A State and Input Constrained Control Method for Air-Breathing Hypersonic Vehicles

Haoyu Du, Jie Yan, Yonghua Fan

Northwestern Polytecnical University
West Youyi Road 127, 710072, Xi'an city, Shaanxi, P.R.China
duhaoyu@mail.nwpu.edu.cn, jyan@nwpu.edu.cn, fanyonghua@nwpu.edu.cn

Abstract: Besides nonlinearity, high coupling and parameter uncertainties, the design of a hypersonic flight control system still faces challenges due to the unstable dynamics under various flight conditions and to the presence of state constraints required by a scramjet. This paper presents a state and input constrained control method for the longitudinal motion of an air-breathing hypersonic vehicle through combining tensor product (TP) model transformation and the command governor approach. This method consists of three steps. Firstly, the paper applies the tensor product (TP) model transformation, making the state space matrices depend on the vector θ of time varying parameters. Secondly, it uses LQ (Linear Quadratic) method to design a set of controllers in the vertex of the TP model, and then, the controllers are checked with the parallel distributed compensation (PDC) controller design framework to ensure global stability and improving control performance. Thirdly, it introduces a command governor (CG) device for command optimization, which modifies the command signal to avoid state and input violations. The significance of this method mainly lies in its capability to avoid excessive flight constraints under various flight conditions. In order to demonstrate the effectiveness of this method, we carried out numerical simulations of the air-breathing hypersonic vehicle in its climbing phase which has state constraints and actuator constraints.

Keywords: tensor product (TP) model; linear matrix inequalities (LMI); constrained control; parallel distributed compensation (PDC); command governor (CG)

1 Introduction

Air-breathing hypersonic vehicles (AHVs) receive great research attention because they present a more promising and economical technology for access to near space for both civilian and military applications [1-2], as witnessed by the success of NASA's scramjet-powered X-43A and X-51A. Compared with the traditional flight vehicles, the AHV flight control design still faces open challenges because the vehicle has high nonlinearity, inherently unstable and complex couplings and parametric uncertainties [3-7]. Most of the research

focuses on the cruising phase. When talking about maneuvering flight, another two aspects should be taken into consideration, the changing characteristics of the vehicle with various flight conditions and constrained flight status restricted by the requirements of the scramjet for safe running [8-12]. Therefore, pursuing an accurate varying model and designing a suitable constrained control method are key problems to be solved in the maneuvering phase.

In the past few decades, the linear parameter varying (LPV) control underwent huge development. Through obtaining parameter-dependent controller in the framework of linear matrix inequalities (LMI), the LPV control system is capable of gain-scheduling in real time with the measured or estimated parameters and ensuring the robust stability and satisfied performance of the control system. Among the various LPV modeling methods, the tensor product (TP) model transformation is very attractive, as it is convenient to be combined with the LMI-based control theory and has high accuracy [13-15]. These advantages have been taken to deal with extreme parameter variations caused by large flight envelope in flight control [9]. Together with the parallel distributed compensation (PDC) controller design, TP-based LMI controllers gain global stability in the area of parameter variation [16-17]. For air-breathing hypersonic vehicle control, flight status such as angle of attack and pitch rate must meet the restrictions by the scramjet. Otherwise, engine stall may fail the flight task. Focusing on state constrained flight control, researchers make great efforts [18]. The command governor approach gains a lot of attention in this field. The specific merit of this approach lies in handling constraints on both input and state-related variables without too many numerical burdens [19-20].

Motivated by the above, the paper addresses the command governor control method that incorporates the benefits of TP-based LPV control. To begin with, this paper transforms the state-space model of the plant by transforming a tensor product (TP) model into convex state-space TP model. Then, the control system is divided into two layers. The outer layer creates overload command for altitude tracking (which is not the focus of this paper). The emphasis lies in the design of inner layer design, which contains a command governor device and a set of LQ controllers. The aim of the inner loop is to improve the flight attitude control performance and implement the appropriate overload command without violating various restrictions. The values of the LQ controllers are checked by solving a set of LMIs generated by PDC technique according to desired specifications in all the TP-model vertex systems. As for the restrictions, the command governor device optimizes the overload command in every control period, which guarantees that all the related variables may not violate constraints. Finally, nonlinear numerical simulation results are provided to verify the effectiveness of the method.

2   Flight Model

With the nonlinear model for longitudinal dynamics of an AHV considered, the nonlinear equations of motion for velocity $V(meter/s)$, altitude $H(meter)$, flight-path angle $\gamma(rad)$, the angle of attack $\alpha(rad)$ and pitch rate $q(rad/s)$ are described as follows:

$$\begin{cases} \dot{h} = V\sin\gamma \\ \dot{\gamma} = \dfrac{L+T\sin\alpha}{mV} - \dfrac{g}{V}\cos\gamma \\ \dot{\alpha} = q - \dot{\gamma} \\ \dot{q} = \dfrac{M_z}{I_{zz}} \end{cases} \tag{1}$$

where m denotes the mass of the flight vehicle; T and L represent thrust and lift force; I_{zz} stands for the longitudinal moment of inertia; M_z is the pitch moment. The control input $u = [\delta_e]$, whose unit is rad.

The CFD results show the thrust coefficient C_T, the lift coefficient C_L, and pitch moment coefficient C_M are related to Mach (Ma) number, angle of attack (α) and elevator (δ_e). The forces and moments have the following forms:

$$L = 0.5\rho V^2 S C_L(Ma,\alpha,\delta_e)$$
$$M = 0.5\rho V^2 S c C_M(Ma,\alpha,\delta_e) \tag{2}$$
$$T = 0.5\rho V^2 S C_T(Ma,\alpha,\delta_T)$$

where ρ, S, c, δ_T stand for the air density, reference area, aerodynamic chord. The unit for L, T is Newton; the unit for M is Newton times meter.

The miscellaneous coefficients of the inner-loop control system, which involves the last two equations in Eq. (1), are simplified under the trimmed condition, while the state and input and output vectors are chosen as follows:

$x_s = [\alpha, q]^T, u = [\delta_e], y = [n_z]$, where n_z is the measured value of normal acceleration.

Small perturbation linearized equations of the flight attitude are derived as follows:

$$\begin{bmatrix} \dot{\alpha} \\ \dot{q} \end{bmatrix} = \begin{bmatrix} -\dfrac{0.5\rho V^2 S C_L^\alpha + T}{mV} & 1 \\ \dfrac{0.5\rho V^2 S c C_M^\alpha}{I_{ZZ}} & 0 \end{bmatrix} \begin{bmatrix} \alpha \\ q \end{bmatrix} + \begin{bmatrix} -\dfrac{0.5\rho V^2 S C_L^{\delta_e}}{mV} \\ \dfrac{0.5\rho V^2 S c C_M^{\delta e}}{I_{zz}} \end{bmatrix} \delta_e \tag{3}$$

$$n_z = [\frac{0.5\rho V^2 S C_L^\alpha}{mg} \quad 0]\begin{bmatrix} a \\ q \end{bmatrix} + [\frac{0.5\rho V^2 S C_L^{\delta_e}}{mg}]\delta_e .$$
(4)

$C_L^\alpha, C_L^{\delta_e}, C_M^\alpha, C_M^{\delta e}$ represent first order partial derivative of coefficients in Eq. (2). $C_L^\alpha, C_L^{\delta_e}, C_M^\alpha, C_M^{\delta e}$ vary with Ma and α ; ρ changes with h , and V equals Ma times speed of sound. The equations (3) and (4) vary with $p(t) = [h, Mach, \alpha]$. We assume that all these parameters can be obtained in real-time (In fact, the Ma and h can be measured by sensors, while α should be estimated according to the inertial guidance system).

Eqs. (3) and (4) can be expressed in the LPV form:

$$\begin{cases} \dot{x}_s(t) = A_s(p(t))x_s(t) + B_s(p(t))u(t) \\ y(t) = C_s(p(t))x_s(t) + D_s(p(t))u(t) \end{cases}$$
(5)

$p(t) = [h(t), Ma(t), \alpha(t)]^T$ is the vector of varying parameters. The desired command values of the normal acceleration is denoted by $r(t) = [n_{zc}]$. The error integral equation is defined as follows:

$$e(t) = \int_0^t [r(t) - y(t)]d\tau$$
(6)

The augmented system is as follows:

$$\begin{bmatrix} \dot{x}_s(t) \\ \dot{e}(t) \end{bmatrix} = \begin{bmatrix} A_s(p(t)) & 0 \\ -C_s(p(t)) & 0 \end{bmatrix}\begin{bmatrix} x_s(t) \\ e(t) \end{bmatrix} + \begin{bmatrix} B_s(p(t)) \\ -D_s(p(t)) \end{bmatrix}u(t) + \begin{bmatrix} 0 \\ I \end{bmatrix}r(t)$$
(7)

$$y = n_z = \begin{bmatrix} C_s(p(t)) & 0 \end{bmatrix}\begin{bmatrix} x_s(t) \\ e(t) \end{bmatrix} + D_s(p(t))u(t)$$
(8)

This paper focuses on the climbing phase to do the flight maneuver. The higher altitude corresponds to a faster cruising velocity. In the climbing phase, its engine works in full mode (which means the ratio of air and oil should always be 1). The mission is to maneuver the cruising vehicle to a higher altitude as soon as possible, without violating any constraints. The longitudinal control system can be considered as a two-layered control system. The outer-loop control system determines a reference longitudinal acceleration n_{zc} based on the altitude error. The inner-loop control system regulates the vehicle's attitude to follow the reference signal using an elevator. Constraints exist in the angle of attack and pitch rate for the stable combustion of the scramjet. Constraints on the elevator prevent input saturation. These constraints can be represented by:

$\alpha_{min} \leq \alpha(t) \leq \alpha_{max}$,

$q_{min} \leq q(t) \leq q_{max}$, (9)

$\delta_{e_{min}} \leq \delta_e(t) \leq \delta_{e_{max}}$.

The attitude control should be quick and accurate with acceptable behaviours of the elevator. The paper uses the linear quadratic proportional plus integral (LQ-PI) controller to make the control performance satisfactory.

3 TP Transformation and Control Method

3.1 TP Transformation for the LPV Model

The LPV state-space model is given as follows:

$$\begin{cases} \dot{x}(t) = A(p(t))x(t) + B(p(t))u(t) \\ y(t) = C(p(t))x(t) + D(p(t))u(t) \end{cases} \quad y(t) \in \mathbb{R}^{l} \tag{10}$$

with input $u(t) \in \mathbb{R}^{k}$, output and state vector $x(t) \in \mathbb{R}^{m}$. The system matrix is written as

$$S(p(t)) := \begin{pmatrix} A(p(t)) & B(p(t)) \\ C(p(t)) & D(p(t)) \end{pmatrix} \in \mathbb{R}^{(m+k)\times(m+l)} \tag{11}$$

where $p(t) \in \Omega$ is a time-varying N-dimensional parametric vector with the hypercube $\Omega = [a_1 \quad b_1] \times [a_2 \quad b_2] \times L \times [a_N \quad b_N] \subset \mathbb{R}^{N}$.

The TP model transformation is capable of converting a model given by a set of continuous multivariable functions into a polytopic TP model. If a model is established with the relevant aerodynamic data, it still works by substituting the sampling data produced by functions with these data. In this way, the establised model can be transformed into products consisting of orthonormal one-variable weighting functions. Based on the concepts of higher-order singular value decomposition (HOSVD), these products are constructed into a tensor product structure according to the significance of each component. For more details, see Ref. [14].

In the qLPV plant, the system matrix $S(p(t))$ is transformed into the following polytopic TP model for parametric vector $p(t)$:

$$S(p(t)) = \sum_{i_1=1}^{I_1} L \sum_{i_N=1}^{I_N} \prod_{n=1}^{N} w_{n,i_n}(p_n(t)) S_{i_1 L i_N} \tag{12}$$

which stands for a parameter-dependent convex combination of linear time-invariant (LTI) system matrices, with the vertex systems $S_{i_1 L i_N} \in \mathbb{R}^{(m+k)\times(m+l)}$ and the weighting function $w_n(p_n(t))$. In Eq.(12), $p_n(t)$ stands for the n th parameter

in $p(t)$, with $n = 1, \text{L} \ N$. I_n represents the number of singular values kept in each $p_n(t)$. The number of vertices is $R = I_1 I_2 \text{L} \ I_N$. After this, the system $S(p(t))$ in Eq. (11) can be expressed by combining the R vertex in Eq. (12).

Based on the tensor product model transformation in a compact form, Eq. (12) can be rewritten as:

$$S(p(t)) = \mathbf{S} \overset{N}{\underset{n=1}{\otimes}} \text{w}_n(p_n(t)) \tag{13}$$

which enables the application of matrix and tensor algebra methods. The $(N+2)$ dimensional coefficient of the core tensor $\mathbf{S} \in \text{¡}^{\ I_1 \times \text{L} \times I_n \times (m+k) \times (m+l)}$ is constructed from the LTI vertex systems S_{i_1, L, i_N} and the row vector $\text{w}_n(p_n(t))$ that contains the one-variable weighting function $w_{n,i_n}(p_n(t))$, $i_n = 1\text{L} \ I_N$.

The weighting function satisfies the sum normalized, non-negative (SNNN) condition[14], which can be expressed as Eqs. (14) and (15):

$$\forall n, i_n, p_n(t) : w_{n,i_n}(p_n(t)) \in [0,1] \tag{14}$$

$$\forall n, i_n, p_n(t) : \sum_{i=1}^{I_n} w_{n,i_n}(p_n(t)) = 1. \tag{15}$$

In the following, we show that types of weighting functions can be modified by the TP model transformation:

Type 1: (the normality [NO] type). The resulting weighting functions satisfies the SNNN conditions, and the largest value of each function is 1.

Type 2: (the close-to-normality [CNO] type). The resulting weighting function satisfies the SNNN conditions, and the largest value of each function is 1 or close to 1.

Type 3: (the inverse normality [INO] type). The resulting weighting function satisfies the SNNN conditions, and the smallest value of all functions is 0.

For a CNO type TP model, some of its weighting functions can acquire 1, leading to $S(p(t))$ being exactly mapped into some of the vertex systems. Compared to other types, the CNO type, which yields a tighter convex hull, is generally best suitable for controller design [14, 21]. We adopt the TP model that uses the CNO type weighting function.

The systems in Eqs. (7-8) then can be rewritten with Eq. (10) to form the following:

$$x = [x_s \quad e]^T,$$

$$A(p(t)) = \begin{bmatrix} A_s(p(t)) & 0 \\ -C_s(p(t)) & 0 \end{bmatrix}, B(p(t)) = \begin{bmatrix} B_s(p(t)) \\ -D_s(p(t)) \end{bmatrix}, \tag{16}$$

$$C(p(t)) = [C_s(p(t)) \quad 0], D(p(t)) = D_s(p(t))$$

The systems' core tensor $\mathbf{S}$ has the following structure:

$$\mathbf{S} = \begin{bmatrix} \mathbf{S}_A & \mathbf{S}_B \\ \mathbf{S}_C & \mathbf{S}_D \end{bmatrix} \tag{17}$$

$$\mathbf{S}_A = \begin{bmatrix} \mathbf{S}_{A_s} & \mathbf{0} \\ -\mathbf{S}_{C_s} & \mathbf{0} \end{bmatrix}, \mathbf{S}_B = \begin{bmatrix} \mathbf{S}_{B_s} \\ -\mathbf{S}_{D_s} \end{bmatrix}, \mathbf{S}_C = \begin{bmatrix} \mathbf{S}_{C_s} & \mathbf{0} \end{bmatrix}, \mathbf{S}_D = \begin{bmatrix} \mathbf{S}_{D_s} \end{bmatrix} \tag{18}$$

With this, we obtain the TP model-based LPV model in Eq.(7) as follows:

$$\begin{bmatrix} \dot{x}_s(t) \\ \dot{e}(t) \end{bmatrix} = \begin{bmatrix} \mathbf{S}_{A_s} \overset{N}{\underset{n=1}{\otimes}} \mathrm{w}_n(p_n(t)) & 0 \\ -\mathbf{S}_{C_s} \overset{N}{\underset{n=1}{\otimes}} \mathrm{w}_n(p_n(t)) & 0 \end{bmatrix} \begin{bmatrix} x_s(t) \\ e(t) \end{bmatrix} + \begin{bmatrix} \mathbf{S}_{B_s} \overset{N}{\underset{n=1}{\otimes}} \mathrm{w}_n(p_n(t)) \\ -\mathbf{S}_{D_s} \overset{N}{\underset{n=1}{\otimes}} \mathrm{w}_n(p_n(t)) \end{bmatrix} u(t) + \begin{bmatrix} 0 \\ I \end{bmatrix} r(t) \tag{19}$$

$$y = n_z = \begin{bmatrix} \mathbf{S}_{C_s} \overset{N}{\underset{n=1}{\otimes}} \mathrm{w}_n(p_n(t)) & 0 \end{bmatrix} \begin{bmatrix} x_s(t) \\ e(t) \end{bmatrix} + \begin{bmatrix} \mathbf{S}_{D_s} \overset{N}{\underset{n=1}{\otimes}} \mathrm{w}_n(p_n(t)) \end{bmatrix} u(t) \tag{20}$$

3.2   Designing LQ Controller in the PDC Framework

The LQ method is widely used in flight control because of its good performance without excessive input. However, it is difficult to ensure the stabilization of a large flight envelope even with a large number of controllers. This part of the paper introduces a new method for combining the LQ controller design with the PDC framework to ensure that the global and asymptotic stabiliy is within the whole flight envelope.

The PDC design technique determines one feedback to each vertex system
$$\mathbf{K} = PDC(S, stability\_theorem) \tag{21}$$

Then define the control value as

$$u(t) = -\left(\mathbf{K} \overset{N}{\underset{n=1}{\otimes}} \mathrm{w}_n(p_n(t)) \right) x(t) \tag{22}$$

The control performance should be guaranteed by the selected "stability_theorem" in Eq. (21), which is a symbolic parameter specifying the stability criteria and the desired control performance predefined in terms of LMIs [23-24]. For example, the H_2 / H_∞ control performance and the pole placement can be considered through LMI-based stability theorems properly. We use the parallel distributed compensation method to guarantee the global asymptotic stability for a given dynamic system.

In order to have a direct link to the typical form of LMI-based stablity theorem, we define its indexing as follows:

$$S_r = \begin{pmatrix} A_r & B_r \\ C_r & D_r \end{pmatrix} = S_{i_1, \mathrm{L}, i_N} \tag{23}$$

where $r = ordering(i_1, \mathrm{L}, i_N)$ (24)

The term 'ordering' presents in the linear index equivalent to an N-dimensional array's index i_1, L, i_N, when the size of the array is $I_1 \times I_2 \times \mathrm{L} \times I_N$, its expression is:

$$\begin{aligned}
r &= (i_{n+1} - 1)I_{n+2}I_{n+3}\mathrm{L}\ I_N I_1 I_2 \mathrm{L}\ I_{n-1} + (i_{n+2} - 1)I_{n+3}I_{n+4}\mathrm{L}\ I_N I_1 I_2 \mathrm{L}\ I_{n-1} \\
&+ \mathrm{L}\ + (i_N - 1)I_1 I_2 \mathrm{L}\ I_{n-1} + (i_1 - 1)I_2 I_3 \mathrm{L}\ I_{n-1} + (i_2 - 1)I_3 I_4 \mathrm{L}\ I_{n-1} \\
&+ \mathrm{L}\ + (i_{n-2} - 1)I_{n-1} + i_{n-1}.
\end{aligned} \tag{25}$$

The basis functions can be defined according to the sequence of:

$$w_r(p(t)) = \prod_{n=1}^{N} w_{n,i_n}(p_n(t)) \tag{26}$$

Then the controller can be design with the Lyapunov stability theorems for global and asymptotic stability as follows:

Find $X > 0$ and M_r satisfies:

$$-XA_r^T - A_r X + M_r^T B_r^T + B_r M_r > 0 \tag{27}$$

for all r and

$$-XA_r^T - A_r X - XA_s^T - A_s X + M_s^T B_r^T + B_r M_s + M_r^T B_s^T + B_s M_r > 0 \tag{28}$$

for $r < s \leq R$, except the pair (r, s) in which $w_r(p(t))w_s(p(t)) = 0, \forall p(t)$.

Finding a positive definite matrix X and M_r or determining that no such matrices exist is a convex feasibility problem. Using the powerful tools in MATLAB-LMI toolbox [25], this problem can be solved very efficiently. The feedback gains can form the solutions X and M_r as follows:

$$K_r = M_r X^{-1} \tag{29}$$

Then, we can match the feedback K_{i_1, L, i_N} from K_r with the $r = ordering(i_1, \mathrm{L}, i_N)$ and store these gains into tensor $\mathbf{K}$ of Eq. (21).

To solve the above problem, we design the controller with the LQ method and check the control system with the PDC framework to enssure the global and asymptotic stability. The performance index becomes:

$$J = \int_0^{\infty} (x^T Q x + u^T E u) dt \tag{30}$$

Q and E are symmetric positive definite matrices. By solving the LQ controller

for the vertex system in Eq. (16) with the algebra riccati equation, we obtain a set of controllers of K_r for the vertex system. Next, using Eq. (29), M_r can be expressed as follows:

$$M_r = K_r X \tag{31}$$

We substitute Eq. (31) into Eq. (27) and (28) and solve the new LMI problem; if there exists a matrix $X > 0$, we conclude that the control system satisfies global and asymptotic stability in the whole flight envelope. Then, the closed loop control system in Eqs. (19) and (20) can be implemented:

$$
\begin{bmatrix} \dot{x}_s(t) \\ \dot{e}(t) \end{bmatrix}
= \left\{ \begin{bmatrix} \mathbf{S}_{A_s} \overset{N}{\underset{n=1}{\otimes}} \mathrm{w}_n(p_n(t)) & 0 \\ -\mathbf{S}_{C_s} \overset{N}{\underset{n=1}{\otimes}} \mathrm{w}_n(p_n(t)) & 0 \end{bmatrix} + \begin{bmatrix} \mathbf{S}_{B_s} \overset{N}{\underset{n=1}{\otimes}} \mathrm{w}_n(p_n(t)) \\ -\mathbf{S}_{D_s} \overset{N}{\underset{n=1}{\otimes}} \mathrm{w}_n(p_n(t)) \end{bmatrix} \left(\mathbf{K} \overset{N}{\underset{n=1}{\otimes}} \mathrm{w}_n(p_n(t)) \right) \right\} \begin{bmatrix} x_s(t) \\ e(t) \end{bmatrix}
$$
$$
+ \begin{bmatrix} 0 \\ I \end{bmatrix} r(t) \tag{32}
$$

$$
y = n_z = \left\{ \begin{bmatrix} \mathbf{S}_{C_s} \overset{N}{\underset{n=1}{\otimes}} \mathrm{w}_n(p_n(t)) & 0 \end{bmatrix} + \begin{bmatrix} \mathbf{S}_{D_s} \overset{N}{\underset{n=1}{\otimes}} \mathrm{w}_n(p_n(t)) \end{bmatrix} \left(\mathbf{K} \overset{N}{\underset{n=1}{\otimes}} \mathrm{w}_n(p_n(t)) \right) \right\} \begin{bmatrix} x_s(t) \\ e(t) \end{bmatrix} \tag{33}
$$

As to the command signal of $r(t) = [n_{zc}]$, the closed-loop control system in Eq.(32) is obviously a Type-1 servo system. The static error is zero. In every sampling time, we derive the time-varying parameter vector $p(t)$ according to the flight condition. The closed-loop system can be determined with the value of $p(t)$.

3.3   Command Governor Design

The model derived from Eqs. (32) and (33) is in continuous form. The discretized model with the sampling time Ts can be expressed with the following equations:
$$x(t+1) = \Phi(p(\mathrm{t}))x(t) + G(p(\mathrm{t}))r(t)$$
$$y(t) = H_y(p(\mathrm{t}))x(t) \tag{34}$$

The matrix $\Phi(p(\mathrm{t})), G(p(\mathrm{t}))$ are the discretized relevant parts in Eq.(32), while the matrix $H_y(p(\mathrm{t}))$ is the discretized relevant part in Eq. (33), with $x = [x_s \quad e]^T$.

With the state and input constrained control problem considered, we choose the command governor method to modify the command signal $r(t)$ and substitute it with the closest value $g(t)$, thus meeting all the constraints in a long enough horizon. The system in Eq.(34) is modified as follows:

$$x(t+1) = \Phi(p(t))x(t) + G(p(t))g(t) + d(t)$$
$$y(t) = H_y(p(t))x(t) \tag{35}$$
$$c(t) = H_c(p(t))x(t)$$

where $t \in \phi_+ := \{0,1,L\}$. $g(t)$ is the command governor (CG) action. A suitable command input, which, if no constraints were present, would coincide with the command reference $r(t)$. $d(t) \in ¡^n$, discribes disturbance and $d(t) \in D \subset ¡^d$; $y(t) \in ¡^m$ describes the output, which is required to track the reference signal $g(t)$; $c(t) \in ¡^c$ is the constrained vector. $H_c(p(t))$ represents the relationship between $c(t)$ and state vector $x(t)$. $c(t) \in C, \forall t \in \phi_+$ with C represents a specified convex and compact set. In the sequel, the following assumptions are made:

A1. Φ is a Schur matrix.

A2. The system in Eq. (35) is offset free, that is, $H_y(p(t))(I_n - \Phi(p(t)))^{-1}G(p(t)) = I_m$.

As to a Type-1 servo system, the A2 assumption is always satisfied.

Under these assumptions, given a constant command $g(t) \equiv \omega, \forall t$, the disturbance-free and steady-state solution of Eq. (35) is

$$\bar{x}_\omega := (I_n - \Phi(p(t)))^{-1}G(p(t))\omega,$$
$$\bar{y}_\omega := H_y(p(t))(I_n - \Phi(p(t)))^{-1}G(p(t))\omega, \tag{36}$$
$$\bar{c}_\omega := H_c(I_n - \Phi(p(t)))^{-1}G(p(t))\omega$$

The $\bar{x}_\omega, \bar{y}_\omega, \bar{c}_\omega$ means the steady value of $x(t), y(t), c(t)$ with a constant command $g(t) \equiv \omega, \forall t$, when the disturbance $d(t)$ is zero.

Consider the following Minkowski difference recursions on the constrained set:

$$C^0 := C, \quad C^k := C^{k-1} \sim H_c\Phi(p(t))^{k-1}D, \quad C^\infty := \bigcap_{k=0}^{k=\infty} C^k \tag{37}$$

The equations show that a CG device can be designed for the system in Eq. (12) and compute the action $g(\bullet)$ instantly and each time according to the convex optimization over a prediction horizon $k_0 \in \mathbb{C}_+$:

$$g(t) = \arg \min_{\omega \in \mathrm{V}(x(t))} \|\omega - r(t)\|^2 \tag{38}$$

where

$$\mathrm{V}(x) = \left\{ \omega \in \mathrm{W}^\delta : \bar{c}(k, x, \omega) \in \mathrm{C}^k, k = 0, \dots, k_0 \right\} \tag{39}$$

with

$$\mathrm{W}^\delta = \left\{ \omega \in \mathbf{R}^m : \bar{c}_\omega \in \mathrm{C}^\delta \right\}, \quad \mathrm{C}^\delta = \mathrm{C}^\infty \sim \mathrm{B}_\delta \tag{40}$$

$$\bar{c}(k, x, \omega) = H_c \left(\Phi(p(t))^{k-1} x + \sum_{i=0}^{k-1} \Phi(p(t))^{k-i-1} G(p(t)) \omega \right) \tag{41}$$

where B_δ a ball of radius δ centered at the origin, which covers the region where all constant virtual commands whose state evolution starts from the current x satisfy all the constraints during the transients, too. For computational details about Eq. (37), refer to Refs. [26-27].

In describing the current system, a small problem is that the $\Phi(p(t))$ used in the prediction sequel is a constant matrix, which, however, should be time-varying like $\Phi(p(t+k))$. Considering the computational efficiency and difficulties in solving complicated optimization problems, using $\Phi(p(t+k))$ instead of $\Phi(p(t))$ is impractical. Instead, a practical method is to increase the sampling frequency to reduce errors at the price of a moderate computational burden. With this method, a practical command governor device in the LPV form can be established. The following numerical simulations verify the effectiveness of this method.

4   Numerical Simulation

This section presents the simulation results to demonstrate the effectiveness of the proposed method for state and input constrained control of the AHV. Suppose that the AHV cruises with the velocity of 1522 m/s and the altitude of 20000 m. The cruise altitude of the maneuvering target changes to 21000 m. This orbital transfer capability is important because it improves the penetration probability and economical flight of the AHV. The initial attitude and angular velocity conditions are chosen as $\alpha_0 = 2 \deg, \gamma_0 = 0 \deg, q_0 = 0 \deg/s$. The attitude and input constraints are given as:

$$-4 \deg \leq \alpha \leq 6 \deg$$
$$-12 \deg/s \leq q \leq 12 \deg/s \tag{42}$$
$$-20 \deg \leq \delta_e \leq 20 \deg$$

The AHV's TP model-based LPV model is established with discrete flight dynamic data; the velocity $Ma \in [5, 5.5, 6]$, the altitude $h \in [20, 21, 22, 23, 24, 25, 26] \text{km}$, and the angle of attack $\alpha \in [-6, -4, -2, 0, 2, 4, 6] \deg$.

We utilize the tensor product transformation to establish the LPV models of the systems in Eq. (7)and (8). The rank of the discretized core tensor S results in 3,3,6 in the first, second and third dimensions; therefore, the $3 \times 3 \times 6 = 54$ vertex describes the exact polytopic TP model of the qLPV state-space model. To acquire the relaxed qLPV state-space model, we disposed of small singular values, which, in each dimension, take the following singular values:

Table 1
The comparison results on paris of singular values

Dimension $h(t)$:	18.8831	2.7874	1.217e-05			
Dimension $Mach(t)$:	18.7274	3.6515	0.5401			
Dimension $\alpha(t)$:	19.0624	0.7876	0.5556	0.1576	0.1054	0.0326

The singular values kept in each dimension are 2, 3, 4, then the relaxed TP model can be described with 24 vertexes.

The LTI vertex systems in Eq. (16) is given as follows:

$$A_{1,1,1} = \begin{pmatrix} -0.0023 & 1 & 0 \\ 0.3685 & 0 & 0 \\ -0.4149 & 0 & 0 \end{pmatrix}, B_{1,1,1} = \begin{pmatrix} -0.0007 \\ -1.4529 \\ -0.1188 \end{pmatrix}$$

$$A_{1,1,2} = \begin{pmatrix} -0.0033 & 1 & 0 \\ 0.4698 & 0 & 0 \\ -0.5941 & 0 & 0 \end{pmatrix}, B_{1,1,2} = \begin{pmatrix} -0.0008 \\ -1.6998 \\ -0.1379 \end{pmatrix}$$

$$A_{1,1,3} = \begin{pmatrix} -0.0030 & 1 & 0 \\ 0.0457 & 0 & 0 \\ -0.5344 & 0 & 0 \end{pmatrix}, B_{1,1,3} = \begin{pmatrix} -0.0006 \\ -1.3882 \\ -0.1137 \end{pmatrix}$$

(43)

$$A_{1,1,4} = \begin{pmatrix} -0.0026 & 1 & 0 \\ 0.4538 & 0 & 0 \\ -0.4631 & 0 & 0 \end{pmatrix}, B_{1,1,4} = \begin{pmatrix} -0.0006 \\ -1.3955 \\ -0.1141 \end{pmatrix}$$

g

g

g

$$A_{2,3,4} = \begin{pmatrix} -0.0008 & 1 & 0 \\ -0.2121 & 0 & 0 \\ -0.1018 & 0 & 0 \end{pmatrix}, B_{2,3,4} = \begin{pmatrix} -0.0004 \\ -0.5823 \\ -0.0479 \end{pmatrix}$$
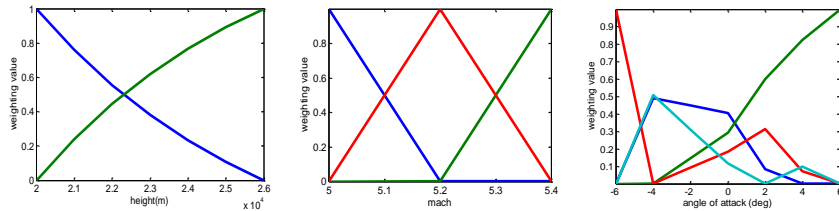

Figure 1

CNO-type scheduling parameter weighting functions for the established TP model

Then, we use the inner-loop control structure (shown in Figure 2) to realize offset –free command tracking.
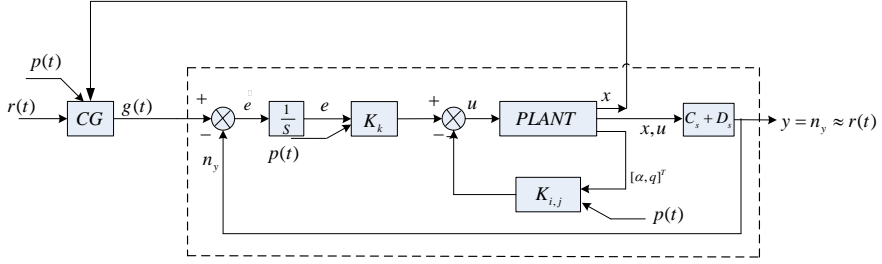
Figure 2

The Inner-loop control structure with command governor

Where the dash-line-surrounded part represents the primal controller. Using the LQ method shown in Eq. (30), we design the primal controller for vertex systems with Q and E as follows:

$$Q = \begin{bmatrix} 80 & 0 & 0 \\ 0 & 1.2 & 0 \\ 0 & 0 & 0.3 \end{bmatrix}, E = 3 \tag{44}$$

We gain the following vertex controllers:

$K(1,1,1) = [-5.3716, -0.6709, 0.3162]$

$K(1,1,2) = [-5.1731, -0.6289, 0.3162]$

$K(1,1,3) = [-5.1263, -0.6465, 0.3162]$

g   (45)

g

g

$K(2,3,4) = [-5.4605, -0.7560, 0.3162]$

Based on PDC theory, we use MATLAB LMI Toolbox to test the feasibility of Eqs. (27) and (28). The monitoring matrix is as follows:

$$X = \begin{bmatrix} 8.9962e-10 & -4.886e-09 & 5.6984e-09 \\ -4.0886e-09 & 3.9156e-08 & 5.4979e-09 \\ 5.6984e-09 & 5.4979e-08 & 1.1643e-07 \end{bmatrix} > 0 \tag{46}$$

The matrix indicates that the compensated LPV system is globally and asymptotically stable. Then the control value is computed according to Eq. (22):

$$u(t) = -(\sum_{i=1}^{2} \sum_{j=1}^{3} \sum_{k=1}^{4} w_{1,i}(h) w_{2,j}(Ma) w_{3,k}(\alpha) K_{i,j,k}) x(t) \tag{47}$$

The compensated system can be described as:

$$
\begin{bmatrix} \dot{x}_s(t) \\ \dot{e}(t) \end{bmatrix} =
$$

$$
\left\{ \begin{bmatrix} \mathbf{S}_{A_s} \sum_{i=1}^{2}\sum_{j=1}^{3}\sum_{k=1}^{4} w_{1,i}(h)w_{2,j}(Ma)w_{3,k}(\alpha)\,\mathrm{K}_{i,j,k} & 0 \\ -\mathbf{S}_{C_s} \sum_{i=1}^{2}\sum_{j=1}^{3}\sum_{k=1}^{4} w_{1,i}(h)w_{2,j}(Ma)w_{3,k}(\alpha)\,\mathrm{K}_{i,j,k} & 0 \end{bmatrix} + \right.
$$

$$
\left. \begin{bmatrix} \mathbf{S}_{B_s} \sum_{i=1}^{2}\sum_{j=1}^{3}\sum_{k=1}^{4} w_{1,i}(h)w_{2,j}(Ma)w_{3,k}(\alpha)\,\mathrm{K}_{i,j,k} \\ -\mathbf{S}_{D_s} \sum_{i=1}^{2}\sum_{j=1}^{3}\sum_{k=1}^{4} w_{1,i}(h)w_{2,j}(Ma)w_{3,k}(\alpha)\,\mathrm{K}_{i,j,k} \end{bmatrix} \left(\mathbf{K}\sum_{i=1}^{2}\sum_{j=1}^{3}\sum_{k=1}^{4} w_{1,i}(h)w_{2,j}(Ma)w_{3,k}(\alpha)\,\mathrm{K}_{i,j,k} \right) \right\} \begin{bmatrix} x_s(t) \\ e(t) \end{bmatrix}
$$

$$
+ \begin{bmatrix} 0 \\ I \end{bmatrix} g(t)
$$

$$
\tag{48}
$$

$$
y =
$$

$$
\left\{ \begin{bmatrix} \mathbf{S}_{C_s} \sum_{i=1}^{2}\sum_{j=1}^{3}\sum_{k=1}^{4} w_{1,i}(h)w_{2,j}(Ma)w_{3,k}(\alpha)\,\mathrm{K}_{i,j,k} & 0 \end{bmatrix} + \right.
$$

$$
\left. \begin{bmatrix} \mathbf{S}_{D_s} \sum_{i=1}^{2}\sum_{j=1}^{3}\sum_{k=1}^{4} w_{1,i}(h)w_{2,j}(Ma)w_{3,k}(\alpha)\,\mathrm{K}_{i,j,k} \end{bmatrix} \left(\mathbf{K}\sum_{i=1}^{2}\sum_{j=1}^{3}\sum_{k=1}^{4} w_{1,i}(h)w_{2,j}(Ma)w_{3,k}(\alpha)\,\mathrm{K}_{i,j,k} \right) \right\} \begin{bmatrix} x_s(t) \\ e(t) \end{bmatrix}
$$

$$
\tag{49}
$$

We translate the system in Eqs. (48) and (49) into its discrete form with the sampling time of 0.005s,

$$
x(t+1) = \Phi(p(\mathrm{t}))x(t) + G(p(\mathrm{t}))g(t) + d(t),
$$
$$
y(t) = H_y(p(\mathrm{t}))x(t)
$$
$$
\tag{50}
$$

We also transform $c = [\alpha \quad q \quad u]^T$ into constrained variables and describe it with the following state variables:

$$
c(t) = H_c x(t) \tag{51}
$$

$$
H_c = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ \mathrm{K}_1 \overset{N}{\underset{n=1}{\otimes}} \mathrm{w}_n(p_n(t)) & \mathrm{K}_2 \overset{N}{\underset{n=1}{\otimes}} \mathrm{w}_n(p_n(t)) & \mathrm{K}_3 \overset{N}{\underset{n=1}{\otimes}} \mathrm{w}_n(p_n(t)) \end{bmatrix} \tag{52}
$$

The $d(t)$ should be the model error, and its region is estimated to be within 0.02.

The nominal constraints are formulated into Eq.(42) and the command governor is design. Then, we design the outer-loop flight control system as the PD control of

height error. The total control system is depicted in Figure 3, with $K_h = 0.02$ and $K_d = 0.1$.
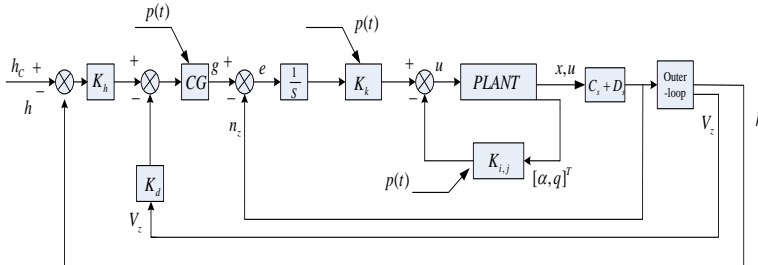

Figure 3

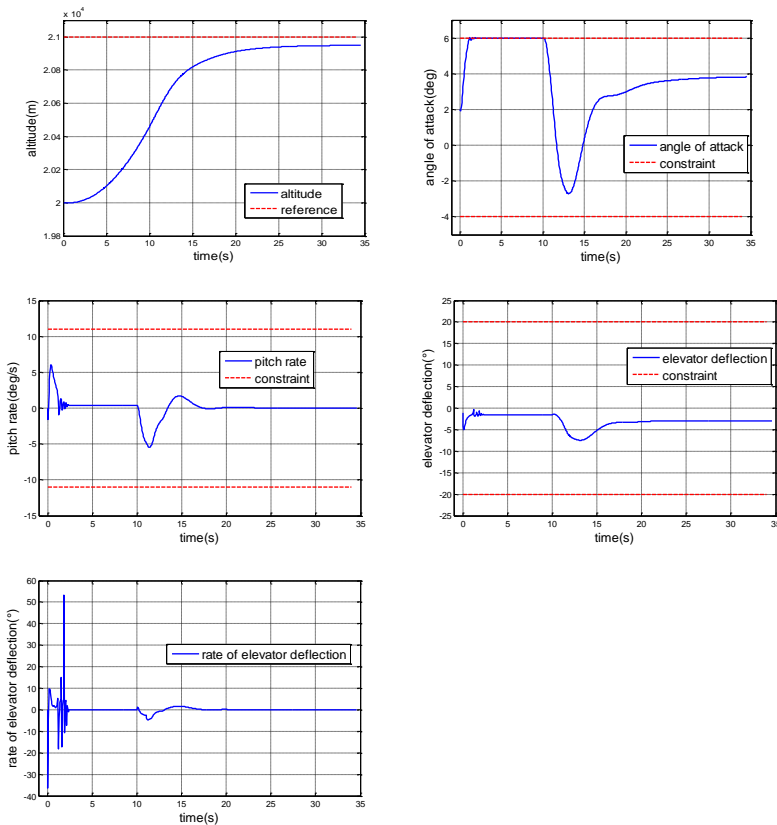The Framework that includes both the inner and outer-loop control systems

Figure 4

Simulation results

Figure 4 illustrates that the AHV climbs for 1km and shows that: the settling time is less than 20 seconds; the steady-state error is within 50 m; the angle of attack stays within -3 to 6 degrees; the pitch rate stays within -5.5 to 6.5 deg/s; the elevator deflection remains -7.5 to 0 degrees. All the constraints are satisfied. The rate of elevator deflection remains -37 to 54 deg/s, indicating that the control signal satisfies the actuator's constraints. The figure also shows that the angle of attack remains to be the maximum value available, namely for a period of almost 10 seconds when the climbing begins, which gains a maximum lift to accelerate the climbing. This is because when the climbing begins, the error of the command signal h_C and the measured altitude h is big and leads to the big normal overload command of the inner-loop control system. Then, the command governor modifies the overload command to a maximum degree that satisfies all the constraints. In this way, the best use is made of the angle of attack available to realize a quick climbing. When the error decreases and the climbing velocity increases, the AHV adjusts its angle of attack to track the desired altitude without overshoot. The final altitude is 20970 meters, indicating that the altitude tracking system. It is because the altitude tracking system has a steady error. This is because the altitude tracking system is a Type-0 servo system.

Conclusion

The purpose of this study is to demonstrate the successful application of a state and input constrained controller to designing an AHV's control system with a large flight envelope. Based on the TP transformation, the AHV's accurate qLPV model is established with the acceptable number of vertexes and therefore lightens the computational burden for model description and controller design. As to the controller design, the authors use the LQ method within the PDC framework,,thus obtaining a good control performance near all the vertexes and a global stability in the flight envelope. Together with command governor design, the AHV meets all the constraints in the climbing phase. At the beginning of the climbing phase, the flight makes the best use of maximal angle of attack available to climb quickly. The simulation results on the altitude tracking system demonstrate the effectiveness of the proposed state and input constrained control method.

Acknowledgement

References

[1]   C Mu, Z Ni, C Sun, H He: Air-Breathing Hypersonic Vehicle Tracking Control Based on Adaptive Dynamic Programming, IEEE Transactions on Neural Networks & Learning Systems, 28(3) pp. 584-598, 2016

[2]   D P Wiese, AM Annaswamy, JA Muse, MA Bolender: Adaptive Control of a Generic Hypersonic Vehicle, Aiaa Guidance, Navigation, and Control conference, Boston, MA, USA, pp. 19-22, 2013

[3]   X Yang, J Li, Y Dong: Flexible Air-Breathing Hypersonic Vehicle Control Based on a Novel Non-Singular Fast Terminal Sliding Mode Controland Nonlinear Disturbance Observer, Proceedings of the Institution of Mechanical Engineers Part G Journal of Aerospace Engineering, 231(11) pp. 2132-2145, 2016

[4]   L Fiorentini, A Serrani, MA Bolender, DB Doman: Nonlinear Robust Adaptive Control of Flexible Air-Breathing Hypersonic Vehicles, Journal of Guidance Control & Dynamics, 32(2) pp. 402-417, 2012

[5]   Z Guo, J Zhou, J Guo, J Cieslak, J Chang: Coupling Characterization-Based Robust Attitude Control Scheme For Hypersonic Vehicles, IEEE Transactions on Industrial Electronics, 64 (8) pp. 6350-6361, 2017

[6]   Y Wei, Y Chen, G Duan, W Liu: Reference Command Tracking of a Hypersonic Vehicle With Elastic Effects, Guidance, Navigation and Control Conference. China, pp. 1-6, 2016

[7]   Y Zhu, H Shen, Y Liu, G Zhang, Y Lu: Optimal Control and Analysis for Aero-Elastic Model ff Hypersonic Vehicle, Guidance, Navigation and Control Conference, China, pp. 1911-1915, 2016

[8]   X Tao, N Li, S Li: Multiple Model Predictive Control for Large Envelope Flight of Hypersonic Vehicle Systems, Information Sciences, 328(C):pp. 115-126, 2016

[9]   W Jiang, H Wang, J Lu, Z Xie: Hosvd-Based LPV Modeling and Mixed Robust H2/H∞ Control Design for Air-Breathing Hypersonic Vehicle, Journal of Systems Engineering and Electronics, 27(1), pp. 183-191, 2016

[10]   LI Gong Jun: Adaptive Tracking Control For Air-Breathing Hypersonic Vehicles With State Constraints, Frontiers of Information Technology and Electronic Engineering, 18(5):pp. 599-614, 2017

[11]   Zhu S, Li Y, Zhang J, Sun T: Characteristic Model-Based Robust Predictive Control for Reentry Hypersonic Vehicle with Constraints, Guidance, Navigation and Control Conference, China, pp. 634-639, 2016

[12]   Hao A, Xia H, Wang C: Barrier Lyapunov Function-Based Adaptive Control for Hypersonic Flight Vehicles, Nonlinear Dynamics, 88(3) pp. 1833-1853, 2017

[13]   P. Baranyi: TP-Model Transformation-Based-Control Design Frameworks, Springer International Publishing, Switzerland, pp. 258-260, 2016

[14]   P. Baranyi, Y. Yam, P. Várlaki: Tensor Product Model Transformation in Polytopic Model-based Control, Taylor&Francis, Boca Raton FL, pp. 248-250, 2013

[15] P. Baranyi: The Generalized TP Model Transformation for TS Fuzzy Model Manipulation and Generalized Stability Verification, IEEE Transaction on Fuzzy Systems, 22(4) pp. 934-948, 2014

[16] P. Baranyi: TP model transformation as a way to LMI based controller design, IEEE Transaction on Industrial Electronics, 51(2) pp. 387-400, 2004

[17] P. Baranyi, D. Tikk, Y. Yam, R. J. Patton: From Differential Equations to PDC Controller Design via Numerical Transformation, Computers in Industry, 51(3) pp. 281-297, 2003

[18] H Huang, Y Ke, W Qin: PD-RMPC for a Flexible Air-Breathing Hypersonic Vehicle with Input Saturation and State Constraints, Guidance, Navigation and Control Conference, Yantai, China, pp. 1224-1229, 2014

[19] A Casavola, E Mosca: Bank-To-Turn Missile Autopilot Design via Observer-Based Command Governor Approach, Journal of Guidance Control & Dynamics, 27(4) pp. 705-710, 2004

[20] A Casavola, E Mosca, D Angeli: Robust Command Governors for Constrained Linear Systems, Automatic Control IEEE Transactions on automatic control, 45(11) pp. 2071-2077, 2000

[21] A Szollosi, P Baranyi: Improved Control Performance of The 3-Dof Aeroelastic Wing Section: a TP Model Based 2D Parametric Control Performance Optimization, Asian Journal of Control, 19(2) pp. 450-466, 2017

[22] A. Szollosi, P. Baranyi: Influence of the Tensor Product Model Representation of QLPV Models on the Feasibility of Linear Matrix Inequality, Asian Journal of control, 18(4) pp. 1328-1342, 2016

[23] B. Takarics, P. Baranyi: Tensor Product Model Based Control of a Three Degrees-of-Freedom Aeroelastic Model, Journal of Guidance, Control and Dynamics, 36(5) pp. 1527-1533, 2013

[24] A. Szollosi, P. Baranyi: Improved control performance of the 3-DoF aeroelastic wing section: a TP model based 2D parametric control performance optimization, Asian Journal of Control, 19(2) pp. 450-466, 2017

[25] P. Gahinet, A. Nemirovski, A. J. Laub, M. Chilali: LMI Control Toolbox, Math Works, 1995

[26] Kolmanovsky I, Gilbert EG: Theory and computation of disturbance invariant sets for discrete-time linear systems, Mathematical Problems in Egineering,4(4) pp. 317-367, 1998

[27] Artstein Z, Rakovi´c SV: Set-invariance under output-feedback: a set-dynamics approach, International Journal of Systems Science 42(4) pp. 539-555, 2011