right tail effector. Thick line: MPCA; thin line: MOIC.

MOIC scheme. Although this is clearly related to the particular choice of the reference command, it must be noted that, under MOIC, loss of tracking has been observed also for roll and yaw rate commands, where the control objective is merely that of regulating body rates to zero. This suggests that static allocation schemes may lose their capability of performing stable control reconfiguration when failures are accompanied with uncompensated actuator dynamics. On the other hand, the MPCA scheme shows a consistent behavior under different failure conditions and with different values of natural frequencies and damping ratios.

Similar to the nominal situation, two tests, representative of a typical and an extreme situation, respectively, are selected from the table for comparative illustration. Specifically, we consider test case 5 in Table 2, corresponding to a natural frequency of 20, 12, and 10 rad/s for the wing, tail, and body effectors, respectively. The damping ratio is $\zeta = 0.7$ for all effectors. The specific failure condition corresponds to having the right wing effector locked at 0 deg, achieved in simulation by reducing the deflection limits of the failed surface to the interval ± 0.05 deg. Because the reference command specifies a constant zero set point for the roll rate, it is expected that this type of failure can be handled by both schemes, at least if the lag introduced by the actuator dynamics is not too severe. This is confirmed by the plots in Fig. 10, showing the pitch and roll rate error obtained under MPCA and MOIC. Note the similarity with the results for the previous test case 5 under nominal conditions, but with a significantly smaller damping ratio for the actuator dynamics. For the sake of clarity, we show only the behavior of the control surfaces for the MPCA scheme in Fig. 11. It can be noted that the left wing effector maintains zero roll moment acting symmetrically with respect to the failed actuator. This is, of course, a consequence of the particular reference trajectory considered in the case study. Arguably, a different behavior must be expected when more severe damage occurs, such as the loss of a tail effector. As an example, test case 4 in Table 2 is concerned with the situation in which the right tail effector is locked at -1.5 deg, whereas the configuration of the actuator dynamics is the same as the nominal case study vi of Table 1. This combination poses an extreme challenge to the control system, which results in loss of closed-loop stability for the static control allocation scheme. Figures 12 and 13 show, respectively, pitch and roll error and control surface deflections for the MPCA scheme. An acceptable tracking error is maintained throughout the entire duration of the test, as the algorithm is capable of reconfiguring

Table 2 Performance comparison between MPCA and MOIC in failure conditions. W: wing effectors; T: tail effectors; B: body effectors; ζ: damping ratio; ω_n: natural frequency

Test case no.	ζ	ω_n			Max error, pitch rate		MSE, pitch rate	
		W	T	B	MPC	MOIC	MPC	MOIC
1	0.7	20[a]	12	10	2.09e − 2	6.26e − 2	3.8e − 3	1.32e − 2
2	0.7	12[a]	8	5	2.96e − 2	7.46e − 2	6.40e − 3	1.53e − 2
3	0.5	20[a]	12	10	2.10e − 2	5.64e − 2	3.90e − 3	1.22e − 2
4	0.5	12[a]	8	5	5.36e − 2	unstable	7.60e − 3	unstable
5	0.7	20	12[a]	10	2.31e − 2	6.72e − 2	5.5e − 3	1.58e − 2
6	0.7	12	8[a]	5	4.18e − 2	4.07e − 1	1.23e − 2	1.12e − 1
7	0.5	20	12[a]	10	2.27e − 2	6.16e − 1	5.4e − 3	1.80e − 1
8	0.5	12	8[a]	5	6.08e − 2	2.53	1.50e − 2	7.07e − 1

[a]A failed control surface.

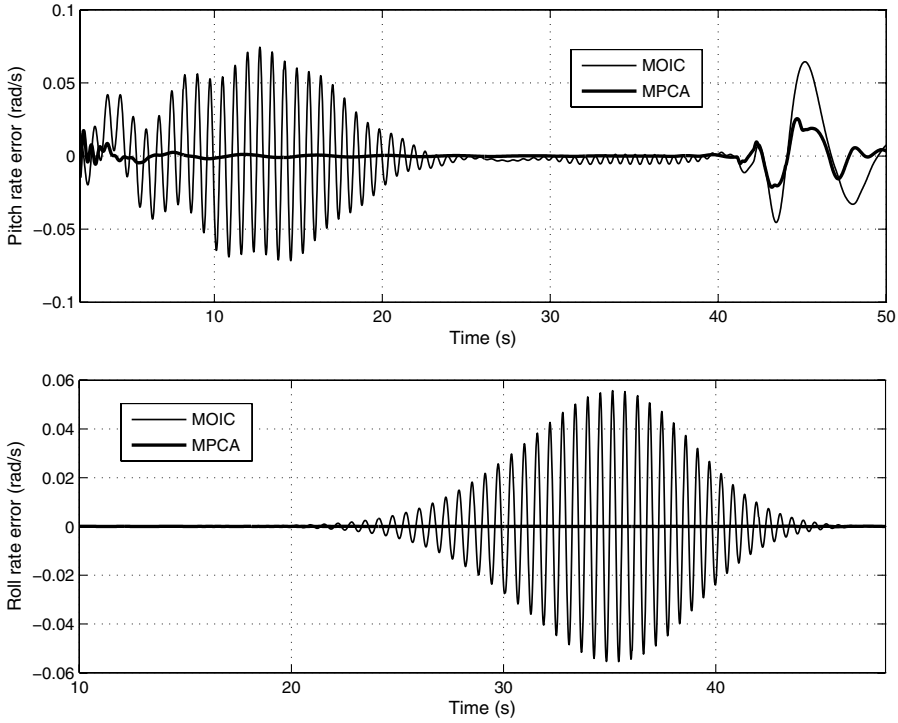

Fig. 10 Pitch rate error (upper plot) and roll rate error (lower plot) for test case 5 in Table 2. Failure occurring to the right wing flap, locked at ±0.05 deg. Thick line: MPCA; thin line: MOIC.

control authority of the vehicle. In comparison, erratic behavior of control surfaces for the MOIC scheme is shown in Fig. 14, from which it is evident that the closed-loop system has been driven to instability.

V. Conclusions

A dynamic control allocation scheme for asymptotically stable tracking of time-varying reference trajectories has been proposed to

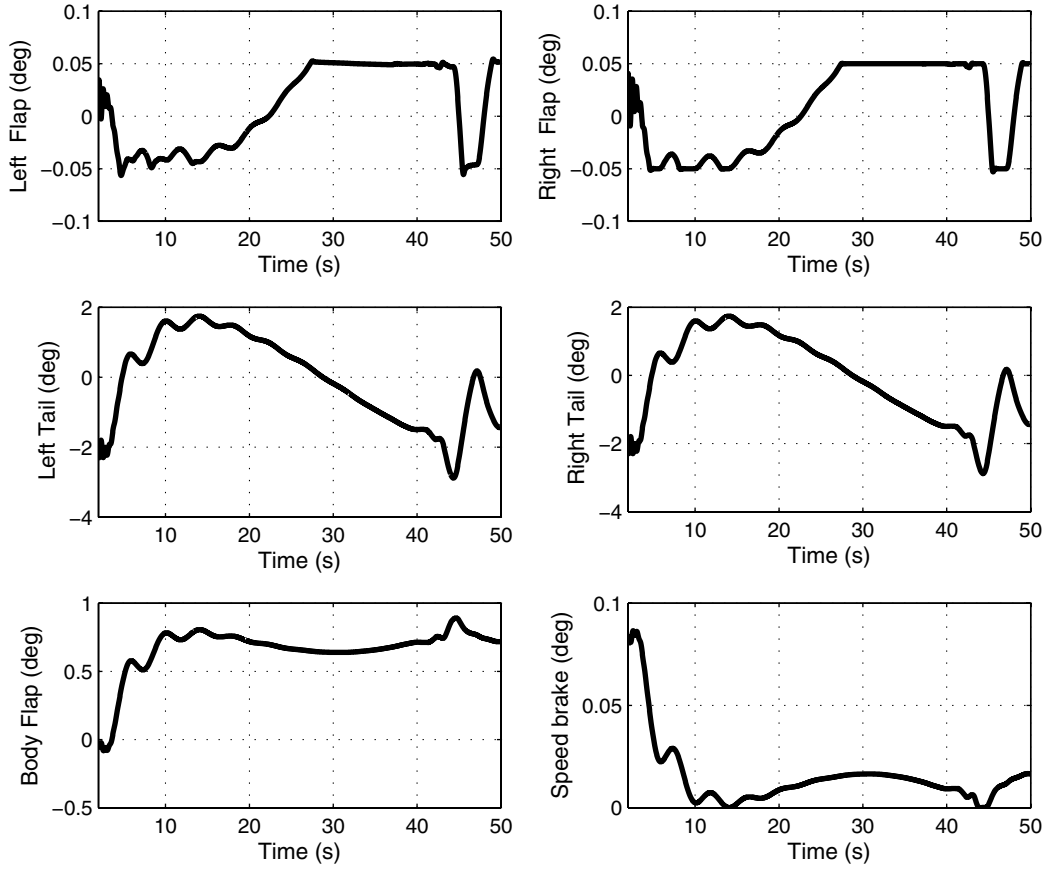

Fig. 11 Control surface deflections for case 5 in Table 2. Failure occurring to the right wing flap, locked at ±0.05 deg. MPCA scheme.

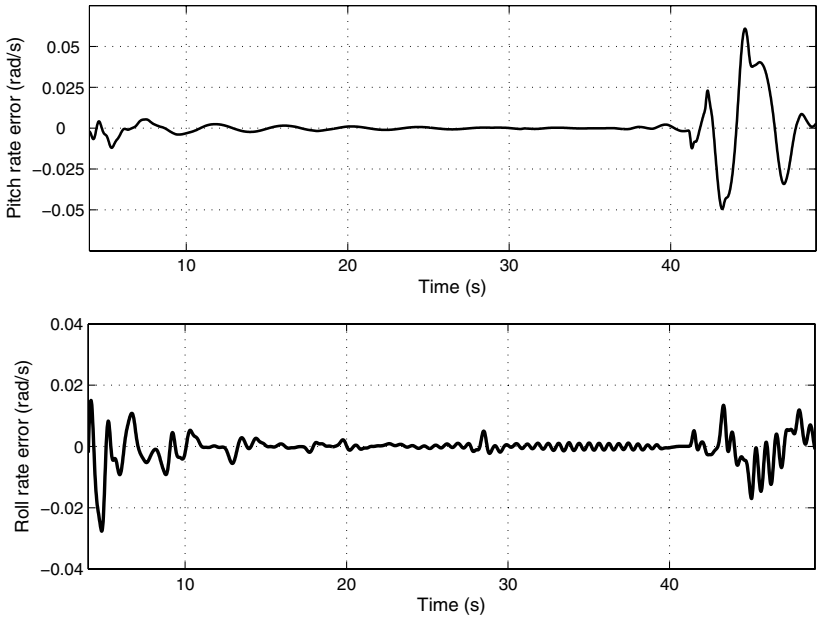

Fig. 12 Pitch rate error (upper plot) and roll rate error (lower plot) for test case 4 in Table 2. Failure occurring to the right tail effector, locked at −1.5 deg. MPCA scheme.

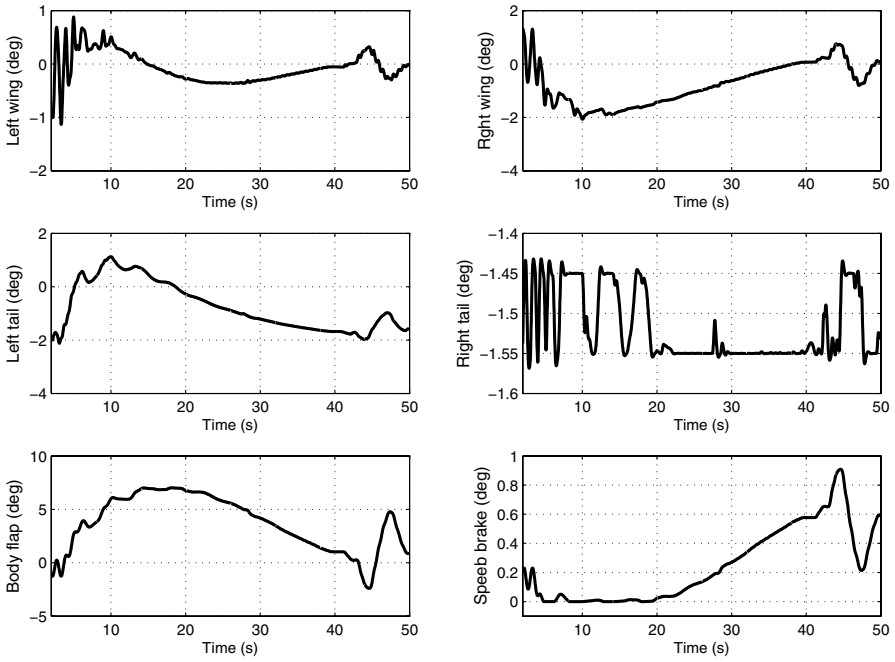

Fig. 13 Control surface deflections for case 4 in Table 2. Failure occurring to the right tail effector, locked at −1.5 deg. MPCA scheme.

account for actuator dynamics and constraints in the inner loop of a reentry vehicle guidance and control system. A time-varying affine internal model, based on a high-fidelity simulation of an experimental reentry vehicle, has been used in a model-predictive control design. The proposed scheme provides a generic approach to distribute control authority among different types of actuators. Extensive simulation studies indicate that the proposed approach shows significant performance improvement over traditional static control allocation algorithms in the presence of realistic actuator dynamics. Other advantages of the dynamic control allocation method proposed in this paper include the scheme's ability to deal with failure conditions. Such results render the proposed methodology appealing for robust reconfigurable control with multiple actuators possessing different timescale behaviors. As in all control allocation schemes, issues relative to the computational complexity of the algorithm remain to be addressed to render the proposed methodology amenable to implementation in real-time flight control systems. Whereas a thorough evaluation of the computational cost is beyond the scope of this paper and a matter of future investigation, it is argued that the suggested LP formulation of the problem may be beneficial to achieving a sufficiently fast implementation, as is the case for state-of-the-art LP-based static control allocation algorithms [37]. Moreover, recent results on explicit solvability of MPC problems that have already been applied to control allocation problems [12] may prove instrumental to achieving guaranteed convergence within a given sampling interval

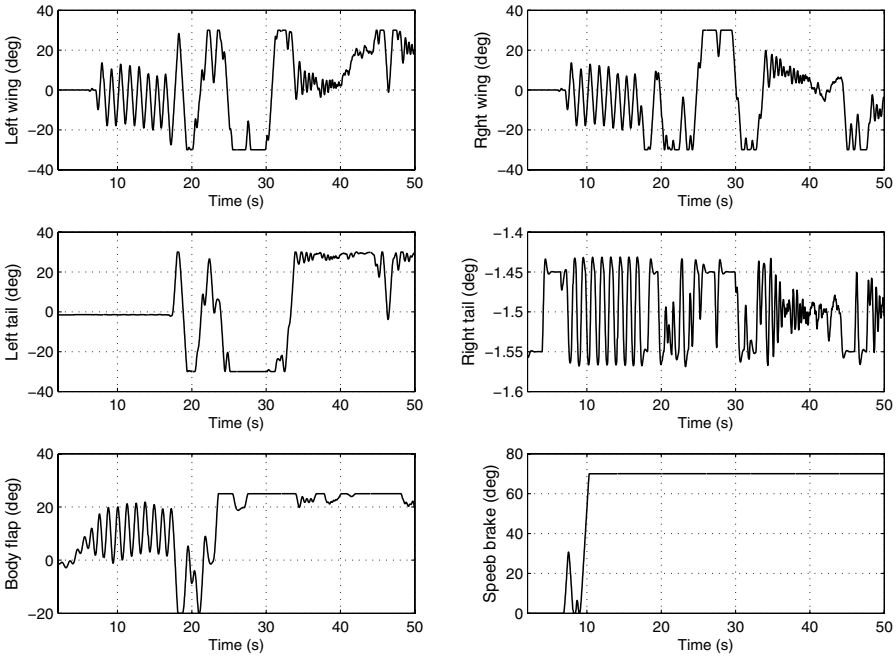

Fig. 14 Control surface deflections for case 4 in Table 2. Failure occurring to the right tail effector, locked at −1.5 deg. MOIC scheme.

of algorithms developed within the framework proposed in this paper.

Acknowledgments

This work has been supported by the U.S. Air Force Research Laboratory Air Vehicles Directorate and the U.S. Air Force Office of Scientific Research through the Collaborative Center of Control Science at The Ohio State University (Contract F33615-01-2-3154.)

References

[1] Durham, W. C., "Constrained Control Allocation," *Journal of Guidance, Control, and Dynamics*, Vol. 16, No. 4, 1993, pp. 717–725.

[2] Durham, W. C., "Attainable Moments for the Constrained Control Allocation Problem," *Journal of Guidance, Control, and Dynamics*, Vol. 17, No. 6, 1994, pp. 1371–1373.

[3] Durham, W. C., "Constrained Control Allocation: Three-Moment Problem," *Journal of Guidance, Control, and Dynamics*, Vol. 17, No. 2, 1994, pp. 330–336.

[4] Bordignon, K. A., and Durham, W. C., "Closed-Form Solutions to Constrained Control Allocation Problem," *Journal of Guidance, Control, and Dynamics*, Vol. 18, No. 5, 1995, pp. 1000–1007.

[5] Buffington, J. M., and Enns, D. F., "Lyapunov Stability Analysis of Daisy Chain Control Allocation," *Journal of Guidance, Control, and Dynamics*, Vol. 19, No. 6, 1996, pp. 1226–1230.

[6] Durham, W. C., Bolling, J. G., and Bordignon, K. A., "Minimum Drag Control Allocation," *Journal of Guidance, Control, and Dynamics*, Vol. 20, No. 1, 1997, pp. 190–193.

[7] Durham,W. C., and Bordignon, K. A., "Multiple Control Effector Rate Limiting," *Journal of Guidance, Control, and Dynamics*, Vol. 19, No. 1, 1996, pp. 30–37.

[8] Durham, W. C., "Efficient, Near-Optimal Control Allocation," *Journal of Guidance, Control, and Dynamics*, Vol. 22, No. 2, 1999, pp. 369–372.

[9] Durham, W. C., "Computationally Efficient Control Allocation," *Journal of Guidance, Control, and Dynamics*, Vol. 24, No. 3, 2001, pp. 519–524.

[10] Burken, J. J., Lu, P., Wu, Z., and Bahm, C., "Two Reconfigurable Flight Control Design Methods: Robust Servomechanism and Control Allocation," *Journal of Guidance, Control, and Dynamics*, Vol. 24, No. 3, 2001, pp. 482–493.

[11] Johansen, T., Fossen, T., and Berge, S., "Constrained Control Allocation Using Quadratic Programming," *IEEE Transactions on Control Systems Technology*, Vol. 12, No. 1, 2004, pp. 211–216.

[12] Johansen, T., Fossen, T., and Tøndel, P., "Efficient Optimal Constrained Control Allocation via Multiparametric Programming," *Journal of Guidance, Control, and Dynamics*, Vol. 28, No. 3, 2005, pp. 506–515.

[13] Bodson, M., "Evaluation of Optimization Methods for Control Allocation," *Journal of Guidance, Control, and Dynamics*, Vol. 25, No. 4, 2002, pp. 703–711.

[14] Page, A., and Steinberg, M., "A Closed-Loop Comparison of Control Allocation Methods," AIAA Paper 2000-4538, Aug. 2000.

[15] Bolender, M., and Doman, D., "Nonlinear Control Allocation Using Piecewise Linear Functions," *Journal of Guidance, Control, and Dynamics*, Vol. 27, No. 6, 2004, pp. 1017–1027.

[16] Doman, D. B., and Sparks, A. G., "Concepts for Constrained Control Allocation of Mixed Quadratic and Linear Effectors," *Proceedings of the American Control Conference*, Vol. 5, American Automatic Control Council, Evanston, IL, May 2002, pp. 3729–3734.

[17] Doman, D., and Oppenheimer, M., "Improving Control Allocation Accuracy for Nonlinear Aircraft Dynamics," AIAA Paper 2002-4667, 2002.

[18] Poonamallee, V., Yurkovich, S., Serrani, A., Doman, D., and Oppenheimer, M., "A Nonlinear Programming Approach for Control Allocation," *Proceedings of the American Control Conference*, American Automatic Control Council, Evanston, IL, June 2004, pp. 1689–1694.

[19] Oppenheimer, M., and Doman, D., "Methods for Compensating for Control Allocator and Actuator Interactions," *Journal of Guidance, Control, and Dynamics*, Vol. 27, No. 5, 2004, pp. 922–927.

[20] Härkegård, O., "Dynamic Control Allocation Using Constrained Quadratic Programming," *Journal of Guidance, Control, and Dynamics*, Vol. 27, No. 6, 2004, pp. 1028–1034.

[21] Hodel, A., and Callahan, R., "Autonomous Reconfigurable Control Allocation for Reusable Launch Vehicles," AIAA Paper 2002-4780, 2002.

[22] Simmons, A., and Hodel, A., "Control Allocation for the X-33 using Existing and Novel Quadratic Programming Techniques," *Proceedings of the 2004 American Control Conference*, American Automatic Control Council, Evanston, IL, June 2004, pp. 1701–1706.

[23] Luo, Y., Serrani, A., Yurkovich, S., Doman, D., and Oppenheimer, M., "Model Predictive Dynamic Control Allocation with Actuator Dynamics," *Proceedings of the American Control Conference*, American Automatic Control Council, Evanston, IL, June 2004, pp. 1695–1700.

[24] Luo, Y., Serrani, A., Yurkovich, S., Doman, D., and Oppenheimer, M., "Dynamic Control Allocation with Asymptotic Tracking of Time-Varying Control Input Commands," *Proceedings of the 2005 American Control Conference*, American Automatic Control Council, Evanston, IL, June 2004, pp. 2098–2103.

[25] Camacho, E., and Bordons, C., *Model Predictive Control*, Springer–Verlag, New York, NY, 1999.
[26] Doman, D., and Ngo, A., "Dynamic Inversion-Based Adaptive/Reconfigurable Control of the X-33 on Ascent," *Journal of Guidance, Control, and Dynamics*, Vol. 25, No. 2, 2002, pp. 275–284.
[27] Schierman, J. D., Ward, D. G., Hull, J. R., Gandhi, N., Oppenheimer, M. W., and Doman, D., "Integrated Adaptive Guidance and Control for Reentry Vehicles with Flight-Test Results," *Journal of Guidance, Control, and Dynamics*, Vol. 27, No. 6, 2004, pp. 975–988.
[28] Buffington, J., "Tailless Aircraft Control Allocation," AIAA Paper 1997-3605, 1997.
[29] Muske, K. R., "Steady-State Target Optimization in Linear Model Predictive Control," *Proceedings of the American Control Conference*, Vol. 6, American Automatic Control Council, Evanston, IL, June 1997, pp. 3597–3601.
[30] Rao, C., and Rawlings, J., "Steady States and Constraints in Model Predictive Control," *AIChE Journal*, Vol. 45, 1999, pp. 1266–1278.
[31] Rawlings, J. B., and Muske, K. R., "The Stability of Constrained Receding Horizon Control," *IEEE Transactions on Automatic Control*, Vol. 38, No. 10, 1993, pp. 1512–1516.
[32] Zheng, A., and Morari, M., "Stability of Model Predictive Control with Mixed Constraints," *IEEE Transactions on Automatic Control*, Vol. 40, No. 10, 1995, pp. 1818–1823.
[33] Mayne, D., Rawlings, J., Rao, C., and Scokaert, P., "Constrained Model Predictive Control: Stability and Optimality," *Automatica*, Vol. 36, No. 6, 2000, pp. 789–814.
[34] Lemke, C., "Some Pivot Schemes for the Linear Complementarity Problem," *Mathematical Programming Studies*, No. 7, 1978, pp. 15–35.
[35] Camacho, E., "Constrained Generalized Predictive Control," *IEEE Transactions on Automatic Control*, Vol. 38, No. 2, 1993, pp. 327–332.
[36] Nocedal, J., and Wright, S., *Numerical Optimization*, Springer–Verlag, New York, NY, 1999.
[37] Bolender, M., and Doman, D., "Nonlinear Control Allocation Using Piecewise Linear Functions: A Linear Programming Approach," *Journal of Guidance, Control, and Dynamics*, Vol. 28, No. 3, 2005, pp. 558–562.