2005 American Control Conference
June 8-10, 2005. Portland, OR, USA

Reconfigurable GPC with Application to Flight Control

Jianjun Shi
Graduate student
Mechanical Engineering
Iowa State University
Ames, Iowa 50011
Email: rabbitsj@iastate.edu

Atul G. Kelkar
Associate Professor
Mechanical Engineering
Iowa State University
Ames, Iowa 50011
Email: akelkar@iastate.edu

Don Soloway
Computational Sciences Division
NASA Ames Research Center
Moffett Field, CA 94035

Abstract— **This paper presents Multi-Input Multi-Output (MIMO) Generalized Predictive Control (GPC) law and its application to reconfigurable control. A method to compute the desired end-point state from the desired output and end-point state weighting matrix is given. In particular, an application to flight control with actuator saturation failure is presented. A numerical simulation using a short-period approximation model of a civil transport aircraft is presented to demonstrate the reconfigurable control architecture.**

I. INTRODUCTION

Over last decade, GPC has emerged as one of the leading control design strategies for robust control of dynamical systems[1], [2], [3]. It places importance on the optimal performance. With the advances made in the computer technology computational speed is not a major concern for many real-life applications and control engineers have started using GPC for many main-stream applications[4]. Recently, in 90's, for Single-Input Single-Output (SISO) systems, stability of GPC under end-point state weighting was shown in [5], [6] where a penalty was imposed on the end-point state to ensure stability. This end-point state weighting idea was used in [7] for establishing stability of Multi-Input Multi-Output (MIMO) reconfigurable GPC-based control architecture. However, several synthesis issues still need further investigation. One of these issues is how to compute the desired end-point state from the desired output. The second is how to find an end-point state weighting matrix. This paper presents approaches to address these problems.

II. BASIC RESULTS FOR MIMO RECONFIGURABLE GPC-BASED CONTROL ARCHITECTURE

Consider the following MIMO Linear Time-Invariant (LTI) system,

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{A}\mathbf{x}(k) + \mathbf{B}\Delta\mathbf{u}(k) \\ \mathbf{y}(k) &= \mathbf{C}\mathbf{x}(k) \end{aligned} \quad (1)$$

where, $\mathbf{x}$ is the n-dimensional state vector, $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$ are system matrices with dimensions $n \times n$, $n \times p$, and $q \times n$, respectively.

In GPC, a set of future control signals is calculated by optimizing a suitable performance index so as to keep the plant output as close to the desired output, or reference trajectory, as possible. In this case, the cost function to be minimized is given by

$$\begin{aligned} &J(N_1, N_2, N_u, \mathbf{Q}, \lambda) = \\ &\quad (\mathbf{x}(k+N_2) - \mathbf{w}_x(k+N_2))^T \\ &\quad\quad \mathbf{Q}(\mathbf{x}(k+N_2) - \mathbf{w}_x(k+N_2)) + \\ &\sum_{j=N_1}^{N_2} [\mathbf{y}(k+j) - \mathbf{w}(k+j)]^T [\mathbf{y}(k+j) - \mathbf{w}(k+j)] + \\ &\quad \lambda \sum_{j=1}^{N_u} [\Delta\mathbf{u}(k+j-1)]^T [\Delta\mathbf{u}(k+j-1)] \end{aligned} \quad (2)$$

where, $\mathbf{w}$ is the q-vector of desired output, or reference input, and $\Delta\mathbf{u}$ is the p-vector of input increments, $\mathbf{y}$ is the q-vector of predicted output, which depends on the past inputs and outputs and on the future control signals $\Delta\mathbf{u}(k+j)$, $j = 0, 1, \ldots, N_u - 1$. N_u is the control horizon. N_1 and N_2 are lower and upper prediction horizons, respectively. $\mathbf{w}_x(k+N_2)$ is the desired value of the state at the end of the prediction horizon. $\mathbf{Q}$ is the end-point state weighting matrix.

Let $N_1 = 1$, $N_2 = N_u = N$. Define

$$\mathbf{G} = \begin{bmatrix} \mathbf{CA}^{N-1}\mathbf{B} & \mathbf{CA}^{N-2}\mathbf{B} & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{CA}^{N-1}\mathbf{B} & \mathbf{CA}^{N-2}\mathbf{B} & \cdots & \mathbf{CA}^{N-1}\mathbf{B} & \mathbf{CB} \end{bmatrix}, \quad (3)$$

$$\overline{C} = \begin{bmatrix} \mathbf{A}^{N-1}\mathbf{B} \cdots & \mathbf{B} \end{bmatrix}, \quad (4)$$

and

$$\mathbf{f} = \underbrace{\begin{bmatrix} \mathbf{CA}^1 & \cdots & \mathbf{CA}^N \end{bmatrix}^T}_{L} \mathbf{x}(k) = L\mathbf{x}(k) \quad (5)$$

The GPC control law with zero reference output is given by

$$\begin{aligned} &\Delta\mathbf{u}^* = \\ &-\left(\mathbf{G}^T\mathbf{G} + \lambda\mathbf{I} + \overline{C}^T\mathbf{Q}\overline{C}\right)^{-1}\left(\mathbf{G}^T L + \overline{C}^T\mathbf{Q}\mathbf{A}^N\right)\mathbf{x}(k) \\ &= -\mathbf{K}\mathbf{x}(k) \end{aligned} \quad (6)$$

The stability result of the end-point state weighting GPC is given by following theorem.

Theorem 1: If the system given by Eq. (1) is stabilizable and detectable, $P_0 \succeq P_1$, $\mathbf{Q} \succeq 0$, $\lambda > 0$, $N_1 = 1$, and $N_2 = N_u = N$, then the GPC control law of Eq. (6) stabilizes the system.
where $P_0 = \mathbf{Q} + \mathbf{C}^T\mathbf{C}$ and P_1 is the solution of the following Riccati Difference Equation (RDE)

$$P_{m+1} = \mathbf{A}^T P_m \mathbf{A} - \mathbf{A}^T P_m \mathbf{B}(\mathbf{B}^T P_m \mathbf{B} + \lambda \mathbf{I})^{-1}\mathbf{B}^T P_m \mathbf{A} + \mathbf{C}^T\mathbf{C}. \tag{7}$$

Note that we use $\succ$, $\succeq$ to denote positive definiteness, positive semidefiniteness. For example, $P_0 \succeq 0$ means that P_0 is positive definite and $P_0 \succeq P_1$ means that $P_0 - P_1$ is positive semidefinite.
Proof: see [7]. ■
We can extend this stability result to the case of reconfigurable control architecture wherein the system has some redundancy in control actuators and in the event of actuator failures, like saturation for example.

Definition 1: Reconfiguration matrix: Any $q \times p$ matrix with binary entries (0 or 1) is called as Reconfiguration Matrix, Q_{rc}, and is used to set the control priorities for each actuator.
Q_{rc} is used to select which actuator is used to control a certain output. The size of the Q_{rc} matrix is q (number of outputs) by p (number of inputs). Each element of Q_{rc} is either 1 or 0 indicating weather a particular actuator is allowed to control a specific output or not. For example, let us take a 5-input 2-output system. An example Q_{rc} matrix could be

$$Q_{rc} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \end{bmatrix} \tag{8}$$

In this example of Q_{rc}, the first output can be controlled only by the second input, and the second output can be controlled by both the third and fourth inputs. This matrix is then used to "reconfigure" the $\mathbf{H}$ matrix (where $\mathbf{H}$ is the transfer function matrix of system in Eq. (1)) as follows which causes the reconfiguration. Reconfigured $\mathbf{H}$ matrix is denoted by $\mathbf{H}_{rc}$ and is given by

$$\mathbf{H}_{rc} = Q_{rc} \otimes \mathbf{H} \tag{9}$$

where, $\otimes$ indicates the elementwise product or Schur-Hadamard product. This product effectively modifies the $\mathbf{B}$ and $\mathbf{C}$ matrix of the system to account for reconfiguration. Define

$$\mathcal{Q} = \left\{ Q_{rc}^i \mid (\mathbf{A}, \tilde{\mathbf{B}}_i, \tilde{\mathbf{C}}_i) \quad \text{is stabilizable and detectable} \quad \forall i \right\} \tag{10}$$

where, $\tilde{\mathbf{B}}_i$ is the reconfigured $\mathbf{B}$ matrix and $\tilde{\mathbf{C}}_i$ is the reconfigured $\mathbf{C}$ matrix as a result of outer product of Q_{rc}^i with $\mathbf{H}$. Then we have the following stability result.

Theorem 2: All systems resulting from reconfiguration of system given by Eq. (1) due to reconfiguration matrix Q_{rc} are stabilized by the control law of Eq. (6) if Q_{rc} belongs to the set $\mathcal{Q}$.

Proof: See [7]. ■

III. Computation of Desired State Trajectories

For nonzero reference trajectory, the GPC control law is given by (see [7])

$$\Delta \mathbf{u}^* = -(\mathbf{G}^T\mathbf{G} + \lambda \mathbf{I} + \overline{C}^T \mathbf{Q} \overline{C})^{-1} \mathbf{b}^T. \tag{11}$$

where $\mathbf{b}$ is given by

$$\mathbf{b} = \left[(\mathbf{A}^N \mathbf{x}(k) - \mathbf{w}_x(k+N))^T \mathbf{Q}\overline{C} + (\mathbf{f} - \mathbf{w})^T \mathbf{G} \right] \tag{12}$$

and $\mathbf{G}$, $\overline{C}$, and $\mathbf{f}$ are defined in Eq. (3), (4) and (5), respectively. We can see that the control law computation requires that the desired state trajectories are known. In Theorem 1, the reason that the end point state weighting is used instead of the output weighting is it facilitates the stability proof which is based on LQR-based arguments. This approach makes it easier to achieve the necessary condition needed for ensuring stability - $P_0 \succeq P_1$. However, in most real life situations it is safe to assume that the desired output trajectory is given to the control designer. This section gives the procedure for computing the desired state trajectories given the desired output trajectories for a general case.
Consider MIMO LTI system given by (1). Let $\mathbf{w}(k+N)$ denote the desired output trajectory that needs to be tracked by the system and let $\mathbf{w}_x(k+N)$ denote the corresponding trajectories for the desired state. Then, $\mathbf{w}_x(k+N)$ must satisfy the following conditions:

$$\mathbf{w}(k+N) = \mathbf{C}\mathbf{w}_x(k+N) \tag{13}$$

$$(\mathbf{A} - \mathbf{I})\mathbf{w}_x(k+N) = 0 \tag{14}$$

The first condition (Eq. 13) is a direct consequence of output equation. The second condition (Eq. 14) is derived next. One consideration in tracking problem is that the choice of $\mathbf{w}_x(k+N)$ should be consistent with the zero steady-state output tracking error. Consider the GPC control law with tracking error feedback:

$$\Delta \mathbf{u}^*(k) = -\mathbf{K}e(k) \tag{15}$$

where

$$e(k) = \mathbf{x}(k) - \mathbf{w}_x(k)$$

The closed-loop error dynamics for the system given by Eq. (1) with control law (15) is given by

$$\begin{aligned} e(k+1) &= \mathbf{x}(k+1) - \mathbf{w}_x(k+1) \\ &= \mathbf{A}\mathbf{x}(k) + \mathbf{B}\Delta\mathbf{u}^*(k) - \mathbf{w}_x(k+1) \\ &= (\mathbf{A} - \mathbf{B}\mathbf{K})e(k) + \mathbf{A}\mathbf{w}_x(k) - \mathbf{w}_x(k+1). \end{aligned}$$

At steady-state, for zero tracking error we need

$$e(k+1) = e(k) = 0, \quad \text{and} \quad \mathbf{w}_x(k) = \mathbf{w}_x(k+1).$$

i.e.

$$(\mathbf{A} - \mathbf{I})\mathbf{w}_x(\infty) = 0$$

To satisfy the end-point constraint of zero tracking error, we let

$$(\mathbf{A} - \mathbf{I})\mathbf{w}_x(k+N) = 0.$$

Combining the conditions of Eqs. (13) and (14), we get

$$\begin{bmatrix} \mathbf{A} - \mathbf{I} \\ \mathbf{C} \end{bmatrix} \mathbf{w}_x(k+N) = \begin{bmatrix} 0 \\ \mathbf{w}(k+N) \end{bmatrix} \tag{16}$$

Note that for the case of redundant actuators or for the case when multiple actuators can affect the same output, the matrix on the left hand side has row rank less than or equal to n as shown in [8]. Therefore, Eq. (16) is typically under-determined. In such case, The least square solution can be obtained as follows:

$$\mathbf{w}_x(k+N) = \left(\begin{bmatrix} \mathbf{A} - \mathbf{I} \\ \mathbf{C} \end{bmatrix} \right)^{\dagger} \begin{bmatrix} 0 \\ \mathbf{w}(k+N) \end{bmatrix} \tag{17}$$

where $(\cdot)^{\dagger}$ denotes pseudo-inverse of $(\cdot)$, will satisfy Eq. (16).

IV. Q-Synthesis using Constrained Nonlinear Optimization

Theorem 1 states that the system can be stabilized by picking an appropriate end-point state weighting matrix $\mathbf{Q}$. Unfortunately, it is not known how to find such a $\mathbf{Q}$, and if such a $\mathbf{Q}$ exists. In general, it is difficult to find such a matrix. However, we can see that if the system given by Eq. (1) is stabilizable, the end-point weighting matrix $\mathbf{Q}$ in Theorem 1 always exists. For example, choose

$$\mathbf{Q}(0) = \mathbf{C}^T\mathbf{C} + D_p$$

where $D_p \succ 0$. Then solve the following Algebraic Riccati Equation (ARE) for P_0.

$$P_0 = \mathbf{A}^T P_0 \mathbf{A} - \mathbf{A}^T P_0 \mathbf{B}(\mathbf{B}^T P_0 \mathbf{B} + \lambda \mathbf{I})^{-1} \mathbf{B}^T P_0 \mathbf{A} + \mathbf{Q}(0)$$

Note that the solution to the above ARE always exists since the system is stabilizable. Now set the weighting matrix

$$\mathbf{Q} = P_0 - \mathbf{C}^T \mathbf{C} \succeq 0.$$

This follows from the monotonicity property of the solution of RDE. Next, we will prove that such $\mathbf{Q}$ will make $P_0 - P_1 \succeq 0$. By the RDE (7),

$$\begin{aligned} P_0 &= \mathbf{A}^T P_0 \mathbf{A} - \mathbf{A}^T P_0 \mathbf{B}(\mathbf{B}^T P_0 \mathbf{B} + \lambda \mathbf{I})^{-1} \mathbf{B}^T P_0 \mathbf{A} + \mathbf{Q}(0) \\ &= P_1 + D_p \end{aligned}$$

Therefore,

$$P_0 - P_1 = D_p \succ 0.$$

Remark 1: Although the above procedure gives a method to find the weighting matrix, it is conservative. It is found that this procedure can yield the matrix $\mathbf{Q}$ very large which can cause numerical problems. A more systematic approach for synthesis of $\mathbf{Q}$ based on constrained nonlinear optimization is given next. For $N_1 = 1$, $N_2 = N_u = N$ case, the cost function with end-point state weighting in Eq.(2) can be written as

$$\begin{aligned} J(\mathbf{Q}, N, \lambda) &= \mathbf{x}^T(k+N)\mathbf{Q}\mathbf{x}(k+N) + \\ &\sum_{j=1}^{N} \left\{ \mathbf{y}^T(k+j)\mathbf{y}(k+j) + \right. \\ &\qquad \left. \lambda \Delta \mathbf{u}^T(k+j-1)\mathbf{u}(k+j-1) \right\} \\ &= \mathbf{x}^T(k+N)(\mathbf{Q} + \mathbf{C}^T \mathbf{C})\mathbf{x}(k+N) + \\ &\sum_{j=0}^{N-1} \left\{ \mathbf{x}^T(k+j)\mathbf{C}^T \mathbf{C}\mathbf{x}(k+j) + \right. \\ &\qquad \left. \lambda \Delta \mathbf{u}^T(k+j)\Delta \mathbf{u}(k+j) \right\} \\ &\qquad - \mathbf{x}^T(k)\mathbf{C}^T \mathbf{C}\mathbf{x}(k) \end{aligned} \tag{18}$$

where, the dependence of J on $\mathbf{Q}$ is emphasized. We can express the GPC problem as the following optimization problem:

$$\begin{aligned} \min_{\Delta \mathbf{u}} \quad & J(\mathbf{Q}, N, \lambda) \\ \text{subject to} \quad \mathbf{x}(k+1) &= \mathbf{A}\mathbf{x}(k) + \mathbf{B}\Delta \mathbf{u}(k) \quad (19) \\ \mathbf{y}(k) &= \mathbf{C}\mathbf{x}(k) \quad (20) \end{aligned}$$

Solution of this optimization yields the optimal control as

$$\Delta \mathbf{u}^*(\mathbf{Q}) = -\left(\mathbf{G}^T \mathbf{G} + \lambda \mathbf{I} + \overline{C}^T \mathbf{Q} \overline{C}\right)^{-1} \left(\mathbf{G}^T L + \overline{C}^T \mathbf{Q} \mathbf{A}^N\right) \mathbf{x}(k)$$

which is a function of $\mathbf{Q}$. As a result, the states $\mathbf{x}(k+j)$, $j = 1, \cdots, N$ and the optimal value J^* of J are also the function of $\mathbf{Q}$. J^* is given by

$$J^*(\mathbf{Q}, N, \lambda) = \mathbf{x}^T(k) P_N \mathbf{x}(k) - \mathbf{x}^T(k)\mathbf{C}^T \mathbf{C}\mathbf{x}(k) \tag{21}$$

where P_N is the solution to the RDE (7). From Theorem 1, in order to achieve closed-loop stability we need $P_0 - P_1 \succeq 0$. That means, the end-point state weighting matrix $\mathbf{Q}$ has to be chosen appropriately. One way to obtain a reasonable $\mathbf{Q}$ is by posing an optimization problem as follows:

$$\begin{aligned} \min_{\mathbf{Q}} \quad & J^*(\mathbf{Q}, N, \lambda) - \mathbf{x}^{*T}(k+N)\mathbf{Q}\mathbf{x}^*(k+N) \\ \text{subject to} \quad & P_0 - P_1 \succeq 0 \end{aligned} \tag{22}$$

where

$$\begin{aligned} & J^*(\mathbf{Q}, N, \lambda) - \mathbf{x}^{*T}(k+N)\mathbf{Q}\mathbf{x}^*(k+N) \\ &= \mathbf{x}^* T(k+N)\mathbf{C}^T \mathbf{C}\mathbf{x}^*(k+N) + \\ &\quad \sum_{j=0}^{N-1} \left\{ \mathbf{x}^{*T}(k+j)\mathbf{C}^T \mathbf{C}\mathbf{x}^*(k+j) + \right. \\ &\qquad \left. \lambda \Delta \mathbf{u}^{*T}(k+j)\Delta \mathbf{u}^*(k+j) \right\} \\ &\qquad - \mathbf{x}^T(k)\mathbf{C}^T \mathbf{C}\mathbf{x}(k) \end{aligned} \tag{23}$$

where, $\mathbf{x}^*(k+j)$, $j = 1, 2, \cdots, N$, is the optimal state due to $\Delta \mathbf{u}^*$. So, $\mathbf{x}^*(k+j)$ is also a function of $\mathbf{Q}$. Clearly, if we minimize the cost in Eq. (23) by appropriately choosing $\mathbf{Q}$, we will get the optimal performance. Using Eq. (7), the constraint (22) can be re-written as

$$\begin{aligned} & \mathbf{A}^T(\mathbf{Q} + \mathbf{C}^T \mathbf{C})\mathbf{A} - \\ &\quad \mathbf{A}^T(\mathbf{Q} + \mathbf{C}^T \mathbf{C})\mathbf{B}[\mathbf{B}^T(\mathbf{Q} + \mathbf{C}^T \mathbf{C})\mathbf{B} + \lambda \mathbf{I}]^{-1} \\ &\qquad \mathbf{B}^T(\mathbf{Q} + \mathbf{C}^T \mathbf{C})\mathbf{A} - \mathbf{Q} \preceq 0. \end{aligned} \tag{24}$$

Then, the modified statement of $\mathbf{Q}$ synthesis optimization problem can be given as

$$\begin{aligned} \min_{\mathbf{Q}} \quad & \mathbf{x}^T(k) P_N \mathbf{x}(k) - \mathbf{x}^T(k)\mathbf{C}^T \mathbf{C}\mathbf{x}(k) \\ & \qquad - \mathbf{x}^{*T}(k+N)\mathbf{Q}\mathbf{x}^*(k+N) \quad (25) \\ \text{subject to} \quad & \text{Constraint (24)} \end{aligned}$$

The problem (25) is a constrained nonlinear optimization problem and one can only aim for best possible local minimum. The optimal solution will depend on the initial value. One can use a Sequential Quadratic Programming (SQP)-based optimization routine in Matlab to solve this problem. If we assume that the weighting matrix $\mathbf{Q}$ is diagonal, the computational complexity can be greatly reduced. The aim is to obtain a better $\mathbf{Q}$ than that is selected by trial and error.

V. Numerical Example

In this section, a numerical example is given to demonstrate the stable reconfiguration capability of the GPC control methodology using a short-period approximation model of a civil transport aircraft. The case of actuator failure considered is the saturation of elevator in the middle of its operating range. The reconfiguration strategy presented previously is used to recover from this failure in stable manner. The short-period approximation model of an aircraft considered is:

$$\begin{aligned} \dot{x}(t) &= \begin{bmatrix} 0 & -0.6839 \\ 2.0000 & -1.5087 \end{bmatrix} x(t) \\ &+ \begin{bmatrix} -0.0665 & -0.0029 & -0.0029 \\ -0.1723 & -0.0086 & -0.0086 \end{bmatrix} u(t) \\ y(t) &= \begin{bmatrix} 0 & 0.1250 \end{bmatrix} x(t) \end{aligned}$$

where the inputs are elevator deflection, left aileron deflection, and right aileron deflection. The output is the pitch rate. The eigenvalues of the open-loop system are $-0.7543 \pm 0.8937i$. The maneuver under consideration is the step change in the pitch-rate. The initial actuator configuration is such that the desirable input is only the elevator input to accomplish this task. The left and right ailerons are considered as redundant actuators and are to be used if elevator fails. The desirable pitch rate to be achieved is assumed to be 5.73 deg/sec. For demonstrating reconfiguration capability, it is assumed that the elevator input freezes at -7.73 deg deflection before the system output (pitch rate) reaches its desired value (5.73 deg/sec). As a result, a constant elevator input

is continuously acting on the system. At this stage reconfiguration comes in effect and the ailerons have to make-up for the control input to reach to the desired steady-state pitch rate value.
Reconfiguration and GPC control law design: The reconfiguration matrix Q_{rc} has its initial settings as:

$$Q_{rc} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix},$$

The outer product of Q_{rc} with the system transfer matrix yields the following system

$$\begin{aligned} \dot{x}(t) &= \begin{bmatrix} 0.0000 & -0.6839 \\ 2.0000 & -1.5087 \end{bmatrix} x(t) \\ &+ \begin{bmatrix} -0.2500 & 0 & 0 \\ 0.0000 & 0 & 0 \end{bmatrix} u(t) \\ y(t) &= \begin{bmatrix} 0.0862 & -0.0317 \end{bmatrix} x(t) \end{aligned} \tag{26}$$

The discretization of the system of Eq. (26) with the sampling rate of $T = 0.05s$ and transformation to an equivalent system with $\Delta u(k)$ as input yields the following system:

$$\begin{aligned} &x(k+1) = \\ &\begin{bmatrix} 0.9983 & -0.0329 & -0.0125 & 0 & 0 \\ 0.0963 & 0.9257 & -0.0006 & 0 & 0 \\ 0 & 0 & 1.0000 & 0 & 0 \\ 0 & 0 & 0 & 1.0000 & 0 \\ 0 & 0 & 0 & 0 & 1.0000 \end{bmatrix} x(k) \\ &+ \begin{bmatrix} -0.0125 & 0 & 0 \\ -0.0006 & 0 & 0 \\ 1.0000 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \Delta u(k) \\ &y(k) = \begin{bmatrix} 0.0862 & -0.0317 & 0 & 0 & 0 \end{bmatrix} x(k) \end{aligned} \tag{27}$$

Now, choosing the GPC design parameters as: $N_1 = 1$, $N_2 = N_u = 5$, $\lambda = 15$ and the end-point weighting matrix $\mathbf{Q} = diag\{50.4860\ 17.2823\ 20.0000\ 1.1292\ 1.1303\}$, we can obtain a stabilizing controller. These Riccati solutions satisfy the conditions of Theorem 1 as the eigenvalues of the difference $P_0 - P_1$ are

$$\begin{aligned} \text{Eig}(P_0 - P_1) &= \begin{bmatrix} 11.4335 & 0 & 2.4226 & 0 & 0 \end{bmatrix} \succeq 0 \\ \Rightarrow\ P_0 - P_1 &\succeq 0. \end{aligned}$$

As the elevator input freezes at -7.73 deg the system is reconfigured and ailerons are actuated to compensate for the short of elevator input. The reconfiguration matrix Q_{rc} is given by:

$$Q_{rc} = \begin{bmatrix} 0 & 1 & 1 \end{bmatrix}. \tag{28}$$

Again, discretization of the system with sampling time, $T = 0.05s$ and transformation to a new system with $\Delta u(k)$ as input yields:

$$\begin{aligned} &x(k+1) = \\ &\begin{bmatrix} 0.9983 & -0.0329 & 0 & -0.0006 & -0.0006 \\ 0.0963 & 0.9257 & 0 & -0.0017 & -0.0017 \\ 0 & 0 & 1.0000 & 0 & 0 \\ 0 & 0 & 0 & 1.0000 & 0 \\ 0 & 0 & 0 & 0 & 1.0000 \end{bmatrix} x(k) \\ &+ \begin{bmatrix} 0 & -0.0006 & -0.0006 \\ 0 & -0.0017 & -0.0017 \\ 0 & 0 & 0 \\ 0 & 1.0000 & 0 \\ 0 & 0 & 1.0000 \end{bmatrix} \Delta u(k) \\ &y(k) = \begin{bmatrix} 0.0000 & 0.0313 & 0 & 0 & 0 \end{bmatrix} x(k) \end{aligned} \tag{29}$$

Now, if we define the new GPC parameters to be $N_1 = 1$, $N_2 = N_u = 5$, $\lambda = 5$, and the end-point weighting matrix as $\mathbf{Q} = diag\{251\ 90\ 0\ 8\ 8\}$, we get

$$\begin{aligned} &\text{Eig}(P_0 - P_1) = \\ &\qquad \begin{bmatrix} 0.0001 & 12.6096 & 4.9236 & 4.9231 & 0 \end{bmatrix} \succeq 0 \\ &\Rightarrow\ P_0 - P_1 \succeq 0. \end{aligned}$$

Thus the conditions in the Theorem 1 are satisfied and the GPC control law (Eq. (6)) stabilizes the system.
The simulation results for this case are shown in Figures 1-4. As shown in Fig. 1, the elevator input freezes approximately at -7.73 deg. At this time ailerons take over as a result of reconfiguration and ailerons deflection reaches a steady-state value of approximately -5.6 deg as pitch rate reaches to desired value of 5.73 deg/sec. Note that, in this case, the row rank of the left hand side matrix in Eq. (16) is 3, which is less than $n = 5$, i.e., the condition (16) is satisfied. We get the desired final state value from Eq. (17). As seen in Fig. 2, overall tracking performance is very satisfactory without steady-state error. The performance function and tracking error time history is shown in Figs. 3 and 4, respectively.

The initial selection of $\mathbf{Q}$ given above was based on trial and error. In the next step, the $\mathbf{Q}$-synthesis method given in section IV was used to search the weights. By picking the initial $\mathbf{Q} = diag\{4000\ 4000\ 0\ 20\ 20\}$, the optimized $\mathbf{Q}$ was found to be $\mathbf{Q} = diag\{4982.6\ 1778.1\ 0.1\ 5.0\ 5.0\}$. The simulation results for the optimized $\mathbf{Q}$ are shown in Figures 5-8. The optimized $\mathbf{Q}$ gave better output response than initial $\mathbf{Q}$.

Although the $\mathbf{Q}$-synthesis method given in section IV results in a constrained nonlinear optimization problem which is difficult to solve, it still has merit as it enables to find $\mathbf{Q}$ which guarantees the stability and yields better performance as demonstrated in this numerical example.

VI. Conclusions

This paper presented a reconfigurable GPC control architecture. The stability of control law was ensured by imposing end-point state weighting matrix in the control cost. A method was given to compute the desired end-point state from desired output. This method guarantees the zero steady state error. A constrained nonlinear optimization methodology was presented for synthesis of end-point weighting matrix $\mathbf{Q}$ which optimizes desirable performance function. A numerical example consisting of longitudinal control of aircraft was presented to demonstrate the effectiveness of the proposed methodology.

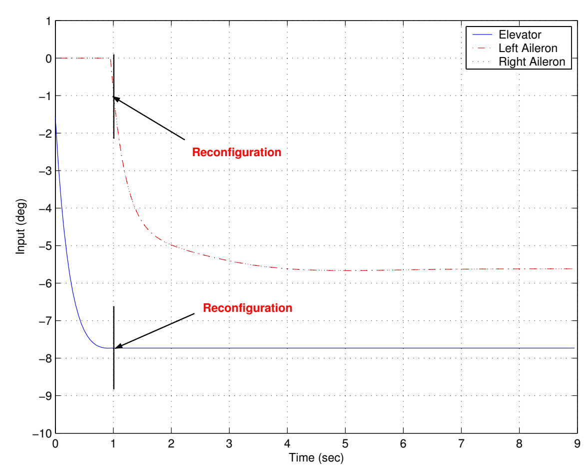

Fig. 1. The GPC control law

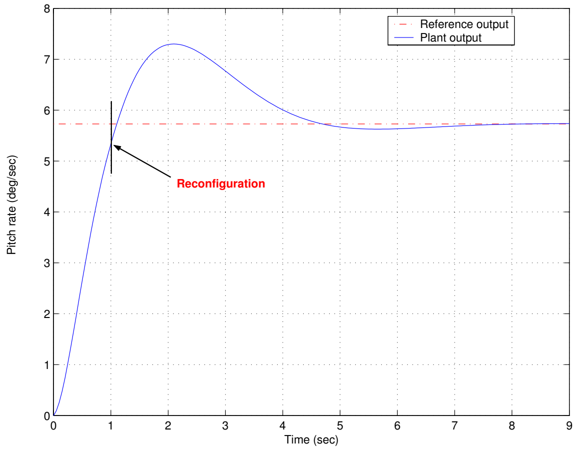

Fig. 2. The Plant Output

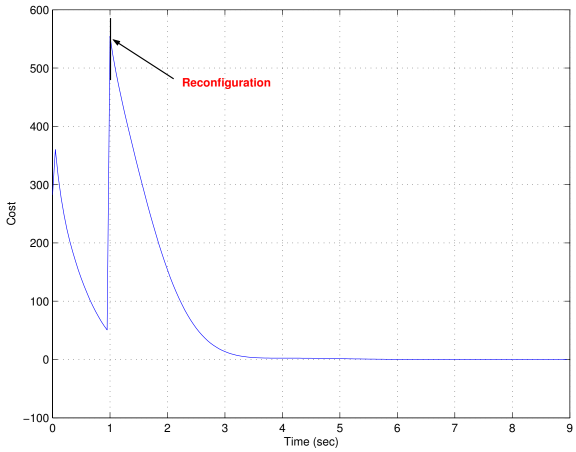

Fig. 3. The Cost Function

REFERENCES

[1] Clarke, D. W., Mohtadi, C., and Tuffs, P. S., 1987, "Generalized Predictive Control- Part I. The Basic Algorithm," Automatica, 23(2), pp. 137-148.

[2] Clarke, D. W., Mohtadi, C., and Tuffs, P. S., 1987, "Generalized Predictive Control- Part II. Extensions and Interpretations," Automatica, 23(2), pp. 149-160.

[3] Clarke, D. W., Mohtadi, C., 1989, "Properties of Generalized Predictive Control," Automatica, 25(6), pp. 859-875.

[4] Hess, R. A., Jung, Y. C., 1989, "An Application of Generalized Predictive Control to Rotorcraft Terrain-Following Flight," IEEE Transaction on Systems, Man and Cybernetics, 19(5), pp. 955-962.

[5] Bitmead, R. R., Gevers, M., and Wertz, V., 1990, *Adaptive Optimal Control: The Thinking Man's GPC*, Prentice Hall, Englewood Cliffs, New Jersey.

[6] Demircioglu, H., Clarke, D. W., 1993, "Generalized predictive control with end-point state weighting," IEE Proceedings-D, Vol. 140, No.4, pp. 275-282.

[7] Shi, J., Kelkar, A.G., and Soloway, D., 2003, "GPC-Based Stable Reconfigurable Control," IMECE2003-42627, Proceedings to ASME International Mechanical Engineering Congress and Exposition, Washington, DC.

[8] Soloway, D., Shi, J., and Kelkar, A., 2004, "GPC-Based Stable Reconfigurable Control," NASA/TP–2004-212823, Moffett Field, CA.

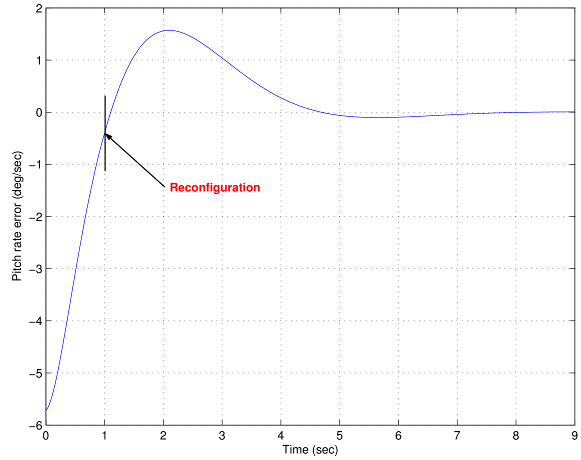

Fig. 4. The Plant Output Error

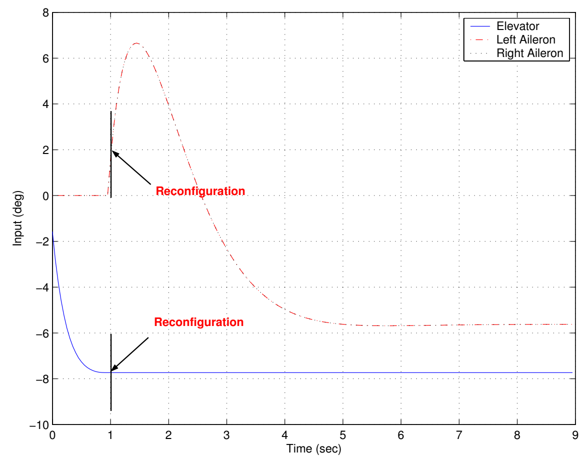

Fig. 5. The GPC control law

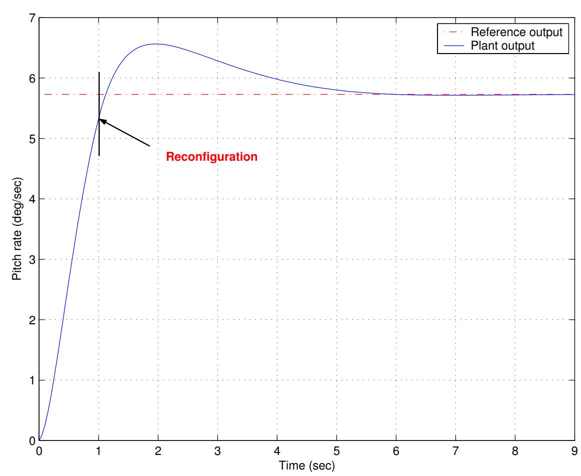

Fig. 6. The Plant Output

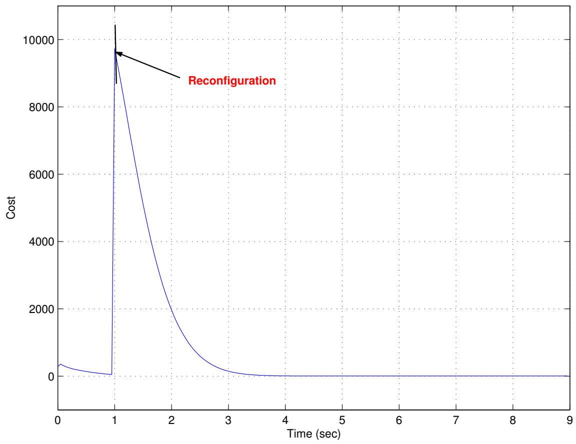

Fig. 7. The Cost Function

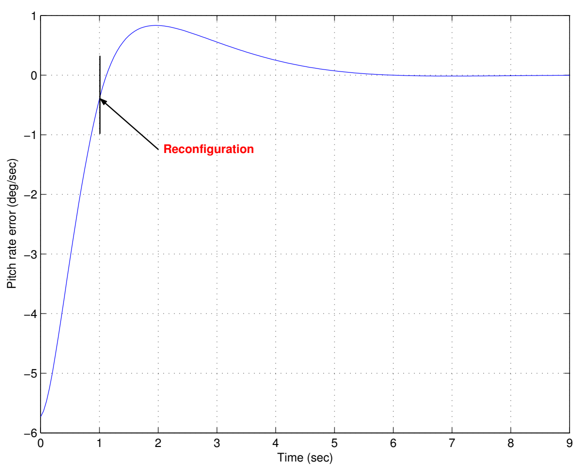

Fig. 8. The Plant Output Error