



The recursive generalized predictive feedback control: theory and experiments

S.-M. Moon, R.L. Clark*, D.G. Cole

Department of Mechanical Engineering and Materials Science, Duke University, Durham, NC 27708, USA

Received 27 June 2002; accepted 30 October 2003

Abstract

The proposed adaptive control algorithm combines the recursive least-squares system identification algorithm and the generalized predictive control (GPC) design algorithm, referred to as recursive generalized predictive control (RGPC.) In the GPC design process, the prediction horizon and control horizon are the constants to be chosen. Two new parameters are defined to describe the effects of the prediction and control horizons and those parameters provide the effective ranges of the horizons. The RGPC algorithm adjusts the control penalty based on the stability of a closed-loop system model. A time-varying algorithm for the control penalty allows to design an aggressive controller. The multi-sampling rate algorithm is added between the system identification and the control design in order to design a higher order controller. The RGPC algorithm is applied to three different systems: a cantilevered beam, a sound enclosure, and an optical jitter suppression testbed.

© 2003 Published by Elsevier Ltd.

1. Introduction

Predictive control, including long-range predictive control (LRPC) [1], can be thought of as an extension of the one-step-ahead predictive control architecture [2–5]. Early predictive control was based on the simple impulse [6] or step [7] response models. These discrete-time control methods provide good control performance but are limited to stable models. Later methods were developed using a more efficient model, such as a controlled autoregressive integrated moving average (CARIMA) [8,9] or CARMA [10] model. Of these, generalized predictive control (GPC) by Clarke and his co-workers has been shown to be particularly effective by removing the shortcomings of the earlier design models [8,9,11].

*Corresponding author. Tel.: +1-919-660-5435; fax: +1-919-660-8963.

E-mail address: rcclark@duke.edu (R.L. Clark).

The GPC method can be related to the linear quadratic (LQ) optimal control design method [12]. The GPC solution approaches the LQ regulator by extending the control and prediction horizons to very large values. The prediction and control horizons are the finite horizons of system output and input prediction, respectively. However, there are also significant differences. Weighting matrices are used as tuning parameters in the LQ approach whereas prediction and control horizons are used in the GPC approach. In the LQ approach, the Riccati equation is solved [13–15]. In the GPC approach, the solution is obtained using the pseudo-inverse of a matrix that results from posing the problem as a least-squares minimization.

One of the extensions of GPC is the continuous generalized predictive control (CGPC) algorithm [16]. The CGPC algorithm rederives the discrete-time GPC in a continuous-time setting [8,9,11]. The CGPC algorithm solves problems with discrete-time methods such as numerical sensitivity, sample rate selection, and non-minimum phase zeros [17–20]. Further research has been performed to solve the stability problem of CGPC [21,22]. The CGPC method guarantees closed-loop stability only in certain limiting cases [16,21]. The CGPC with guaranteed closed-loop stability is referred to as stable continuous generalized predictive control (SCGPC) [23]. Several discrete-time GPC methods were proposed to solve the stability problem [24–27]. The guaranteed stable GPC algorithm modifies the constraint limits on the inputs in the presence of a class of bounded disturbances [26]. Another stable GPC algorithm is proposed for a nominal model [28].

Recent developments based on GPC concepts are related to multivariable control [29–32] and adaptive control [8,9,33]. The GPC concept for multivariable control and multi-input and multi-output (MIMO) control was proposed using a non-minimal linear model [34]. The minimum variance control [35] and generalized minimum variance controllers [36] were extended for a MIMO linear system. These controllers cannot properly handle both an unstable system and a system with non-minimum phase. In order to control these systems, the MIMO GPC algorithm using the extended horizon approach was developed [37], and later was further extended for a deterministic system [38,39].

In this research, the adaptive control algorithm, which combines the process of the recursive least-squares (RLS) system identification and the process of the generalized predictive control (GPC) design, is developed and referred to as recursive generalized predictive control (RGPC). The algorithm is designed for real-time control and applied to actual testbed. The experimental results from this research will demonstrate the feasibility of the proposed control algorithm.

2. Recursive system identification

Using the linear combinations of past output and input measurements as states, a system can be identified in ARX representation,

$$y(t) = - \sum_{i=1}^p \alpha_i y(t-i) + \sum_{i=1}^p \beta_i u(t-i) + e(t), \quad (1)$$

where $y(t)$ is the measured output and $u(t)$ is the input to the system [40–43]. For m outputs and n inputs system, each α_i ($i = 1, 2, \dots, p$) is an $m \times m$ matrix and each β_i is an $m \times n$ matrix. The RLS

system identification process is the experimental determination of the model parameters in such a way that the output computed from the model in Eq. (1) is as close as possible with the measured output in the sense of least squares.

The cost function is defined to be

$$V = \frac{1}{2} \sum_{i=1}^t \rho^{t-i} (y(i) - \theta \varphi^T(i))^2, \quad (2)$$

where

$$\theta = [\alpha_1 \quad \cdots \quad \alpha_p \quad \beta_1 \quad \cdots \quad \beta_p], \quad (3)$$

$$\varphi(t) = [-y^T(t-1) \quad \cdots \quad -y^T(t-p) \quad u^T(t-1) \quad \cdots \quad u^T(t-p)] \quad (4)$$

and $0 < \rho \leq 1$. The parameter ρ is the forgetting factor or discounting factor. The cost function of Eq. (2) implies that a temporal weighting of the data is introduced. The most recent data is given unit weight, but data that are M time steps old is weighted by ρ^M [44,45].

The minimization of the cost function, given in Eq. (2), results in

$$\theta = \mathbf{y} \mathbf{W} \Psi^T (\Psi \mathbf{W} \Psi^T)^{-1}, \quad (5)$$

where

$$\mathbf{y} = [y(1) \quad y(2) \quad \cdots \quad y(t)], \quad (6)$$

$$\Psi = [\varphi^T(1) \quad \varphi^T(2) \quad \cdots \quad \varphi^T(t)] \quad (7)$$

and a data weighting matrix, \mathbf{W} , is a diagonal matrix with the weights in the diagonal.

Using the non-recursive system parameter estimation given in Eq. (5), the recursive parameter estimation algorithm using the least-squares method can be obtained [14,40,45–47]. The model parameter, θ , is obtained recursively by

$$\theta(t) = \theta(t-1) + \{y(t) - \theta(t-1)\varphi^T(t)\}L(t), \quad (8)$$

where the correcting factor, $L(t)$, becomes

$$L(t) = \frac{\varphi(t)\mathbf{P}(t-1)}{\rho + \varphi(t)\mathbf{P}(t-1)\varphi^T(t)} \quad (9)$$

and

$$\mathbf{P}(t) = \frac{1}{\rho} \mathbf{P}(t-1) [\mathbf{I} - \varphi^T(t)L(t)]. \quad (10)$$

When the recursive equation is used in all time steps, $t > 0$, it is convenient to start the algorithm with zero initial condition for $\theta(0)$ and

$$\mathbf{P}(0) = \mathbf{P}_0, \quad (11)$$

where \mathbf{P}_0 is positive definite. One choice is

$$\mathbf{P}_0 = \alpha \mathbf{I} \quad (12)$$

for $\alpha > 0$ and the matrix \mathbf{I} is an identity matrix. The constant, α , then becomes a parameter to be chosen. In the literature, it is shown that the value of α is related to the convergence

speed of the model parameters [45,48]. The constant α is commonly assigned a small value ($0 < \alpha \leq 1$) for high signal-to-noise ratio and a large value ($\alpha > 1$) for low signal-to-noise ratio.

The last parameter to be chosen is the forgetting factor, ρ , in Eqs. (9) and (10). The choice of the forgetting factor is presented in several adaptive control literature [45,47,48]. A simple strategy is

$$\rho = \exp[-t_s/T_f], \tag{13}$$

where t_s is the sampling time in seconds and T_f is the time constant for exponential forgetting [45].

3. The generalized predictive control

The generalized predictive controller is designed for a system written in ARX form such as

$$\sum_{i=0}^p \alpha_i y(t-i) = \sum_{i=0}^p \beta_i u(t-i), \tag{14}$$

where $\alpha_0 = \mathbf{I}(m)$ and $\beta_0 = \text{zeros}(m, n)$. The matrix $\mathbf{I}(m)$ is an $m \times m$ identity matrix and the matrix $\text{zeros}(m, n)$ is an $m \times n$ matrix with zero entries. The parameters with subscript zero are used for the convenience in later expressions.

The system equation (14) can be written in matrix form

$$\mathbf{A} \mathbf{y}_p = \mathbf{B} \mathbf{u}_p, \tag{15}$$

where the output parameter matrix \mathbf{A} and the input parameter matrix \mathbf{B} are written as

$$\mathbf{A} = [\alpha_0 \quad \alpha_1 \quad \cdots \quad \alpha_p], \quad \mathbf{B} = [\beta_0 \quad \beta_1 \quad \cdots \quad \beta_p]. \tag{16}$$

Constructing the q step predictor at time t and partitioning into past and future parts, the following matrix equation is obtained:

$$[\mathbf{F} \mid \mathbf{G}] \begin{bmatrix} \mathbf{y}_p(t) \\ \mathbf{y}_f(t) \end{bmatrix} = [\mathbf{H} \mid \mathbf{J}] \begin{bmatrix} \mathbf{u}_p(t) \\ \mathbf{u}_f(t) \end{bmatrix}, \tag{17}$$

where the matrices \mathbf{F} and \mathbf{G} are the coefficient matrices for the past and future output data, $\mathbf{y}_p(t)$ and $\mathbf{y}_f(t)$, respectively, and written as

$$[\mathbf{F} \mid \mathbf{G}] = \left[\begin{array}{ccc|cccccc} \alpha_p & \cdots & \alpha_1 & \alpha_0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \vdots & 0 & \alpha_p & \vdots & \alpha_1 & \alpha_0 & 0 & \cdots & 0 \\ \vdots & \vdots & 0 & \alpha_p & \vdots & \alpha_1 & \alpha_0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \vdots & \vdots \\ 0 & \cdots & 0 & 0 & 0 & \alpha_p & \cdots & \alpha_1 & \alpha_0 \end{array} \right] \tag{18}$$

and the matrices \mathbf{H} and \mathbf{J} are the coefficient matrices for the past and future control input data, $\mathbf{u}_p(t)$ and $\mathbf{u}_f(t)$, respectively, and written as

$$[\mathbf{H} | \mathbf{J}] = \left[\begin{array}{ccc|cccccc} \beta_p & \cdots & \beta_1 & \beta_0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots & \vdots & \vdots \\ \vdots & 0 & \beta_p & \vdots & \beta_1 & \beta_0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & 0 & \beta_p & \vdots & \beta_1 & \beta_0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \vdots & \vdots & \vdots \\ 0 & \cdots & 0 & 0 & 0 & \beta_p & \cdots & \beta_1 & \beta_0 \end{array} \right]. \tag{19}$$

It should be noted that the model parameters in Eqs. (18) and (19) are reordered from model parameters given in Eq. (3). Defining the prediction horizon $h_p = p + q - 1$, the size of each coefficient matrix can be obtained as given in Table 1. Note that the matrix \mathbf{G} is a square matrix.

The $mp \times 1$ past output history vector, $\mathbf{y}_p(t)$, and the $m(h_p + 1) \times 1$ future output vector, $\mathbf{y}_f(t)$, can be written as

$$\mathbf{y}_p(t) = \begin{bmatrix} y(t-p) \\ \vdots \\ y(t-1) \end{bmatrix}, \quad \mathbf{y}_f(t) = \begin{bmatrix} y(t) \\ \vdots \\ y(t+h_p) \end{bmatrix}. \tag{20}$$

The $np \times 1$ input past history vector, $\mathbf{u}_p(t)$, and the $n(h_p + 1) \times 1$ future input vector, $\mathbf{u}_f(t)$, can be written as

$$\mathbf{u}_p(t) = \begin{bmatrix} u(t-p) \\ \vdots \\ u(t-1) \end{bmatrix}, \quad \mathbf{u}_f(t) = \begin{bmatrix} u(t) \\ \vdots \\ u(t+h_p) \end{bmatrix}. \tag{21}$$

Solving Eq. (17) for the future outputs and normalizing the coefficients by multiplying the inverse of the coefficient matrix of the future output, \mathbf{G} , Eq. (17) can be written as

$$\mathbf{y}_f(t) = \mathbf{T}_c \mathbf{u}_f(t) + \mathbf{B}_c \mathbf{u}_p(t) + \mathbf{A}_c \mathbf{y}_p(t), \tag{22}$$

where the normalized coefficient matrices are written as

$$\mathbf{T}_c = \mathbf{G}^{-1} \mathbf{J}, \quad \mathbf{B}_c = \mathbf{G}^{-1} \mathbf{H}, \quad \mathbf{A}_c = -\mathbf{G}^{-1} \mathbf{F}. \tag{23}$$

Table 1
Size of the coefficient matrix in Eq. (17)

Matrix	Size
\mathbf{F}	$m(h_p + 1) \times mp$ block upper triangular matrix
\mathbf{G}	$m(h_p + 1) \times m(h_p + 1)$ block lower triangular matrix
\mathbf{H}	$m(h_p + 1) \times np$ block upper triangular matrix
\mathbf{J}	$m(h_p + 1) \times n(h_p + 1)$ block lower triangular matrix
\mathbf{M}	$m(h_p + 1) \times n_d p$ block upper triangular matrix
\mathbf{N}	$m(h_p + 1) \times n_d(h_p + 1)$ block lower triangular matrix

The cost function is defined as

$$J(t) = \sum_{j=0}^{h_p} \{e(t+j)^T e(t+j)\} + \lambda \sum_{j=0}^{h_c} \{u^T(t+j)u(t+j)\}, \quad (24)$$

where $e(t) = y_r(t) - y(t)$ and $y_r(t)$ is the reference output. Parameter λ is the input weighting factor or control penalty which is a positive scalar. Parameter h_p is the prediction horizon and h_c is the control horizon which is less than the prediction horizon, i.e., $h_c \leq h_p$ [29,49].

The future input, \mathbf{u}_f , that satisfies the equation, $dJ(k)/d\mathbf{u}_f = 0$, is

$$\mathbf{u}_f(t) = -[\mathbf{T}_c^T \mathbf{T}_c + \lambda \mathbf{I}]^{-1} \mathbf{T}_c^T (\mathbf{B}_c \mathbf{u}_p(t) + \mathbf{A}_c \mathbf{y}_p(t) - \mathbf{y}_r(t)), \quad (25)$$

where $\mathbf{y}_r(t)$ is the reference output vector or the set points.

The control input is then defined by the first n rows of the future input, i.e.,

$$\mathbf{u}(t) = \text{the first } n \text{ rows of } \{-[\mathbf{T}_c^T \mathbf{T}_c + \lambda \mathbf{I}]^{-1} \mathbf{T}_c^T (\mathbf{B}_c \mathbf{u}_p(t) + \mathbf{A}_c \mathbf{y}_p(t) - \mathbf{y}_r(t))\}. \quad (26)$$

The control input given in Eq. (26) occurs when the control horizon is equal to the prediction horizon. When the control horizon is less than the prediction horizon, the solution that minimizes the cost function, $J(t)$, in Eq. (24), can be achieved by reducing the size of the coefficient matrix \mathbf{J} in Eq. (19) to be $\mathbf{J}(1 : m(h_p + 1), 1 : n(h_c + 1))$.

4. Stability analysis

The control input, given in Eq. (26), can be written as

$$u(t) = \mathbf{K}_y y(t) + \mathbf{K}_u u(t) + \mathbf{K}_r r(t), \quad (27)$$

where \mathbf{K}_y , \mathbf{K}_u , and \mathbf{K}_r are the polynomial matrices for output, input, and reference point, respectively,

$$\mathbf{K}_y = \alpha_1^c z^{-1} + \alpha_2^c z^{-2} + \dots + \alpha_p^c z^{-p}, \quad (28)$$

$$\mathbf{K}_u = \beta_1^c z^{-1} + \beta_2^c z^{-2} + \dots + \beta_p^c z^{-p} \quad (29)$$

and

$$\mathbf{K}_r = \eta_1^c z^1 + \eta_2^c z^2 + \dots + \eta_{h_p}^c z^{h_p}, \quad (30)$$

where the superscript c is used to distinguish the control parameters from the model parameters. Recall that z is the shift variable, for example, $z^{-1}y(t) = y(t-1)$. Note that the polynomial \mathbf{K}_r is written for the vector of set positions for the current instant. A block diagram using the control input in Eq. (27) is illustrated in Fig. 1, where \mathbf{G}_{sys} denotes a transfer function matrix of the plant.

Using block-diagram algebra operations, the relationship between y and r is obtained as

$$\mathbf{G}_{yr} = \mathbf{G}_{\text{sys}} [(\mathbf{I}_n - \mathbf{K}_u) - \mathbf{K}_y \mathbf{G}_{\text{sys}}]^{-1} \mathbf{K}_r \quad (31)$$

and the relationship between u and r is written as

$$\mathbf{G}_{ur} = [(\mathbf{I}_n - \mathbf{K}_u) - \mathbf{K}_y \mathbf{G}_{\text{sys}}]^{-1} \mathbf{K}_r. \quad (32)$$

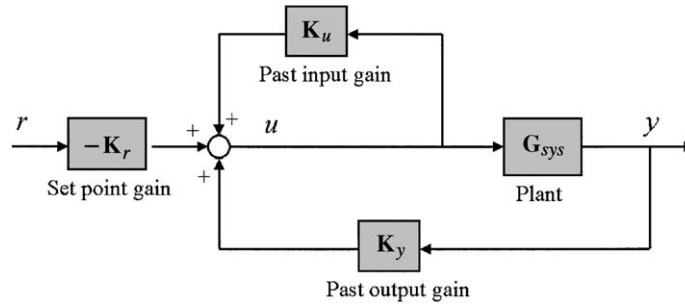


Fig. 1. Block diagram of GPC.

For a multiple-inputs and multiple-outputs system, consider the right-coprime factorization of the system transfer function matrix

$$\mathbf{G}_{sys} = \mathbf{N}_{sys} \mathbf{D}_{sys}^{-1} \quad (33)$$

where \mathbf{N}_{sys} and \mathbf{D}_{sys} , are the numerator and denominator matrices, respectively.

With Eq. (33), the transfer function matrices in Eqs. (31) and (32) can be written as

$$\mathbf{G}_{yr} = \mathbf{N}_{sys} [(\mathbf{I}_n - \mathbf{K}_u) \mathbf{D}_{sys} - \mathbf{K}_y \mathbf{N}_{sys}]^{-1} \mathbf{K}_r \quad (34)$$

and

$$\mathbf{G}_{ur} = \mathbf{D}_{sys} [(\mathbf{I}_n - \mathbf{K}_u) \mathbf{D}_{sys} - \mathbf{K}_y \mathbf{N}_{sys}]^{-1} \mathbf{K}_r. \quad (35)$$

Writing the following polynomial matrix:

$$\mathbf{\Omega} = (\mathbf{I}_n - \mathbf{K}_u) \mathbf{D}_{sys} - \mathbf{K}_y \mathbf{N}_{sys}, \quad (36)$$

there may be pole-zero cancellations between (1) \mathbf{K}_r and $\mathbf{\Omega}$ in Eqs. (34) and (35); (2) \mathbf{N}_{sys} and $\mathbf{\Omega}$ in Eq. (34); (3) \mathbf{D}_{sys} and $\mathbf{\Omega}$ in Eq. (35).

Poles and zeros of a transfer function matrix are defined as the roots of the pole polynomial and zero polynomial of the transfer function matrix written in the Smith–McMillan form [50,51]. In pole-zero cancellations, however, there are two kinds of pole-zero cancellations; one is the stable pole-zero cancellation and the other is the unstable pole-zero cancellation. The stable pole-zero cancellation, also referred to as ‘pole shifting’, is a very common and effective control design strategy [15,52]. On the other hand, unstable pole-zero cancellation must not happen. When an unstable pole is cancelled by a zero, it creates hidden modes within a system, which leads to a system that is not internally stable.

As a result, the closed-loop system consisting of the system, \mathbf{G}_{sys} , and the controller in Eq. (27) is stable if and only if all the roots of the polynomial equation of

$$\det[\mathbf{\Omega}] = 0 \quad (37)$$

lie inside the unit circle.

5. Control parameter

In order to discuss the effect of the prediction and control horizons, two new parameters are defined—prediction horizon parameter, L_{hp} , and control horizon parameter, L_{hc} . The effective ranges of the prediction horizon and control horizon can be observed by plotting each prediction and control horizon as a function of the prediction horizon parameter, L_{hp} , and the control horizon parameter, L_{hc} , respectively.

5.1. Prediction horizon parameter L_{hp}

In order to express the effect of a greater prediction horizon, the GPC algorithm is rederived in the following manner. For a given set of system parameters, the coefficient matrices are given as Eqs. (18) and (19). When one-step greater prediction horizon is added to the coefficient matrices, the coefficient matrices with a greater prediction horizon can be written as

$$\hat{\mathbf{F}} = \begin{bmatrix} \mathbf{F} \\ \mathbf{F}_{hp} \end{bmatrix}, \quad \hat{\mathbf{G}} = \begin{bmatrix} \mathbf{G} & \mathbf{0} \\ \mathbf{G}_{hp} & \alpha_0 \end{bmatrix}, \quad \hat{\mathbf{H}} = \begin{bmatrix} \mathbf{H} \\ \mathbf{H}_{hp} \end{bmatrix}, \quad \hat{\mathbf{J}} = \begin{bmatrix} \mathbf{J} & \mathbf{0} \\ \mathbf{J}_{hp} & \beta_0 \end{bmatrix}, \quad (38)$$

where the coefficient matrix with caret accent (^) denotes the coefficient matrices with a greater prediction horizon, $h_p + 1$. Vectors \mathbf{F}_{hp} and \mathbf{G}_{hp} are generated from the system output parameters, α_i , and vectors \mathbf{H}_{hp} and \mathbf{J}_{hp} are generated from the system input parameters, β_i . It is noted that vectors \mathbf{F}_{hp} and \mathbf{H}_{hp} are zero vectors when the prediction horizon is larger than the number of system parameters. In the later derivation, $\alpha_0 = 1$ and $\beta_0 = 0$.

From Eq. (38), the normalized coefficient matrices with a greater prediction horizon are obtained as

$$\hat{\mathbf{T}}_c = \hat{\mathbf{G}}^{-1} \hat{\mathbf{J}} = \begin{bmatrix} \mathbf{T}_c & \mathbf{0} \\ -\mathbf{G}_{hp} \mathbf{T}_c + \mathbf{J}_{hp} & \mathbf{0} \end{bmatrix}, \quad (39)$$

$$\hat{\mathbf{A}}_c = \hat{\mathbf{G}}^{-1} \hat{\mathbf{F}} = \begin{bmatrix} -\mathbf{A}_c \\ \mathbf{G}_{hp} \mathbf{A}_c + \mathbf{F}_{hp} \end{bmatrix}, \quad (40)$$

$$\hat{\mathbf{B}}_c = \hat{\mathbf{G}}^{-1} \hat{\mathbf{H}} = \begin{bmatrix} \mathbf{B}_c \\ -\mathbf{G}_{hp} \mathbf{B}_c + \mathbf{H}_{hp} \end{bmatrix}, \quad (41)$$

where \mathbf{T}_c , \mathbf{A}_c and \mathbf{B}_c , are the normalized coefficient matrices, given in Eq. (23). In these equations, the following Matrix Inversion Lemma is used:

$$\begin{bmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{B} & \mathbf{C} \end{bmatrix}^{-1} = \begin{bmatrix} \mathbf{A}^{-1} & \mathbf{0} \\ -\mathbf{C}^{-1} \mathbf{B} \mathbf{A}^{-1} & \mathbf{C}^{-1} \end{bmatrix}. \quad (42)$$

Since $\hat{\mathbf{T}}_c^T \hat{\mathbf{T}}_c + \lambda \mathbf{I}$ is a block diagonal matrix, its inverse can be obtained easily:

$$(\mathbf{T}_c^T \mathbf{T}_c + \lambda \mathbf{I})^{-1} = \begin{bmatrix} \{\bullet\}^{-1} & \mathbf{0} \\ \mathbf{0} & \lambda^{-1} \end{bmatrix}, \quad (43)$$

where

$$\{\bullet\} = \mathbf{T}_c^T \mathbf{T}_c + \lambda \mathbf{I} + (-\mathbf{G}_{hp} \mathbf{T}_c + \mathbf{J}_{hp})^T (-\mathbf{G}_{hp} \mathbf{T}_c + \mathbf{J}_{hp}). \quad (44)$$

Hence, the GPC gains of the past output and input data are obtained as

$$\begin{aligned} (\hat{\mathbf{T}}_c^T \hat{\mathbf{T}}_c + \lambda \mathbf{I})^{-1} \hat{\mathbf{T}}_c^T \hat{\mathbf{A}}_c &= \text{the first } n \text{ rows of} \\ \{-\{\bullet\}^{-1} \mathbf{T}_c^T \mathbf{A}_c + \{\bullet\}^{-1} (-\mathbf{G}_{hp} \mathbf{T}_c + \mathbf{J}_{hp})^T (\mathbf{G}_{hp} \mathbf{A}_c + \mathbf{F}_{hp})\} \end{aligned} \quad (45)$$

and

$$\begin{aligned} (\hat{\mathbf{T}}_c^T \hat{\mathbf{T}}_c + \lambda \mathbf{I})^{-1} \hat{\mathbf{T}}_c^T \hat{\mathbf{B}}_c &= \text{the first } n \text{ rows of} \\ \{-\{\bullet\}^{-1} \mathbf{T}_c^T \mathbf{B}_c + \{\bullet\}^{-1} (-\mathbf{G}_{hp} \mathbf{T}_c + \mathbf{J}_{hp})^T (-\mathbf{G}_{hp} \mathbf{B}_c + \mathbf{H}_{hp})\}. \end{aligned} \quad (46)$$

Eqs. (45) and (46) represent the GPC gain for a greater prediction horizon, $h_p + 1$, obtained from those for the one-step short prediction horizon, h_p . The term of $(\mathbf{T}_c^T \mathbf{T}_c + \lambda \mathbf{I})^{-1} \mathbf{T}_c^T \mathbf{A}_c$ in Eq. (45) and the term of $(\mathbf{T}_c^T \mathbf{T}_c + \lambda \mathbf{I})^{-1} \mathbf{T}_c^T \mathbf{B}_c$ in Eq. (46) are the control gains for the one-step short prediction horizon, given in Eq. (26). All other terms in Eqs. (45) and (46) are additional resulting from the one-step added prediction horizon.

There could be several ways to define the effect of an extra term. However, it can be seen that the term of $-\mathbf{G}_{hp} \mathbf{T}_c + \mathbf{J}_{hp}$ appears in each extra term (see Eq. (39)). In Eq. (39), the matrices \mathbf{G}_{hp} and \mathbf{J}_{hp} represent the last row of the coefficient matrix for a greater prediction horizon, $h_p + 1$, while \mathbf{T}_c is defined for h_p .

With the fact that the longer prediction horizon improves the response [11,16,32], the following parameter is defined to represent the effect of an extra prediction horizon:

$$L = \frac{1}{\text{trace}((- \mathbf{G}_{hp} \mathbf{T}_c + \mathbf{J}_{hp})^T (- \mathbf{G}_{hp} \mathbf{T}_c + \mathbf{J}_{hp}))}. \quad (47)$$

Since the parameter L is defined from the extra term by adding one-step prediction horizon, the parameter L is considered as the magnitude difference in the controller gains. Now, let L_p be the parameter value of L when the prediction horizon is same as the order of the system model, p , and define the normalized parameter value, L_{hp} , as

$$L_{hp} = \frac{L}{L_p} = \frac{\text{trace}((- \mathbf{G}_{hp} \mathbf{T}_c + \mathbf{J}_{hp})^T (- \mathbf{G}_{hp} \mathbf{T}_c + \mathbf{J}_{hp}))|_{h_p=p}}{\text{trace}((- \mathbf{G}_{hp} \mathbf{T}_c + \mathbf{J}_{hp})^T (- \mathbf{G}_{hp} \mathbf{T}_c + \mathbf{J}_{hp}))}, \quad (48)$$

where $L_{hp} > 0$. Hence, the normalized parameter L_{hp} represents the magnitude difference ratio of the controller gains with respect to the gains when $h_p = p$.

Using the normalized parameter, L_{hp} , the effective range of the prediction horizon can be chosen. For a given set of model parameters, the parameter value of L_{hp} is calculated by increasing the prediction horizon. When the change of L_{hp} is small, it tells that the changes of the controller gains are small with respect to that of the controller gains when the prediction horizon is the same as the model order. Let \hat{h}_p be the prediction horizon when the change of L_{hp} is small, then prediction horizons near \hat{h}_p provide the effective range of the prediction horizon. The ‘small’ variation of L_{hp} can be observed by plotting the value of L_{hp} as a function of h_p .

5.2. Control horizon parameter L_{hc}

As mentioned earlier, shorter control horizon than the prediction horizon can be defined. Shorter control horizon assumes that the control input beyond the control horizon is constant, and the control gains can be obtained by reducing the size of the coefficient matrix \mathbf{J} . As the control horizon decreases, the number of columns of matrix \mathbf{J} decreases while the number of rows of the matrix remains the same. Other coefficient matrices, \mathbf{F} , \mathbf{G} , and \mathbf{H} , remain the same.

Let the matrices $\hat{\mathbf{F}}_{hp}$, $\hat{\mathbf{G}}_{hp}$, $\hat{\mathbf{H}}_{hp}$, and $\hat{\mathbf{J}}_{hp}$ be the coefficient matrices when the prediction horizon is \hat{h}_p . In order to express the effect of a shorter control horizon, the coefficient matrix $\hat{\mathbf{J}}_{hp}$ is partitioned by the last column,

$$\hat{\mathbf{J}}_{hp} = [\mathbf{J}_{hc} \quad \mathbf{J}_{hc-1}], \quad (49)$$

where the matrix \mathbf{J}_{hc} is an $m(\hat{h}_p + 1) \times m\hat{h}_p$ matrix.

Using the coefficient matrix $\hat{\mathbf{J}}_{hp}$ in Eq. (49), the GPC gains are obtained. The normalized coefficient matrix of the future output is written as

$$\hat{\mathbf{T}}_c = \hat{\mathbf{G}}_{hp}^{-1} \hat{\mathbf{J}}_{hp} = [\hat{\mathbf{T}}_{c1} \quad \hat{\mathbf{T}}_{c2}], \quad (50)$$

where $\hat{\mathbf{T}}_{c1} = \hat{\mathbf{G}}_{hp}^{-1} \mathbf{J}_{hc}$ and $\hat{\mathbf{T}}_{c2} = \hat{\mathbf{G}}_{hp}^{-1} \mathbf{J}_{hc-1}$.

The GPC gains of the past output and input data are obtained as

$$(\hat{\mathbf{T}}_c^T \hat{\mathbf{T}}_c + \lambda \mathbf{I})^{-1} \hat{\mathbf{T}}_c^T \hat{\mathbf{A}}_c = \text{the first } n \text{ rows of } \{(\hat{\mathbf{T}}_{c11} \hat{\mathbf{T}}_{c1}^T + \hat{\mathbf{T}}_{c12} \hat{\mathbf{T}}_{c2}^T) \hat{\mathbf{A}}_c\} \quad (51)$$

and

$$(\hat{\mathbf{T}}_c^T \hat{\mathbf{T}}_c + \lambda \mathbf{I})^{-1} \hat{\mathbf{T}}_c^T \hat{\mathbf{B}}_c = \text{the first } n \text{ rows of } \{(\hat{\mathbf{T}}_{c11} \hat{\mathbf{T}}_{c1}^T + \hat{\mathbf{T}}_{c12} \hat{\mathbf{T}}_{c2}^T) \hat{\mathbf{B}}_c\}, \quad (52)$$

where $\hat{\mathbf{A}}_c = -\hat{\mathbf{G}}_{hp}^{-1} \hat{\mathbf{F}}_{hp}$ and $\hat{\mathbf{B}}_c = \hat{\mathbf{G}}_{hp}^{-1} \hat{\mathbf{H}}_{hp}$. The matrices $\hat{\mathbf{T}}_{c11}$ and $\hat{\mathbf{T}}_{c12}$ are the elements of the matrix $(\hat{\mathbf{T}}_c^T \hat{\mathbf{T}}_c + \lambda \mathbf{I})^{-1}$, which is written as

$$(\hat{\mathbf{T}}_c^T \hat{\mathbf{T}}_c + \lambda \mathbf{I})^{-1} = \begin{bmatrix} \hat{\mathbf{T}}_{c11} & \hat{\mathbf{T}}_{c12} \\ \hat{\mathbf{T}}_{c21} & \hat{\mathbf{T}}_{c22} \end{bmatrix}, \quad (53)$$

where

$$\begin{aligned} \hat{\mathbf{T}}_{c11} &= \mathbf{M}, \\ \hat{\mathbf{T}}_{c12} &= -\mathbf{M} \hat{\mathbf{T}}_{c1}^T \hat{\mathbf{T}}_{c2} (\hat{\mathbf{T}}_{c2}^T \hat{\mathbf{T}}_{c2} + \lambda \mathbf{I})^{-1}, \\ \hat{\mathbf{T}}_{c21} &= -(\hat{\mathbf{T}}_{c2}^T \hat{\mathbf{T}}_{c2} + \lambda \mathbf{I})^{-1} \hat{\mathbf{T}}_{c2}^T \hat{\mathbf{T}}_{c1} \mathbf{M}, \\ \hat{\mathbf{T}}_{c22} &= (\hat{\mathbf{T}}_{c2}^T \hat{\mathbf{T}}_{c2} + \lambda \mathbf{I})^{-1} (\mathbf{I} + \hat{\mathbf{T}}_{c2}^T \hat{\mathbf{T}}_{c1} \mathbf{M} \hat{\mathbf{T}}_{c1}^T \hat{\mathbf{T}}_{c2} (\hat{\mathbf{T}}_{c2}^T \hat{\mathbf{T}}_{c2} + \lambda \mathbf{I})^{-1}) \end{aligned}$$

and

$$\mathbf{M} = ((\hat{\mathbf{T}}_{c1}^T \hat{\mathbf{T}}_{c1} + \lambda \mathbf{I}) - \hat{\mathbf{T}}_{c1}^T \hat{\mathbf{T}}_{c2} (\hat{\mathbf{T}}_{c2}^T \hat{\mathbf{T}}_{c2} + \lambda \mathbf{I})^{-1} \hat{\mathbf{T}}_{c2}^T \hat{\mathbf{T}}_{c1})^{-1}. \quad (54)$$

In Eq. (53), the following Matrix Inversion Lemma is used:

$$\begin{bmatrix} \mathbf{A} & \mathbf{B}^T \\ \mathbf{B} & \mathbf{C} \end{bmatrix}^{-1} = \begin{bmatrix} \mathbf{K} & -\mathbf{K} \mathbf{B}^T \mathbf{C}^{-1} \\ -\mathbf{C}^{-1} \mathbf{B} \mathbf{K} & \mathbf{C}^{-1} + \mathbf{C}^{-1} \mathbf{B} \mathbf{K} \mathbf{B}^T \mathbf{C}^{-1} \end{bmatrix}, \quad (55)$$

where $\mathbf{K} = (\mathbf{A} - \mathbf{B}^T \mathbf{C}^{-1} \mathbf{B})^{-1}$.

In Eqs. (51) and (52), all terms except $(\hat{\mathbf{T}}_{c1}^T \hat{\mathbf{T}}_{c1} + \lambda \mathbf{I})^{-1} \hat{\mathbf{T}}_{c1} \hat{\mathbf{B}}_c$ and $(\hat{\mathbf{T}}_{c1}^T \hat{\mathbf{T}}_{c1} + \lambda \mathbf{I})^{-1} \hat{\mathbf{T}}_{c1} \hat{\mathbf{A}}_c$ are generated by the one-step control horizon. Among all extra terms, the term of $\hat{\mathbf{T}}_{c2} (\hat{\mathbf{T}}_{c2}^T \hat{\mathbf{T}}_{c2} + \lambda \mathbf{I})^{-1} \hat{\mathbf{T}}_{c2}^T$ exists in every term (see Eq. (54)).

Based on the fact that the shorter control horizon improves the system response [11,16,32], the following parameter is defined to represent the effect of a shorter control horizon:

$$L_{hc} = \frac{1}{\text{trace}(\mathbf{S}^T \mathbf{S})}, \quad (56)$$

where

$$\mathbf{S} = \hat{\mathbf{T}}_{c2} (\hat{\mathbf{T}}_{c2}^T \hat{\mathbf{T}}_{c2} + \varepsilon \mathbf{I}_4)^{-1} \hat{\mathbf{T}}_{c2}^T \quad (57)$$

and λ is replaced by ε . With a very small positive number, ε , the parameter L_{hc} , can be defined for $h_c = \hat{h}_p$ and $\hat{h}_p - 1$. When $h_c = \hat{h}_p$, the matrix $\hat{\mathbf{T}}_{c2}$ is zero matrix, and $L_{hc} \approx \infty$. When $h_c = \hat{h}_p - 1$,

$$\hat{\mathbf{T}}_{c2} = \begin{bmatrix} 0 & 0 \\ \vdots & \vdots \\ \beta_1 & 0 \end{bmatrix}, \quad (58)$$

and $L_{hc} \approx 1$.

6. Examples

In this section, the prediction horizon parameter, L_{hp} , and the control horizon parameter, L_{hc} , are numerically computed for given model parameters. The model parameters are obtained from experimental data. The numerical examples illustrated in this section will provide better understanding for those parameters.

6.1. Identified beam model

The prediction horizon parameters, L_{hp} , and the control horizon parameter, L_{hc} , are computed for the identified model parameters of a cantilevered beam. A detailed testbed setup is described in Section 8.1. A band-limited random signal is applied to a cantilevered beam using piezoelectric devices and an accelerometer sensor signal is used for output measurement. Using sampled input and output data at 250 Hz, the system model with order $p = 16$ is identified using the non-recursive method described in Section 2. The identified model is obtained as

$$\mathbf{A}(1)y(t) + \cdots + \mathbf{A}(p+1)y(t-p) = \mathbf{B}(1)u(t) + \cdots + \mathbf{B}(p+1)u(t-p), \quad (59)$$

where the output parameter vector, \mathbf{A} , and the input parameter vector, \mathbf{B} , are

$$\mathbf{A} = [1, -0.044, -0.023, -0.086, 0.190, -0.022, -0.150, -0.016, -0.273, \\ -0.102, 0.125, -0.061, 0.165, 0.038, -0.229, -0.047, -0.176],$$

$$\mathbf{B} = [0, -0.005, -0.016, 0.026, -0.008, 0.007, 0.021, -0.039, 0.003, 0.002, \\ -0.015, 0.027, -0.0004, -0.004, 0.007, -0.014, 0.002].$$

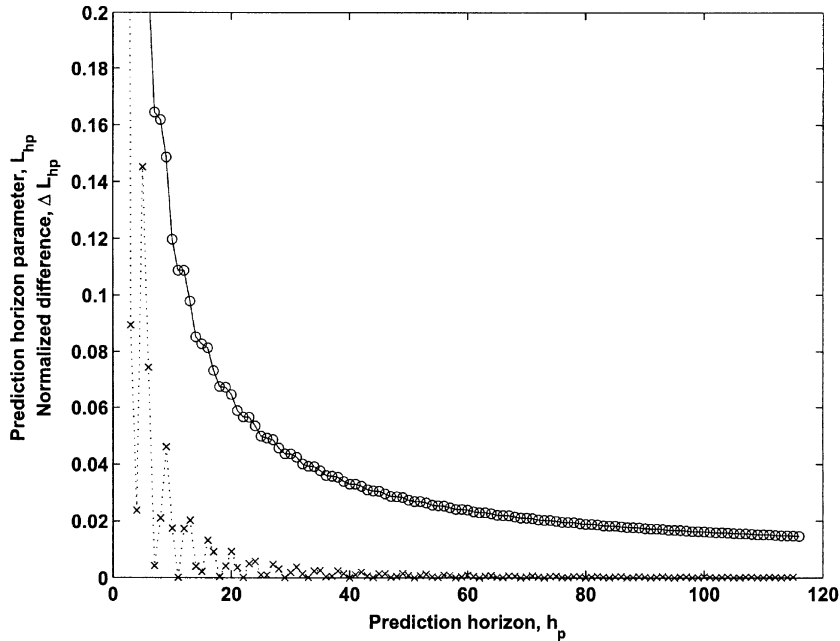


Fig. 2. The prediction horizon parameter, L_{hp} (—○—), and the normalized difference, ΔL_{hp} (· · × · · ·), for an identified beam model

The identified model, given in Eq. (59), is simplified to a second order model,

$$y(t) + 1.2967y(t - 1) + 0.9928y(t - 2) = 0.9964u(t - 2), \tag{60}$$

which is obtained from the complex conjugate set of the dominant poles [15,50,53]. The model given in Eq. (60) describes the dominant system response. The prediction horizon parameter L_{hp} is obtained using the parameters given in Eq. (60) and illustrated by the solid line in Fig. 2. The prediction horizon is varied from $h_p = 2$ to $7p$. As seen in the figure, the value of prediction horizon parameter, L_{hp} , becomes smaller as the prediction horizon, h_p , becomes greater.

In order to observe the changes of the prediction horizon parameter, let $L_{hp}(k)$ be the value of the prediction horizon parameter at $h_p = k$, and write the difference of the prediction horizon parameter, $\delta L_{hp}(k)$, as

$$\delta L_{hp}(k) = |L_{hp}(k) - L_{hp}(k - 1)|, \tag{61}$$

where $k \geq 2$. The difference $\delta L_{hp}(k)$ can be normalized by $\delta L_{hp}(2)$. The normalized difference, ΔL_{hp} , is illustrated by the dashed line in Fig. 2. As seen in the figure, the normalized difference ΔL_{hp} becomes smaller as the prediction horizon becomes greater. The normalized difference $\Delta L_{hp} \approx 1\%$ is a reasonable choice for the selection of the prediction horizon. The corresponding prediction horizon to $\Delta L_{hp} \approx 1\%$ is $h_p \approx 18$. The control horizon parameter L_{hc} is plotted as a function of a control horizon in Fig. 3. The smaller value of the control horizon parameter gives the shorter control horizon.

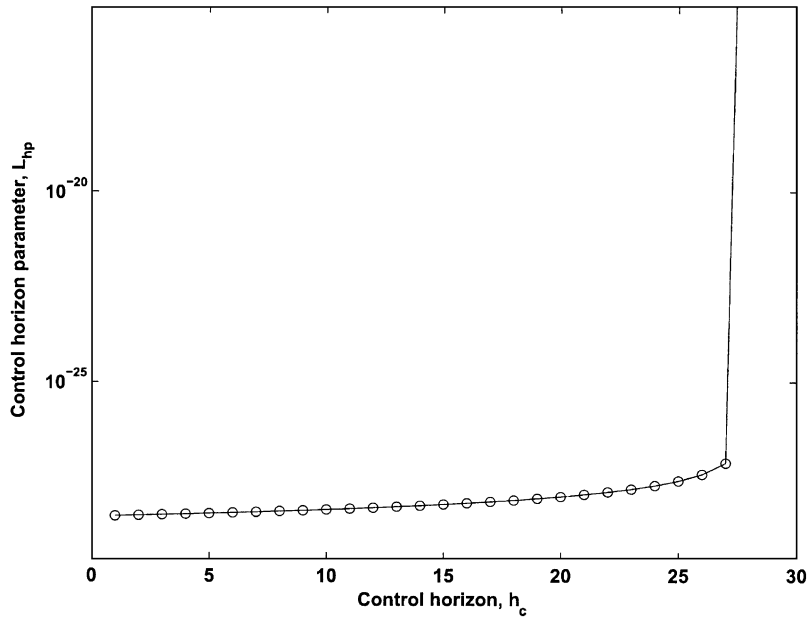


Fig. 3. The control horizon parameter, L_{hc} , for an identified beam model.

6.2. Identified acoustic enclosure model

The prediction horizon parameters, L_{hp} , and the control horizon parameter, L_{hc} , are computed for the identified model parameters of an acoustic enclosure. A detailed testbed setup is described in Section 8.2. A random signal is applied to an acoustic speaker and an acoustic microphone is used for output measurement. Using sampled input and output data at 500 Hz, the system model with order $p = 16$ is identified using the nonrecursive method. The identified model has the same form as given in Eq. (59) and the output parameter vector, \mathbf{A} , and the input parameter vector, \mathbf{B} , are obtained as

$$\mathbf{A} = [1, -0.061, -0.155, -0.245, 0.084, -0.069, 0.031, -0.062, -0.047, \\ -0.124, -0.300, 0.370, 0.023, -0.015, -0.137, 0.049, -0.084],$$

$$\mathbf{B} = [0, 0.018, 0.131, -0.044, -0.139, -0.101, -0.995, -0.777, 1.839, 1.499, \\ -0.107, -0.689, -0.318, -0.251, -0.113, 0.070, -0.047].$$

In order to observe the changes of the prediction horizon parameter, the identified model is simplified to a second order model,

$$y(t) + 0.7230y(t-1) + 0.9146y(t-2) = 0.9563u(t-2), \quad (62)$$

which describes the dominant system response. The prediction horizon parameter L_{hp} is obtained using the parameters given in Eq. (62) and illustrated by the solid line in Fig. 4. The prediction horizon is varied from $h_p = 2$ to $7p$. The normalized difference ΔL_{hp} is also illustrated by the dashed line in Fig. 4. The corresponding prediction horizon to $\Delta L_{hp} \approx 1\%$ is $h_p \approx 24$.

The control horizon parameter L_{hc} is plotted as a function of a control horizon in Fig. 5. The smaller value of the control horizon parameter gives the shorter control horizon and the difference of the control horizon parameter becomes smaller as the control horizon becomes shorter.

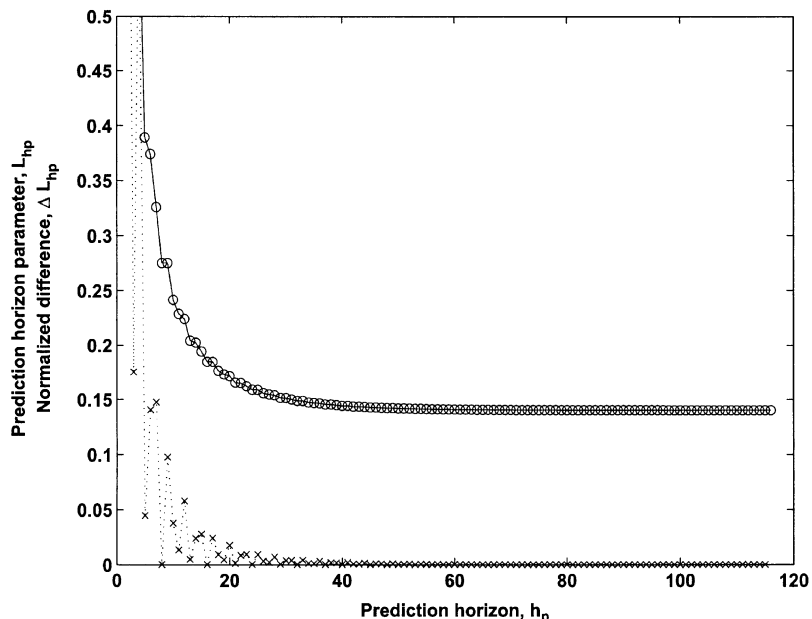


Fig. 4. The prediction horizon parameter, L_{hp} (—○—), and the normalized difference, ΔL_{hp} (· · × · ·), for an identified acoustic enclosure model.

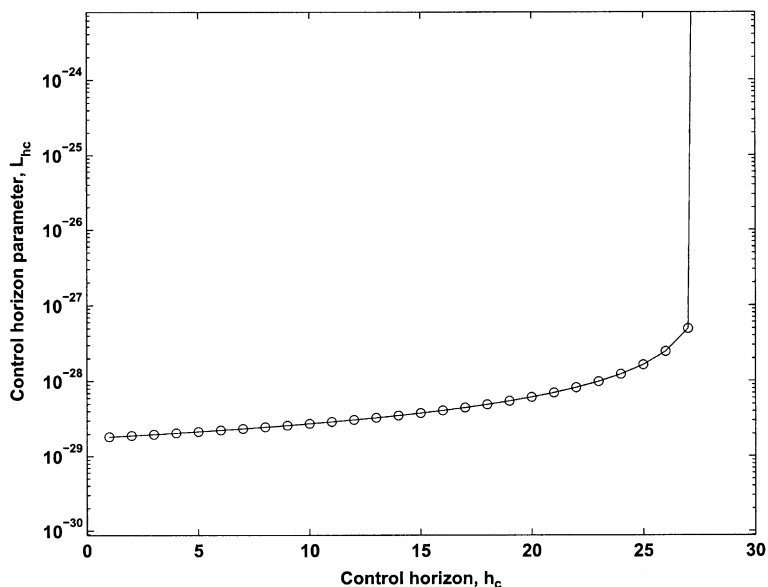


Fig. 5. The control horizon parameter, L_{hc} , for an identified acoustic enclosure model.

The longer prediction horizon and the shorter control horizon, which improve the system performance, can be assigned for a fixed-gain (non-adaptive) controller which is not restricted in time spent for the controller design. The computation time, however, is an important factor in the development of an adaptive algorithm because a controller must be designed in each sample period. A longer prediction horizon increases the size of the coefficient matrices (see Eq. (38)), resulting in a longer computational time. In such a case, the prediction horizon should be reduced and the prediction horizon parameter can be used for a reasonable selection of the prediction horizon in the discrete-time GPC design. The shorter control horizon reduces the size of the coefficient matrix (see Eq. (49)) and results in shorter computational time.

7. Recursive generalized predictive control

The block diagram of the overall RGPC algorithm is shown in Fig. 6. The RGPC algorithm starts with system identification using known control inputs and measured sensor outputs. A model is identified using the RLS algorithm. The controller is designed using the updated model. The control input is then calculated using the controller gains and measured data.

In the following sections, a multi-sampling-rate algorithm and a time-varying input weighting factor algorithm are introduced to design higher order controllers and aggressive controllers. A stability test algorithm is added to avoid implementation of unstable controllers.

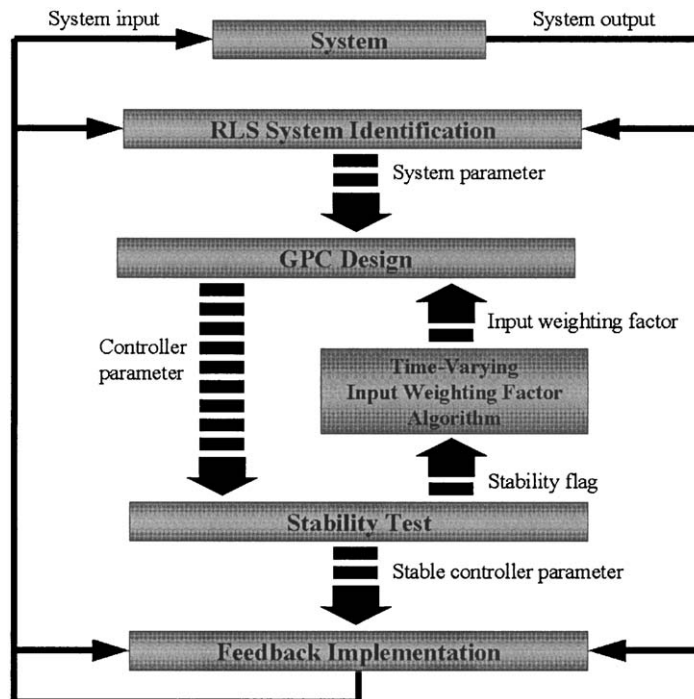


Fig. 6. The RGPC design process with multi-sampling-rate algorithm (solid line for faster sampling rate and dashed line for slower sampling rate) and time-varying input weighting factor algorithm.

7.1. Multi-sampling-rate algorithm

When RGPC is implemented in real-time, a multi-sampling-rate algorithm can be applied to design a higher order controller or a controller with greater prediction horizon. First, the RLS system identification is applied with the same sampling rate at the data acquisition rate since it updates the system parameters using matrix summations and multiplications without any time-consuming process such as matrix inversion. Second, assuming that the system parameters do not change significantly, a slower sampling rate than the data acquisition sampling rate is used for controller design because matrix inversion must be performed during control design (see Eq. (26)). The controller is fixed until a new controller is computed and updated. Third, the control input is calculated with the measured and stored data using the same sampling rate as that used for the data acquisition. Graphical implementation for multi-sampling-rate algorithm is shown in Fig. 6.

7.2. Time-varying input weighting factor

A time-varying input weighting factor algorithm is applied to solve the following problems. When a small positive constant value of the input weighting factor is chosen in the controller design, the controller can create an instability since the matrix \mathbf{T}_c in Eq. (23) is close to singular. Even if the closed-loop system model is stable, when applied to a real experimental setup, it may result in a large magnitude of initial control input, which may cause an overload in the experimental hardware. A large constant value of the input weighting factor, however, may reduce the magnitude of the initial control input, but limits performance.

To avoid those problems, a time-varying input weighting factor is added to the controller design,

$$\lambda_{new} = \lambda_{old} \pm \tau, \quad (63)$$

where λ_{new} is the updated input weighting factor from the previous input weighting factor, λ_{old} . The value of τ is added or subtracted, depending on the stability of the closed-loop system model. The negative sign is applied for stable controllers and the positive sign is applied for unstable controllers. The different values, τ , for stable controllers and unstable controllers can be assigned:

$$\lambda_{new} = \begin{cases} \lambda_{old} - \tau_1 & \text{for stable controller,} \\ \lambda_{old} + \tau_2 & \text{for unstable controller,} \end{cases} \quad (64)$$

where $\tau_1 > 0$ is the decrement of the input weighting factor when controllers are stable and $\tau_2 > 0$ is the increment of the input weighting factor when controllers are unstable.

By assigning a large initial value of the input weighting factor, a large magnitude of the initial control input can be avoided. Once the controller is designed and applied to the system, the input weighting factor will be updated based on the stability of the closed-loop system model. The value will be reduced for a stable controller and increased for an unstable controller until a stable controller is obtained. Graphical implementation for time-varying input weighting factor algorithm is shown in Fig. 6.

The value of τ in Eq. (63) or τ_1 and τ_2 in Eq. (64) can be scaled with respect to the input weighting factor. For example, when the input weighting factor is 0.01, the proper size of τ is

0.001, an order of magnitude less. When controllers are to be updated faster in case of instability, a larger value of τ_2 than τ_1 can be assigned.

To apply the time-varying input weighting factor to the system, the followings process must be added in the controller design. When the value of λ approaches a very small number, λ_{min} , the decrement should be zero because the input weighting factor is a positive real number. By adding this process, a zero or negative value of the input weighting factor can be avoided. Second, since the weighting factor is updated after the stability test, application of unstable controllers must be avoided. If unstable controller results upon checking for stability, the previous stable controller is implemented instead. By adding this process in the controller design, only stable controllers are applied to the system.

8. Experimental results

The RGPC algorithm is applied to three different testbeds: a cantilevered beam, a sound enclosure, and an optical jitter suppression testbed. The algorithm was implemented using a National Instrument PCI-6024E data acquisition board. The interface between the computer and the board are performed using MATLAB, Simulink, and xPC target.

8.1. Cantilevered beam

The first experimental system is a cantilevered beam (see Fig. 7). The cantilevered beam is configured with two piezoelectric devices and PCB Model 352C67 accelerometers mounted on the beam. One of the piezoelectric patches is used to disturb the beam and the other patch is used as a control source. In addition, Ithaco 24 dB/octave low-pass filters, set at a cutoff frequency of 125 Hz, and Krohn-Hite Model 7600 wideband amplifiers, set at a signal gain of 14 dB, are connected to the piezoelectric device. Another Ithaco 24 dB/octave low-pass filter, also set at 125 Hz, and a PCB model 480E09 signal conditioner are used for each accelerometer sensor. The sampling rate of the overall process is set at 250 Hz to reduce the vibration of the first two natural modes.

In this experiment, in addition to the error sensor output for control design, an extra sensor, whose signal is not used in the controller design, is mounted on the beam. The purpose of an extra sensor is to observe the performance at other than the location of the error sensor. For convenience, the extra sensor will be called the performance sensor. While the control algorithm is running, the signal from the performance sensor is measured with a Siglab spectrum analyzer and the frequency response function (FRF), the frequency domain representation of a process, is obtained by the discrete Fourier transforms (DFT) of input and output time responses [52].

Using the measured sensor signal and stored disturbance signal, the FRF is computed for the open-loop case (controller off) and the closed-loop case (controller on). When there is a magnitude reduction at a certain frequency, the control algorithm achieves the disturbance rejection at the sensor location.

Keeping a single control source and a single disturbance source configuration, two different output configurations are selected. The first setup is a single error sensor output and a single performance sensor output. The system related to the control design is a SISO system. The other

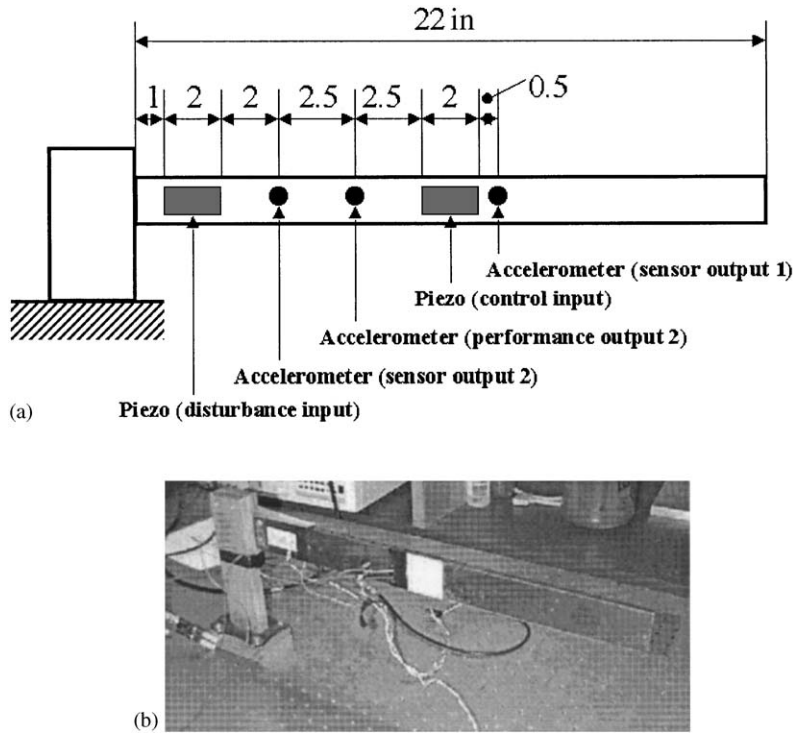


Fig. 7. Schematic diagram and picture of a cantilevered beam: (a) schematic diagram of beam testbed, (b) picture of beam testbed.

output setup is based upon two error sensor outputs and a performance sensor output. In this case, the system becomes a single-input and two-output system. All the output sensors are placed in order to observe the first two natural modes of the beam.

The RGPC feedback algorithm is applied to each output configuration. During each experiment, the unfiltered error sensor signal and the unfiltered performance sensor signal are measured using Siglab spectrum analyzer, and the FRF is obtained between a band-limited random disturbance applied to the voltages of the disturbance piezoelectric device and the voltage signal from each accelerometer sensor. The unfiltered signal is measured to observe the performance in the higher frequency range. Figs. 8–12 illustrate the open- and closed-loop FRF of each signal.

Figs. 8 and 9 show the computed open- and closed-loop FRFs of the first output configuration: a single error sensor output and a single performance sensor output. In each figure, the dashed line represents the open-loop process and the solid line represents the closed-loop process. Fig. 8 provides the vibration reduction at the location of the error sensor. The vibration at the first and the second modes, 32.5 and 90.0 Hz, is reduced 28.4 and 17.1 dB, respectively. It is also noted that no increase in response is observed in the higher frequency range (> 125 Hz). Fig. 9 exhibits the open- and closed-loop FRFs at the location of the performance sensor. The reduction of 28.3 and 17.3 dB at the first and second natural modes is achieved using the RGPC algorithm.

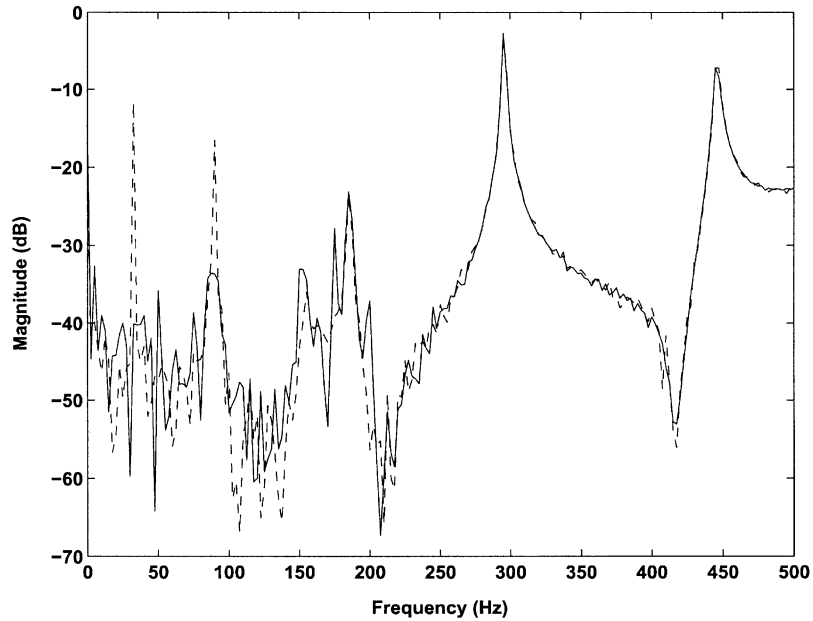


Fig. 8. Open-loop (dashed line) and closed-loop (solid line) FRF magnitude plot of the error sensor output of a cantilevered beam (single error sensor output case and $t_s = 250$ Hz).

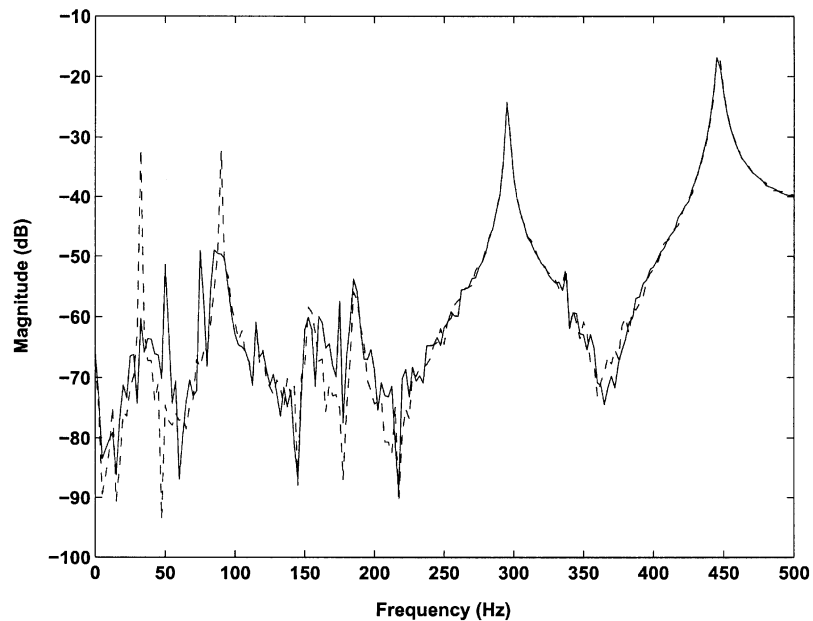


Fig. 9. Open-loop (dashed line) and closed-loop (solid line) FRF magnitude plot of the performance sensor output of a cantilevered beam (single error sensor output case and $t_s = 250$ Hz).

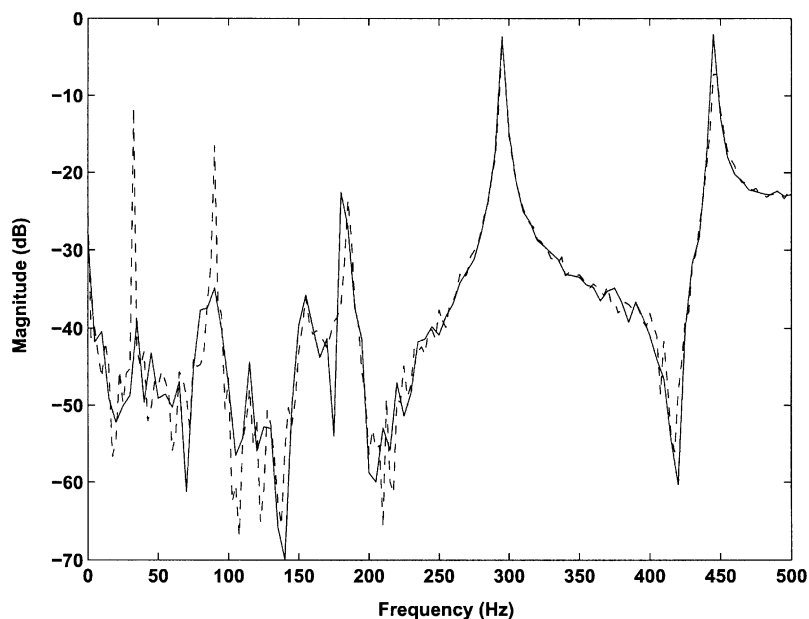


Fig. 10. Open-loop (dashed line) and closed-loop (solid line) FRF magnitude plot of the first feedback sensor output of a cantilevered beam (two error sensor output case and $t_s = 250$ Hz).

Figs. 10–12 show results for the second output configuration: two error sensor outputs and a single performance sensor output. Fig. 10 is the open- and closed-loop FRFs of the first error sensor output signal and Fig. 11 is that of the second error sensor output. The vibration at the first natural mode, 32.5 Hz, is reduced 32.3 and 28.8 dB at each location of the first and second errors sensor, respectively. The second mode, 90.0 Hz, is also reduced 18.3 and 16.9 dB by the feedback of two error sensor output signals. Fig. 12 shows the FRF of the performance sensor output. The mode at 32.5 Hz is reduced 30.0 dB and the mode at 90.0 Hz is reduced 17.1 dB.

In the control algorithm, the RGPC parameters are set the same for each output configuration. The order of the identified system and the controller is chosen to be 16 which is large enough to characterize the process. While the overall process including system identification and data acquisition is performed at a sampling rate of 250 Hz, the controller is updated at 1/2 the speed of the system identification, i.e., 125 Hz.

8.2. Sound enclosure

The application of the RGPC algorithm is used for acoustic noise control. The algorithm is tested on a rigid-walled acoustic enclosure with a square cross-section, shown in Fig. 13. The enclosure is configured with two DY-NAUDIO 15 W-75 acoustic loudspeakers placed at each end. In the experiments, one loudspeaker is used for the acoustic disturbance source and the other is used for the control source. Amplifiers are used to amplify both the band-limited (500 Hz) random disturbance noise and the control signal. A Brüel & Kjær type 4190 microphone sensor is used as a feedback error sensor. An Ithaco 24 dB/octave low-pass filter, set to a cutoff frequency

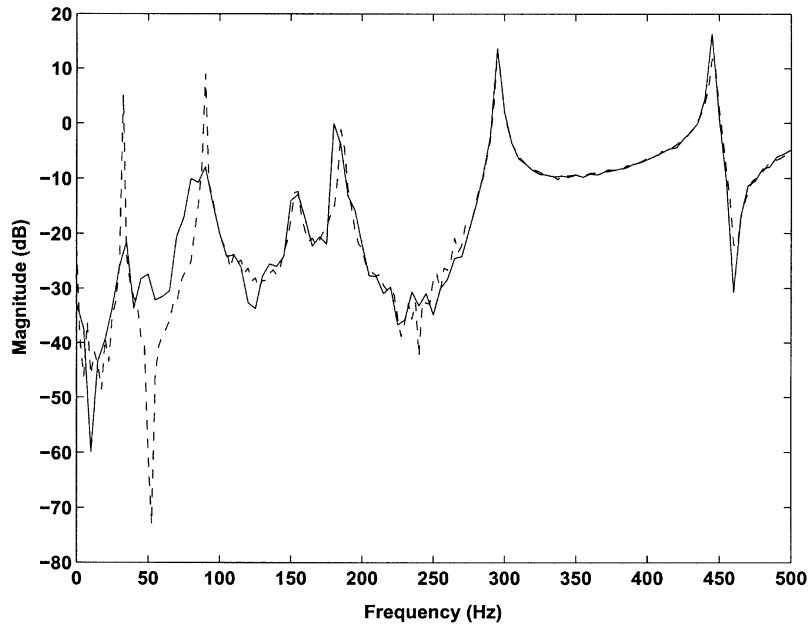


Fig. 11. Open-loop (dashed line) and closed-loop (solid line) FRF magnitude plot of the second feedback sensor output of a cantilevered beam (two error sensor output case and $t_s = 250$ Hz).

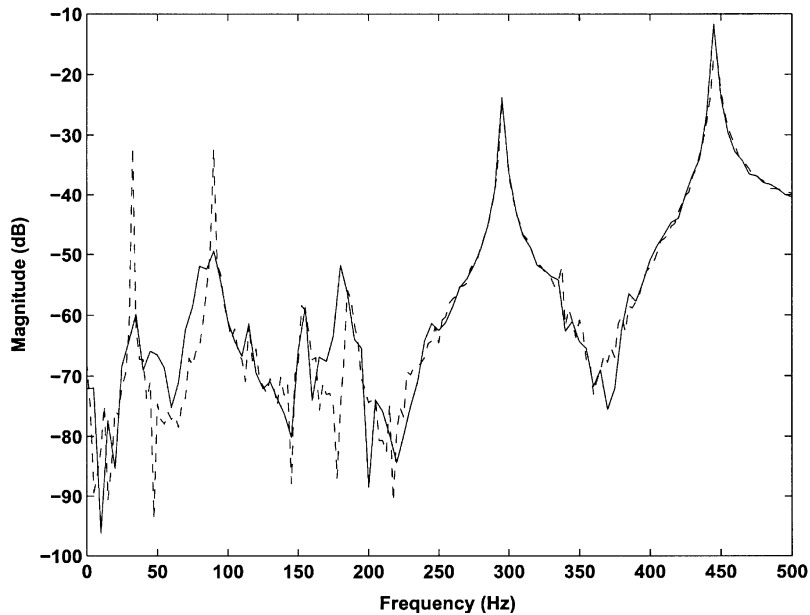


Fig. 12. Open-loop (dashed line) and closed-loop (solid line) FRF magnitude plot of the performance sensor output of a cantilevered beam (two error sensor output case and $t_s = 250$ Hz).

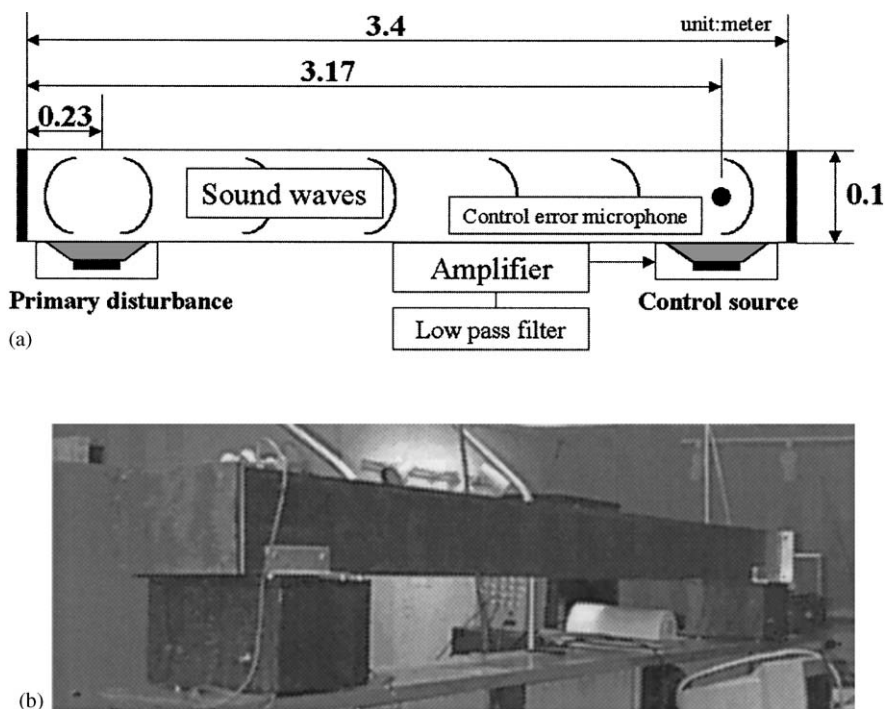


Fig. 13. Schematic diagram and picture of closed-closed sound enclosure configured with acoustic loudspeakers and microphone: (a) schematic diagram of acoustic testbed, (b) picture of acoustic testbed.

of 250 Hz, and a Brüel & Kjær type 2635 amplifier are connected to the microphone sensor. The feedback sensor microphone is placed above the control speaker diaphragm and fixed at the center of the square cross-section to minimize the acoustic reflection by the wall. The control objective is to minimize the sound pressure level around the acoustic microphone sensor.

The RGPC algorithm is applied while both endcaps are covered, which yields a closed–closed acoustic enclosure. A band-limited random signal is applied to the disturbance loudspeaker and the unfiltered microphone signal is measured in voltage using a Siglab spectrum analyzer. With the stored disturbance signal and the measured microphone signal, the averaged open- and closed-loop FRFs are computed. The FRF shows the frequency domain response between the disturbance signal and the error sensor signal [13,52]. The magnitude response of the FRF is presented in Fig. 14. The dashed line is when the controller is open-loop and the solid line is closed loop. The system model is identified from the data acquired at the sampling rate of 500 Hz and the controller is updated at $1/4$ the speed of the system identification process, i.e., 125 Hz. The order of model and controller is set to be 16.

The modes at frequencies of 54, 106, 155, and 205 Hz are reduced 9.0, 5.8, 6.3, and 13.2 dB, respectively. The integrated response of the enclosure is attenuated by approximately 2.1 dB at resonant frequencies between 50 and 250 Hz, without observed increase in response outside of the bandwidth (> 250 Hz).

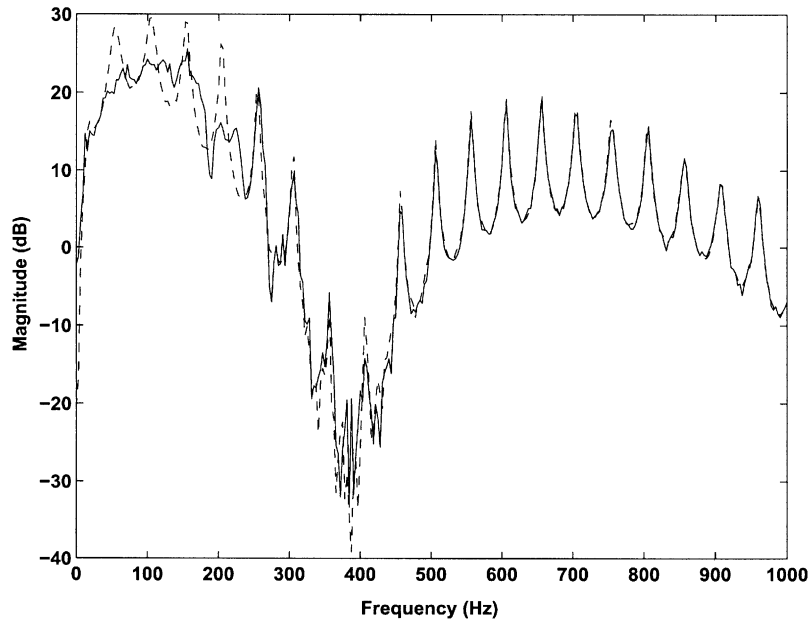


Fig. 14. Open-loop (dashed line) and closed-loop (solid line) FRF magnitude plot of the control error microphone signal for closed-closed acoustic enclosure ($t_s = 500$ Hz).

While the feedback controller for the closed-closed acoustic enclosure is running, the endcap on the control loudspeaker side is opened manually to yield a open-closed acoustic enclosure configuration. The endcap is removed slowly (~ 2 s) to prevent the overload that may happen to the experimental hardware. As soon as the cover is removed, the microphone sensor signal is measured using Siglab spectrum analyzer and the FRF with the disturbance signal is computed. The magnitude responses of the open- and closed-loop FRFs are shown in Fig. 15. The acoustic modes at 78, 128, 177, and 227 Hz are reduced 4.7, 7.4, 7.5, and 5.9 dB, respectively. The integrated response of the open-closed acoustic enclosure is attenuated by 2.2 dB at resonant frequencies between 50 and 250 Hz.

8.3. Optical jitter suppression testbed

The application of the RGPC algorithm is performed on the optical jitter testbed. The testbed, illustrated in Fig. 16, is in an anechoic chamber at Duke University. The purpose of an anechoic chamber is to minimize the jitter effect by extraneous, ambient, acoustic disturbance. Two Newport optical benches are used to isolate metrology components from vibrating optical elements. An Edmund Scientific 5 mW uniphase helium-neon cylindrical laser and an on-track photonics 10 mm dual axis position sensing detector (PSD) are mounted on one optical bench and a 2 in turning, flat mirror and a three-axis, fast steering mirror (FSM) are on the other optical bench to maximize the beam length. The path of the laser beam is drawn in Fig. 16. A laser shines a beam on the turning flat mirror, reflects off of the FSM, and shines on the PSD, which provides a measure of the acoustically induced jitter.

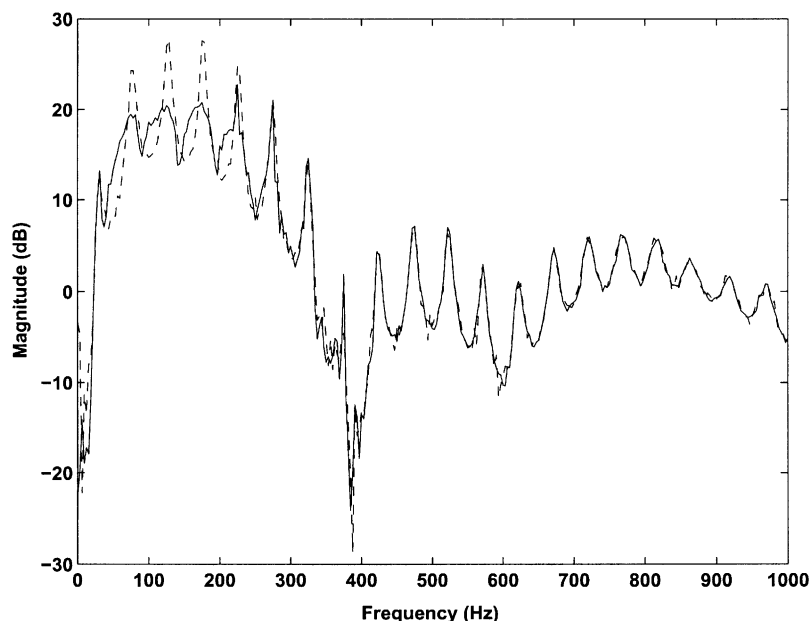


Fig. 15. Open-loop (dashed line) and closed-loop (solid line) FRF magnitude plot of the control error microphone signal for open-closed acoustic enclosure ($t_s = 500$ Hz).

An acoustic band-limited random disturbance applied to the loudspeaker(s)—part (f) in Fig. 16—is used to generate vibration in the optical components. The experiments are performed using a single disturbance loudspeaker or three disturbance loudspeakers to generate acoustic disturbance around the optical components in all directions. For multiple disturbance sources, each loudspeaker is generated by different random source and placed facing directly at the optical components—a turning flat mirror and a FSM—in order to observe maximum jitter effect.

In order to isolate the laser source and the PSD from the acoustic disturbance, these components are kept in a wood structure enclosure—part (i) in Fig. 16. Acoustic treatment is applied inside the enclosure and a small glass window is added for the laser beam.

Other required hardware such as a DSP system, amplifiers and filters are placed outside of the anechoic chamber. An Ithaco 24 dB/octave filter sets the cutoff frequency to 250 Hz, is used for anti-aliasing as well as a 10 \times amplification of the PSD signal. An additional Ithaco high-pass filter, set at 10 Hz, is used to remove the DC offset in the PSD signal to prevent the static position of the laser from saturating the ± 10 V A/D channels on the DSP board. The sampling rate of the experiments are set to be 600 Hz because the jitter effect occurs below 300 Hz [54,55].

In the RGPC algorithm, the order of the identified model and controller is chosen to be 16. The prediction horizon and the control horizon are chosen to be 32 and 4, respectively. The controller is updated at 1/4 the speed of the system identification process, i.e., 150 Hz.

Figs. 17 and 18 present the computed open- and closed-loop auto-spectrum of the vertical position signal from the PSD. Although the PSD provides the horizontal position of the beam as well, the acoustically induced jitter is hardly observed in the horizontal direction.

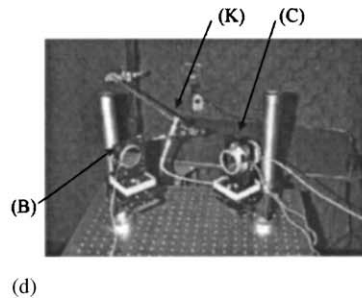
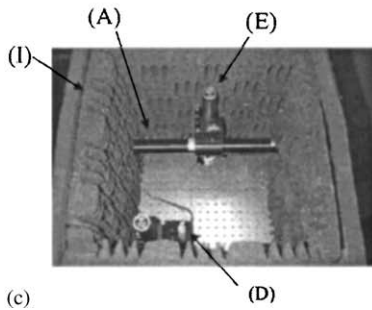
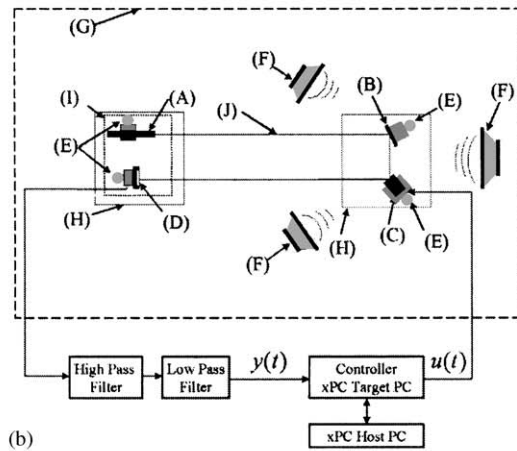
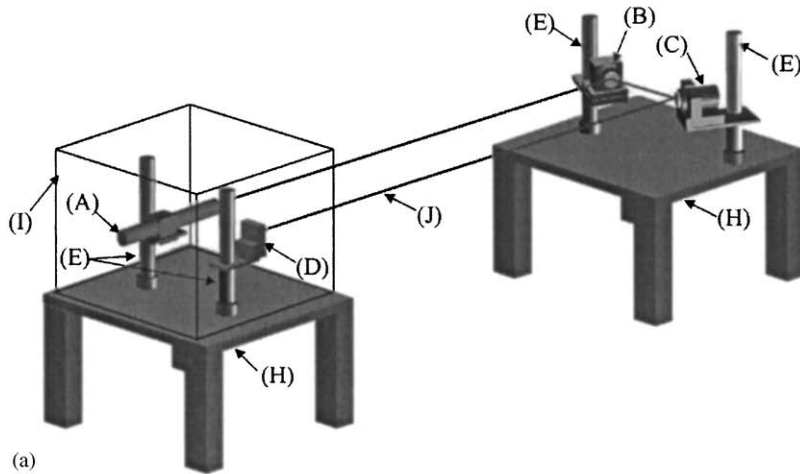


Fig. 16. Experimental system setup of an optical jitter suppression testbed: (a) schematic diagram of jitter testbed, (b) top view of schematic diagram, (c) picture of laser source and PSD, (d) picture of turning mirror and FSM. Experimental parts: (A) HeNe laser, (B) turning flat mirror, (c) fast steering mirror, (D) position sensing detector, (E) mounting rods, (F) disturbance speakers, (G) anechoic chamber, (H) optics benches, (I) enclosure, (J) laser beam.

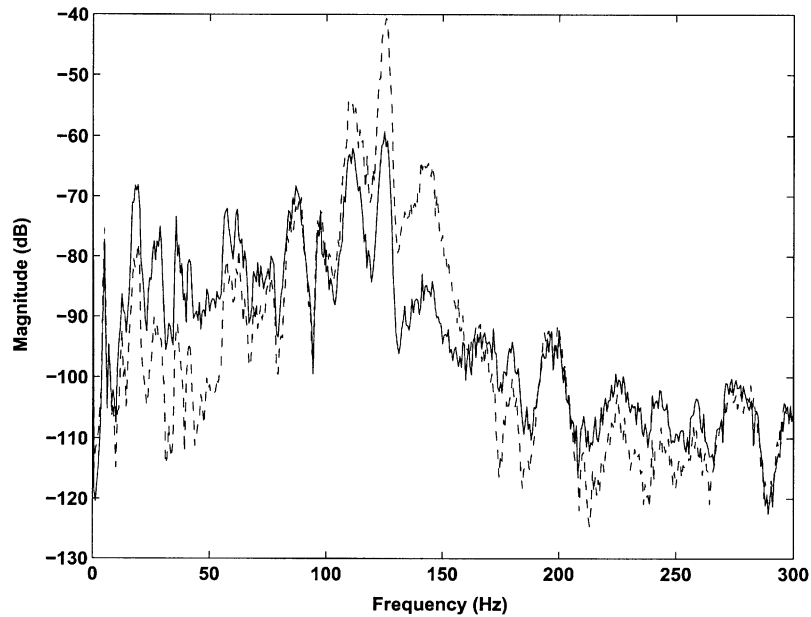


Fig. 17. Open-loop (dashed line) and closed-loop (solid line) auto-spectrum estimation with a single disturbance source.

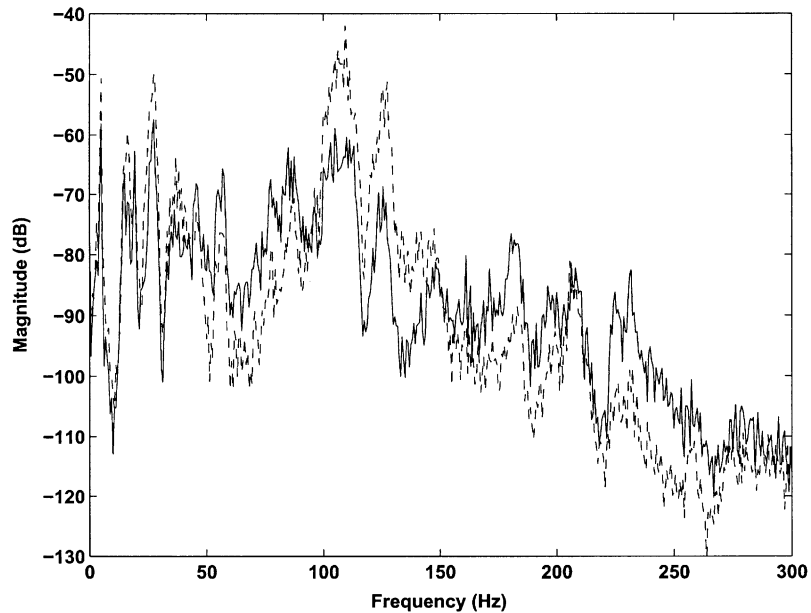


Fig. 18. Open-loop (dashed line) and closed-loop (solid line) auto-spectrum estimation with three disturbance sources.

Fig. 17 shows auto-spectrum of vertical position when a band-limited random disturbance is applied to a single disturbance loudspeaker, which is placed behind the optical bench. The dashed line is the open-loop response and the solid line is the closed-loop response. A reduction of 15.6 dB is obtained between 30 and 300 Hz.

Fig. 18 shows the auto-spectrum when three different random disturbances are applied to all three loudspeakers. In this case, a reduction of 11.9 dB is obtained between 30 and 300 Hz.

9. Conclusion

An adaptive control algorithm is widely used when there are variations in process dynamics and in the disturbances. In the viewpoint of using an adaptive control, the RGPC algorithm satisfies the requirements. The dynamics of the process is estimated in the process of the RLS system identification and a controller based on the GPC concepts is designed from the most up-to-date system model.

The RGPC algorithm combines the processes of system identification and controller design in a single process. As a result, the algorithm is efficient when it is implemented to a system because the application of the RGPC algorithm is much less time consuming than the alternative path of modelling, design, and implementation of a conventional control system.

The advantages of the algorithm are that no prior system information is required since models are estimated from real-time data, and that the controller is updated adaptively in the presence of a changing operating environment.

References

- [1] G. Favier, D. Dubois, A review of k-step-ahead predictors, *Automatica* 26 (1) (1990) 75–84.
- [2] C. Kambhampati, J.D. Mason, K. Warwick, A stable one-step-ahead predictive control of non-linear systems, *Automatica* 36 (4) (2000) 485–495.
- [3] G.C. Goodwin, K.S. Sin, *Adaptive Filtering Prediction and Control*, Prentice-Hall, Englewood Cliffs, NJ, 1984.
- [4] J.T. Bialasiewicz, L.G. Horta, M.Q. Phan, Identified predictive control, *Proceedings of the American Control Conference*, Vol. 3, 1994, pp. 3243–3247.
- [5] D. Jia, B.H. Krogh, Distributed model predictive control, *Proceedings of the American Control Conference*, Vol. 4, 2001, pp. 2767–2772.
- [6] J. Richalet, A. Rault, J.L. Testud, J. Papon, Model predictive heuristic control: applications to industrial processes, *Automatica* 14 (1978) 413–428.
- [7] C.R. Cutler, B.L. Ramaker, Dynamic matrix control: a computer control algorithm, in: *Proceedings of Joint Automatic Control Conference*, San Francisco, CA, 1980.
- [8] D.W. Clarke, C. Mohtad, P.S. Tuffs, Generalized predictive control—Part I. The basic algorithm, *Automatica* 23 (2) (1987) 137–148.
- [9] D.W. Clarke, C. Mohtad, P.S. Tuffs, Generalized predictive control—Part II. Extensions and interpretations, *Automatica* 23 (2) (1987) 149–160.
- [10] R.M.C. De Keyser, A.R. Van Cauwenberghe, Extended prediction self-adaptive control, *Proceedings of the IFAC Symposium on Identification and System Parameter Estimation*, 1975, pp. 929–934.
- [11] D.W. Clarke, C. Mohtadi, Properties of generalized predictive control, *Automatica* 25 (6) (1989) 859–875.
- [12] R.R. Bitmead, M. Gevers, V. Wertz, *Adaptive Optimal Control, The Thinking Man's GPC*, Prentice-Hall, Englewood Cliffs, NJ, 1990.
- [13] S. Skogestad, I. Postlewaite, *Multivariable Feedback Control*, Wiley, Chichester, England, 1996.
- [14] S.M. Kuo, D.R. Morgan, *Active Noise Control Systems: Algorithms and DSP Implementation*, Wiley, New York, 1996.
- [15] B. Shahian, M. Hassul, *Control System Design using MATLAB*, Prentice-Hall Inc., Englewood Cliffs, NJ, 1993.

- [16] H. Demircioglu, P.J. Gawthrop, Continuous-time generalized predictive control (CGPC), *Automatica* 27 (1) (1991) 55–74.
- [17] P.J. Gawthrop, A continuous-time approach to discrete-time self-tuning control, *Optimal Control Applications & Methods* 3 (4) (1982) 399–414.
- [18] K.J. Åström, P. Hagander, J. Sternby, Zeros of sampled systems, *Optimal Control Applications & Methods* 3 (4) (1982) 399–414.
- [19] D.W. Clarke, Self-tuning control of nonminimum-phase systems, *Automatica* 20 (5) (1984) 501–518.
- [20] N.K. Sinha, S. Puthenpura, Choice of the sampling interval for the identification of continuous-time systems from samples of input/output data, *Proceedings of the Institution of Electrical Engineers* 132 (6) (1985) 263–267.
- [21] B. Kouvaritakis, M. Cannon, J.A. Rossiter, Recent developments in generalized predictive control for continuous-time systems, *International Journal of Control* 72 (2) (1999) 164–173.
- [22] L. Chisci, E. Mosca, Stabilizing I/O receding horizon control of CARMA plants, *IEEE Transactions on Automatic Control* 39 (3) (1994) 614–618.
- [23] J.R. Gossner, B. Kouvaritakis, J.A. Rossiter, Cautious stable predictive control: a guaranteed stable predictive control algorithm with low input activity and good robustness, *International Journal of Control* 67 (5) (1997) 675–697.
- [24] J.R. Gossner, B. Kouvaritakis, J.A. Rossiter, Stable and generalized predictive control with constraints and bounded disturbances, *Automatica* 33 (4) (1997) 551–568.
- [25] D.W. Clarke, R. Scattolini, Constrained receding-horizon predictive control, *Proceedings of the Institution of Electrical Engineers* 138 (1991) 347–354.
- [26] B. Kouvaritakis, J.A. Rossiter, A.O.T. Chang, Stable generalized predictive control: an algorithm with guaranteed stability, *Proceedings of the Institution of Electrical Engineers* 139 (1992) 349–362.
- [27] E. Mosca, *Optimal, Predictive, and Adaptive Control*, Prentice-Hall, Englewood Cliffs, NJ, 1995.
- [28] B. Kouvaritakis, J.A. Rossiter, G.J. Ju, Robust stable generalized predictive control, *International Journal of Control* 67 (3) (1997) 411–434.
- [29] J.-N. Juang, M. Phan, Deadbeat predictive controllers, Technical Report TM-112862, NASA, May 1997.
- [30] M.Q. Phan, J.-N. Juang, Predictive controllers for feedback stabilization, *Journal of Guidance, Control, and Dynamics* 21 (5) (1998) 747–753.
- [31] M. Kinnaert, Adaptive generalized predictive controller for MIMO systems, *International Journal of Control* 50 (1) (1989) 161–172.
- [32] W. Wang, A direct adaptive generalized predictive control algorithm for MIMO systems, *International Journal of Control* 60 (6) (1994) 1371–1381.
- [33] B.E. Ydstie, Extended horizon adaptive control, in: J. Gertler, L. Keviczky (Eds.), *A Bridge Between Control Science and Technology*, Pergamon Press, Oxford, 1984, pp. 134–137.
- [34] K.S. Lee, W.-K. Lee, Extended discrete-time multivariable adaptive control using long-term predictor, *International Journal of Control* 38 (1983) 495–514.
- [35] U. Borison, Self-tuning regulators for a class of multivariable systems, *Automatica* 15 (1979) 209–215.
- [36] H.N. Koivo, A multivariable self-tuning controller, *Automatica* 16 (1980) 351–366.
- [37] L. Dugard, G.C. Goodwin, X. Xianya, The role of the interactor matrix in multivariable stochastic adaptive control, *Automatica* 20 (1984) 701–709.
- [38] S.L. Shah, C. Mohtadi, D.W. Clarke, Multivariable adaptive control without a prior knowledge of the delay matrix, *Systems and Control Letters* 9 (1987) 295–306.
- [39] J.M. Martin-Sanchez, S.L. Shah, Multivariable adaptive predictive control of a binary distillation column, *Automatica* 20 (5) (1984) 607–620.
- [40] J.-N. Juang, *Applied System Identification*, Prentice-Hall, Englewood Cliffs, NJ, 1994.
- [41] A.V. Oppenheim, R.W. Schaffer, J.R. Buck, *Discrete-Time Signal Processing*, 2nd Edition, Prentice-Hall, Englewood Cliffs, NJ, 1998.
- [42] B. Widrow, S.D. Stearns, *Adaptive Signal Processing*, Prentice-Hall, Englewood Cliffs, NJ, 1985.
- [43] A.V. Oppenheim, R.W. Schaffer, *Digital Signal Processing*, Prentice-Hall Inc., Englewood Cliffs, NJ, 1975.
- [44] R. Isermann, K.-H. Lachmann, D. Matko, *Adaptive Control Systems*, Prentice-Hall, Englewood Cliffs, NJ, 1992.
- [45] S. Haykin, *Adaptive Filter Theory*, 2nd Edition, Prentice-Hall, Englewood Cliffs, NJ, 1991.

- [46] L. Ljung, *System Identification: Theory for the User*, 2nd Edition, Prentice-Hall, Englewood Cliffs, NJ, 1999.
- [47] K.J. Åström, B. Wittenmark, *Adaptive Control*, 2nd Edition, Addison-Wesley, Reading, MA, 1995.
- [48] G.V. Moustakides, Study of the transient phase of the forgetting factor RLS, *IEEE Transactions on Signal Processing* 45 (10) (1997) 2468–2476.
- [49] J.-N. Junag, K.W. Eure, Predictive feedback and feedforward control for systems with unknown disturbance, Technical Report TM-208744, NASA, December 1998.
- [50] J.M. Maciejowski, *Multivariable Feedback Design*, Addison-Wesley, Reading, MA, 1989.
- [51] T. Kailath, *Linear Systems*, Prentice-Hall Inc., Englewood Cliffs, NJ, 1980.
- [52] R.L. Clark, G.P. Gibbs, W.R. Saunders, *Adaptive Structures, Dynamics and Control*, Wiley, New York, 1998.
- [53] L. Meirovitch, *Principles and Techniques of Vibrations*, Prentice-Hall, Englewood Cliffs, NJ, 1997.
- [54] R.M. Glaese, E.H. Anderson, P.C. Janzen, Active suppression of acoustically induced jitter for the airborne laser, *Proceedings of SPIE*, Vol. 4034, 2000, pp. 151–164.
- [55] W.B. DeShetler, J.D. Dillow, Suppression of noise in the airborne laser system, *Proceedings of SPIE*, Vol. 3706, 1999, pp. 249–257.