Data-Based Predictive Control with Multirate Prediction Step

Jonathan S. Barlow, *Member, AIAA, IEEE*

Abstract— **Data-based predictive control is an emerging control method that stems from Model Predictive Control (MPC). MPC computes current control action based on a prediction of the system output a number of time steps into the future and is generally derived from a known model of the system. Data-based predictive control has the advantage of deriving predictive models and controller gains from input-output data. Thus, a controller can be designed from the outputs of complex simulation code or a physical system where no explicit model exists. If the output data happens to be corrupted by periodic disturbances, the designed controller will also have the built-in ability to reject these disturbances without the need to know them. When data-based predictive control is implemented online, it becomes a version of adaptive control. One challenge of MPC is computational requirements increasing with prediction horizon length. This paper develops a closed-loop dynamic output feedback controller that minimizes a multi-step-ahead receding-horizon cost function with multirate prediction step. One result is a reduced influence of prediction horizon and the number of system outputs on the computational requirements of the controller. Another result is an emphasis on portions of the prediction window that are sampled more frequently. A third result is the ability to include more outputs in the feedback path than in the cost function.**

I. Introduction

DATA-Based Predictive Control is an emerging control method that stems from Model Predictive Control (MPC). MPC is the concept where the current control action is based on a prediction of the system output a number of time steps into the future [1]-[3]. A sequence of control actions (from the present time to some future time) is computed that minimizes a finite-duration cost function. Out of this sequence, only the present control input is applied to the system. At the next time step, the entire process repeats. Thus the starting and ending time steps of the cost function shift one time step forward. The term "receding-horizon" is often associated with this strategy.

In MPC, the system output is generally predicted using the state-space model based approach or the input-output model approach. These two approaches can be unified via an interaction matrix, which offers a convenient mapping from the state-space description to the input-output description [4], [5]. One important feature of the interaction matrix formulation is that, although the starting point of the MPC derivation is state-space based, in the end the controller has a dynamic output feedback form and can be implemented without an observer, and without explicit computation of the entire future output and input histories.

Using the interaction matrix formulation, MPC combines well with system identification to produce so-called "data-based" designs. Data-based designs are advantageous because they can be designed from the outputs of complex simulation code or a physical system where no explicit model exists [6]. Also, if the output data happens to be corrupted by periodic disturbances, then the designed controller will have the built-in ability to reject these disturbances without the need to know them.

One of the challenges of data-based predictive control, and MPC in general, is computational requirements increasing with prediction horizon length. For traditional MPC, the computational burden grows exponentially with horizon length. Reference [7] proposes a strategy for reducing computational burden of non-linear MPC by implementing a multi-rate open-loop control strategy, sampled in a non-equidistant way, where the shortest sampling interval is placed at the beginning of the horizon, and the following intervals are expanded exponentially with time.

For data-based predictive control, the computational burden grows only with the cube of the prediction horizon the number of inputs and outputs, and the order of the controller. However, this burden can still be large for large horizon lengths, large controller orders, and systems with many inputs and outputs. The computational burden can be further compounded when it is implemented online, and can limit the sampling rate.

While this contribution is inspired by computational requirements, it also enables more freedom in the control design by emphasizing portions of the prediction window that are sampled more frequently, and by allowing the inclusion of more outputs in the feedback path than in the cost function.

This approach derives a relationship between a multi-step-ahead receding-horizon cost function with a uniform prediction step and a multi-step-ahead receding-horizon cost function with multirate prediction step. The result is a closed-loop dynamic output feedback controller that minimizes the multi-step-ahead receding-horizon cost function with multirate prediction step.

II. State-space and Input-output Representations

Consider an r-input, m-output system with the system state $x(k)$ and output $y(k)$ given by

Manuscript received September 22, 2009. This work was supported by the NASA Fundamental Aeronautics Program, Subsonic Fixed-Wing Project.

J. S. Barlow is with the Stinger-Gaffarian Technologies, NASA Ames Research Center, Moffett Field, CA 94035-1000 USA phone: 650-604-3378; fax: 650-604-3594; e-mail: jonathan.s.barlow@nasa.gov.

$$x(k+1) = Ax(k) + Bu(k) + B_d u_d(k) \qquad (1)$$
$$y(k) = Cx(k) + Du(k) .$$

Neither the system model, defined by A, B, B_d, C, and D, nor the initial state of the system, $x(0)$, are assumed known, but a set of sufficiently rich and long excitation input $u(k)$ and possibly disturbance–corrupted output data $y(k)$ is available. The disturbance input, $u_d(k)$, if present, is assumed to be a sum of a finite number of unknown harmonics. Only an upper bound of the number of harmonics is known.

The representation of the data history can be simplified by the introduction of "super-vector" notation, defined by

$$g_w(k) \equiv \begin{bmatrix} g(k)^T & g(k+1)^T & \cdots & g(k+w-1)^T \end{bmatrix}^T \qquad (2)$$

where g will generally represent an output or control input (column) vector, w is the length of the vector.

For the system in (1), the output y(k) is dependent on the initial state $x(0)$ and the disturbance inputs $u_d(k)$. Since the disturbance input and the initial state are assumed unknown, it is beneficial to describe the system using a relationship between the excitation input and disturbance-corrupted output that does not explicitly include the terms involving the initial state and the disturbances. In [4], the interaction matrix formulation captures this input-output relationship, which does not depend on initial state and disturbances. It was shown that the following relationship holds for excitation input and possibly disturbance-corrupted output,

$$y_s(k+q) = P_1 u_p(k-p) - P_2 y_p(k-p) + W u_{s+q}(k) \qquad (3)$$

when p is selected such that $mp \geq n + 2f + 1$ and $0 \leq q \leq p$, where n is the system order, f is the number of distinct disturbance frequencies, and the 1 accounts for a constant disturbance if present. The parameters P_1, P_2, W are the coefficients of an s-step ahead predictor model. In this context, p is the number of past data points, q is the start of the prediction horizon, and s is the length of the prediction horizon. A conservative value for p can be chosen using an upper bound on the order of the system and the number of distinct disturbance frequencies.

III. Model Predictive Control Laws With Uniform Prediction Step

A predictive controller can be designed to minimize the receding-horizon cost function

$$J(k) = \begin{pmatrix} \left[y_s(k+q) - z_s(k+q)\right]^T Q \left[y_s(k+q) - z_s(k+q)\right] \\ + u_{s+q}^T(k) R u_{s+q}(k) \end{pmatrix}, \qquad (4)$$

where $z_s(k+q)$ is the desired output trajectory to be tracked. The output error cost is evaluated over the interval from time $k+q$ to $k+s+q-1$, with a weight matrix of Q. The control input cost is evaluated over the interval from time k to $k+s+q-1$, with a weight matrix of R.

The future control input history $u_{s+q}(k)$ that minimizes the resultant cost function can be found by substituting (3) into (4), and taking the derivative with respect to $u_{s+q}(k)$. The future control input history becomes

$$u_{s+q}(k) = \mathsf{A}_1 u_p(k-p) + \mathsf{A}_2 y_p(k-p) + \mathsf{B} z_s(k+q) \qquad (5)$$

$$\mathsf{A}_1 = -\mathsf{B}P_1 \text{ , } \mathsf{A}_2 = \mathsf{B}P_2 \text{ , } \mathsf{B} = (R + W^T Q W)^+ W^T Q . \qquad (6)$$

The optimal control law for the r control inputs is extracted from the first r rows of (5). It assumes a dynamic output feedback form shown in (7).

$$u(k) = Gu_p(k-p) + Hy_p(k-p) + Kz_s(k+q) \qquad (7)$$

The gains G, H, and K are the first r rows of A_1, A_2, and B, respectively.

IV. Model Predictive Control Laws With Multirate Prediction Step

Now consider the following cost function with a multirate prediction step,

$$J(k) = \begin{pmatrix} \left[\bar{y}(k+q) - \bar{z}(k+q)\right]^T \bar{Q} \left[\bar{y}(k+q) - \bar{z}(k+q)\right] \\ + u_{s+q}^T(k) R u_{s+q}(k) \end{pmatrix} \qquad (8)$$

where $\bar{y}(k+q)$ and $\bar{z}(k+q)$ are L vectors of future outputs and desired outputs at arbitrary time steps $e = \begin{bmatrix} e_1 & e_2 & e_3 & \cdots & e_L \end{bmatrix}$, as in (9), and $\bar{Q}$ is the weight matrix of the multirate output error.

$$\bar{y}(k+q) = \begin{bmatrix} y(v_1, k+q+e_1) \\ y(v_2, k+q+e_2) \\ y(v_3, k+q+e_3) \\ \vdots \\ y(v_L, k+q+e_L) \end{bmatrix}, \bar{z}(k+q) = \begin{bmatrix} z(v_1, k+q+e_1) \\ z(v_2, k+q+e_2) \\ z(v_3, k+q+e_3) \\ \vdots \\ z(v_L, k+q+e_L) \end{bmatrix} (9)$$

The outputs $y(v_i, k+q+e_i)$ and desired outputs $z(v_i, k+q+e_i)$ are arbitrary subsets of the system outputs and desired outputs. The indices $v_i = \begin{bmatrix} v_{i,1} & v_{i,2} & \cdots & v_{i,N_i} \end{bmatrix}, v_{i,j} \leq m$ denote the outputs used at each time e_i. Using this notation, each output can be assigned a distinct sampling rate, or be excluded from the cost function entirely, e.g. a vector $\bar{y}(k+q)$ with a first output sampled every other time step, a second output sampled every third time step, and a third output omitted from the cost function can be expressed as outputs at time steps $\left[e_1, e_2, e_3, e_4, e_5\right] = \left[1,3,4,5,7\right]$ and the outputs used

$\left[v_1, v_2, v_3, v_4, v_5\right] = \left[\begin{bmatrix}1\\2\end{bmatrix}, 1, 2, 1, \begin{bmatrix}1\\2\end{bmatrix}\right]$. The selection of e_i and v_i can be selected by the control designer to tune the distribution of weights in the cost function. The selection of e_i can be used to tune the weighting matrices to weight different portions of the prediction window with different relative amounts. Portions that are sampled more frequently have a higher relative weighting of the error than portions that are weighted less frequently.

Equation (3) was derived with a uniform step size, and cannot be directly applied. However, the cost function (8) can be transformed to an equivalent form by defining a selector matrix E, which is formed from the rows $\begin{bmatrix} v_1 + me_1 & v_2 + me_2 & \cdots & v_L + me_L \end{bmatrix}$ of an $s\,m$ by sm identity matrix. E relates $\bar{y}(k+q)$ to the uniformly sampled future outputs $y_s(k+q)$ by

$$\bar{y}(k+q) = Ey_s(k+q). \tag{10}$$

Equation (10) assumes $0 \le e_i < s$, but since s is a design parameter, this is not restrictive. The future control input history $u_{s+q}(k)$ that minimizes (8) is then found by substituting (3) into (10), substituting (10) into (8), and taking the derivative with respect to $u_{s+q}(k)$. The future control input history becomes

$$u_{s+q}(k) = \overline{\mathsf{A}_1}u_p(k-p) + \overline{\mathsf{A}_2}y_p(k-p) + \bar{\mathsf{B}}\bar{z}(k+q) \tag{11}$$

$$\overline{\mathsf{A}_1} = -\bar{\mathsf{B}}EP_1, \overline{\mathsf{A}_2} = \bar{\mathsf{B}}EP_2, \bar{\mathsf{B}} = (R + W^T E^T \overline{Q} EW)^+ W^T E^T \overline{Q}. \tag{12}$$

The counterpart to (7) is then

$$u(k) = \overline{G}u_p(k-p) + \overline{H}y_p(k-p) + \overline{K}\bar{z}(k+q). \tag{13}$$

The gains $\overline{G}$, $\overline{H}$, and $\overline{K}$, are the first r rows of $\overline{\mathsf{A}_1}$, $\overline{\mathsf{A}_2}$, and $\bar{\mathsf{B}}$, respectively.

V. Adaptive Data-Based Controller Design

To employ the optimal control law in (13), the controller gains $\overline{G}$, $\overline{H}$, and $\overline{K}$, must be either known *a priori* or estimated online. The controller gains can be designed a priori for the model predictive approach by relying on a model of the system. Similarly, a data-based design can be done *a priori* in a two step approach, where the model P_1, P_2, W is identified from input-output data and the gains $\overline{G}$, $\overline{H}$, and $\overline{K}$, are then designed. For an online, adaptive implementation of the control laws, the gains are designed directly from input-output data via a relationship that relates $\overline{G}$, $\overline{H}$, and $\overline{K}$, to input-output data. The equation that enables the direct relationship for the uniform prediction step was derived in [6] and is shown in (14).

$$\begin{aligned}\left[I - \left(R + W^T QW\right)^{-1} R\right]u_{s+q}(k) = \mathsf{A}_1 u_p(k-p) \\ + \mathsf{A}_2 y_p(k-p) + \mathsf{B}y_s(k+q)\end{aligned} \tag{14}$$

In (14) Q and R are the weighting matrices of the cost function with uniform prediction step, and W is the coefficient matrix of the s-step ahead predictor model in (3). Further detail about the derivation of this equation can be found in [6]. The corresponding equation for the multirate prediction step can be shown to be

$$\begin{aligned}\left[I - \left(R + W^T E^T \overline{Q} EW\right)^{-1} R\right]u_{s+q}(k) = \overline{\mathsf{A}}_1 u_p(k-p) \\ + \overline{\mathsf{A}}_2 y_p(k-p) + \bar{\mathsf{B}}\bar{y}(k+q)\end{aligned} \tag{15}$$

The first r rows of (15) are extracted to produce the input-output relationship shown in (16).

$$\bar{S}u_{s+q}(k) = \overline{G}u_p(k-p) + \overline{H}y_p(k-p) + \overline{K}\bar{y}(k+q) \tag{16}$$

$\bar{S}$ is the first r rows of $\left[I - \left(R + W^T E^T \overline{Q} EW\right)^{-1} R\right]$. Equation (16) has the property of being an open-loop input-output equation with the controller gains $\overline{G}$, $\overline{H}$, and $\overline{K}$, from (13) included explicitly as coefficients of the equation. Using (16), the coefficients of the open-loop input-output model can be identified, and used in (13) as the gains of a dynamic feedback controller.

The data-based predictive controller developed in the simulation updates $\overline{G}$, $\overline{H}$, and $\overline{K}$, online using (16) and past input and output data. Since (16) is a non-causal input-output relationship, the approach begins with a time shift of $-(s+q)$ to the data sets within (16) in order to fully populate the super-vectors of collected data, with the most recent data used being $y(k-1)$ and $u(k-1)$. The time-shifted equation is then

$$\begin{aligned}\bar{S}u_{s+q}(k-s-q) = \overline{G}u_p(k-p-s-q) \\ + \overline{H}y_p(k-p-s-q) + \overline{K}\bar{y}(k-s).\end{aligned} \tag{17}$$

Equation (17) is then arranged in the form

$$\bar{S}u_{s+q}(k-s-q) = \gamma(k)\Gamma(k-1) \tag{18}$$

$$\gamma(k) = \left[\overline{G}(k) \;\; \overline{H}(k) \;\; \overline{K}(k)\right], \Gamma(k-1) = \begin{bmatrix} u_p(k-p-s-q) \\ y_p(k-p-s-q) \\ \bar{y}(k-s) \end{bmatrix}. \tag{19}$$

In general, any linear estimation algorithm may be used to identify the parameters in $\gamma(k)$. For this application a recursive least-squares [8] estimation of the form

$$\gamma(k) = \gamma(k-1) + \left\{\bar{S}u_{s+q}(k-q-s) - \gamma(k-1)\Gamma(k-1)\right\}\Phi(k) \tag{20}$$

$$\Phi(k)=\frac{\Gamma(k-1)^T\Theta(k-1)^T}{1+\Gamma(k-1)^T\Theta(k-1)\Gamma(k-1)} \tag{21}$$

$$\Theta(k)=\Theta(k-1)-\Theta(k-1)\Gamma(k-1)\Phi(k) \tag{22}$$

is used to update $\overline{G}(k)$, $\overline{H}(k)$, and $\overline{K}(k)$, in $\gamma(k)$, starting with some initially large covariance matrix $\Theta(0)$ and an initial guess of the controller gains $\gamma(0)$ and $\overline{S}$. In practice, the control error $\left\{\overline{S}u_{s+q}(k-q-s)-\gamma(k-1)\Gamma(k-1)\right\}$ found in (20) is subject to a dead-band. The estimation of $\gamma(k)$ is conducted every time step. The optimal control input is then

$$u(k)=\overline{G}(k)u_p(k-p)+\overline{H}(k)y_p(k-p)+\overline{K}(k)\overline{z}(k+q). \tag{23}$$

VI. Simulation Results

The 5-degree-of-freedom system shown in Fig. 1 is used to illustrate the control design method. The set up allows various combinations of inputs, outputs, and disturbance locations for illustration. The model mass matrix is a diagonal matrix with m1, m2, … m5 on the main diagonal, and the damping, and stiffness matrices are

$$C=\begin{bmatrix} c_1+c_2 & -c_2 & 0 & 0 & 0 \\ -c_2 & c_2+c_3 & -c_3 & 0 & 0 \\ 0 & -c_3 & c_3+c_4 & -c_4 & 0 \\ 0 & 0 & -c_4 & c_4+c_5 & -c_5 \\ 0 & 0 & 0 & -c_5 & c_5 \end{bmatrix}$$
$$K=\begin{bmatrix} k_1+k_2 & -k_2 & 0 & 0 & 0 \\ -k_2 & k_2+k_3 & -k_3 & 0 & 0 \\ 0 & -k_3 & k_3+k_4 & -k_4 & 0 \\ 0 & 0 & -k_4 & k_4+k_5 & -k_5 \\ 0 & 0 & 0 & -k_5 & k_5 \end{bmatrix} \tag{24}$$

where m_i=1.5, k_i=5000, and c_i=10, in consistent units. A discrete-time model was generated from the continuous model using a sampling interval of 0.01 second. The examples illustrate various multi-input multi-output controller designs with uniform and multirate cost functions. The two inputs are the forces acting on masses 1 and 4 and the two collocated outputs are positions of the same masses 1 and 4. The system has no direct transmission term, thus the smallest value for q that can be selected is 1, which is used here. In these examples, the disturbance input acts on mass 3, and is unknown to the controller. The disturbance is a sum of 5 harmonics at 2 Hz, 8 Hz, 12 Hz, 15 Hz, and 17 Hz. A typical disturbance input time history is shown in Fig. 2. The mass 1 is to track a sinusoid with frequency 0.159 Hz for and mass 2 is to track a sinusoids with frequency 0.398. For all examples, the tracking gains $\overline{G}(k)$, $\overline{H}(k)$, and $\overline{K}(k)$, are computed from (19)-(22). The selected weighting matrices are $Q=10^4 I$, $R=I$. Larger Q relative to R allows for faster tracking with better accuracy at the expense of larger initial control effort. A large initial control effort during convergence of the recursive algorithm can result in temporary instability, which the controller must then overcome.

A. Baseline Examples

1) Example 1: Uniform prediction step with long prediction horizon

This example illustrates the case of a tracking controller with a uniform prediction step using the recursive least-squares solution in the presence of disturbances. The order of the system is 10 (n = 10), there are 5 disturbance frequencies (f = 5) and no constant disturbance, and the system has 2 outputs (m = 2), therefore the minimum order of the controller is p = 10 in order to satisfy the requirement $mp \geq n+2f$. In a practical application one may not know the order of the system and the number of disturbance frequencies, but only reasonable estimates of their upper bounds. In that case a much higher value of p should be used. In this example we select p = 50 as such a “safe” value. Next we select the duration of the prediction horizon in the cost function s. Typically we select $s \geq p$, as larger s tends to enhance closed-loop stability. Here we select s = 50. A larger value for s results in an equal emphasis on the short-term and long-term tracking error. All other control parameters are kept at their previous values. The order of the controller is p = 50. Figure 3 shows the performance of the resulting controller when the controller is turned on after 3 seconds. Note that the controller initial inputs are of the same magnitude as the inputs needed to track the desired outputs, and the controlled outputs take some time to converge to the desired outputs.

2) Example 2: Uniform prediction step with short prediction horizon

This example illustrates the case of a tracking controller with a uniform prediction step and a shorter prediction length. Computational requirements and design considerations may influence the selection of s, and a smaller s may be chosen. Here we select s = 10. A smaller value for s results in an emphasis on the short term tracking error and ignores long-term errors. Figure 4 shows the performance of the resulting controller when the controller is turned on after 3 seconds. The controller causes the controlled outputs to track the prescribed output trajectories while simultaneously rejecting the disturbances. Note the controller initial inputs are very large, and the outputs converge quickly to the desired outputs. Decreasing the length of the prediction horizon results in a 38% shorter simulation time.

B. Multirate prediction step for emphasizing portions of the prediction window.

1) Example 3: multirate prediction step with long prediction horizon.

This example illustrates the reduced computational cost of a controller with a multirate prediction step and a longer prediction length. Here we select s = 50, and $e=[1,2,\ldots,10,11,13,\ldots,19,21,26,\ldots,46]$. In this example, e

was selected to weight the first 10 time steps heavily, the next 10 time steps moderately, and the last 30 time steps lightly. All other control parameters are kept at their previous values. Figure 5 shows the performance of the resulting controller when the controller is turned on after 3 seconds. Note that the controller initial inputs are of the same magnitude as the inputs needed to track the desired outputs as in Example 1, but the controlled outputs converge more quickly to the desired outputs. Using the Multirate cost function results in a 24% shorter simulation time.

C. Multirate prediction step for including more outputs in the feedback path than in the cost function.

1) Example 4: multirate prediction step tracking only one output.

This example illustrates the decreased computational cost of a controller tracking only one output with a multirate prediction step and a long prediction length. Here we select the sampling rate of the first output to be every time step, and we exclude the second output from the cost function only, i.e. $e_i = i$, $v_i = 1$. Because there are still two outputs from the system, m = 2, the value for p = 50 is kept as in example 1. All other control parameters are kept at their previous values. Figure 7 shows the performance of the resulting controller when the controller is turned on after 3 seconds. Note that the controller initial inputs are of the same magnitude as the inputs needed to track the desired outputs and the output 1 converges to the desired output as quickly as the same output in example 2. Note that output 2 does not have a desired trajectory to be tracked and is uncontrolled. Output 2 is used only a feedback output for the controller. Because the same value of p is used, and only one output is tracked, the simulation time decreases by about 1%.

Table I summarizes the results of all four simulations.

TABLE I
SUMMARY OF SIMULATION RESULTS

Control Technique	Control aggressiveness	Computation Time (%)
uniform prediction step, long prediction length.	low	100%
uniform prediction step, short prediction length.	high	62%
multirate prediction step, long prediction length.	low	76%
multirate prediction step, long prediction length, one output used in cost function.	low	99%

VII. CONCLUSION

A multirate data-based predictive control has been presented which reduces the computational cost of increasing the prediction horizon length s and/or the number of outputs m. The multirate data-based predictive control laws were derived using a multi-rate cost function and the resulting control laws are in dynamic output feedback form.

In general, the computation of the gains G, H, K, whether done in batch or online is proportional to the cube of the sum of the lengths of $u_p(k-p)$, $y_p(k-p)$, and $y_s(k+q)$, i.e. $(rp+mp+ms)^3$. Larger values of m, p, and s are desirable for enhanced stability and tracking performance, but incur additional computational costs. Computation for the multirate controller design is proportional to the length of $\bar{y}(k+q)$ instead of $y_s(k+q)$ which does not increase with s and m.

This contribution can be applied to data-based predictive control and model predictive control. Data-based predictive control has been successfully employed for a number of applications, including linear time-invariant systems and multiple-vehicle formations, [9], [10]. This contribution is especially applicable to adaptive data-based predictive control, since the computational cost is incurred every time step, and large computations can limit the bandwidth of the controller. The adaptive version of data-based predictive control is of particular interest for its potential application to linear time-varying systems, such as for the control of diffusion dependent chemical processes, control of aircraft and in particular the flight and propulsion control of a Short Take-off and Landing (STOL) aircraft [11].

REFERENCES

[1] Richalet, J., Rault, A., Testud, J. L., and Papon, J., "Model Predictive Heuristic Control: Applications to Industrial Processes," Automatica, Vol. 14, No. 5, 1978, pp. 413-428.

[2] Clarke, D. W., Mohtadi, C., and Tuffs, P. S., "Generalized Predictive Control - Parts I and II," Automatica, Vol. 23, No. 2, 1987, pp. 137-148 and pp. 149-160.

[3] Clarke, D. W., and Mohtadi, C., "Properties of Generalized Predictive Control," Automatica, Vol. 25, No. 6, 1989, pp. 859-875.

[4] Darling, R. S. and Phan, M. Q., "Dynamic Output Feedback Predictive Controllers for Vibration Suppression and Periodic Disturbance Rejection," Proceedings of the 14th AAS/AIAA Space Flight Mechanics Meeting, Maui, HI, February 2004.

[5] Darling, R. S. and Phan, M. Q., "Predictive Controllers for Simultaneous Tracking and Disturbance Rejection," Proceedings of the AIAA Conference on Guidance, Navigation, and Control, Providence, RI, August 2004.

[6] Barlow, J. S. and Phan, M. Q., Direct Data-Based Model Predictive Controller Design, Proceedings of the AIAA Conference on Guidance, Navigation, and Control, Keystone, CO, August 2006.

[7] Halldorsson, U., Fikar, M., and Unbehauen, H., Control Thoery and Application Nonlinear predictive control with multirate optimisation step lengths, Control Theory and Applications, IEE Proceedings, Vol. 152, No. 3, May 2005, pp. 273-284

[8] Goodwin, G. C. and Sin, K. S., Adaptive Filtering Prediction and Control, Prentice-Hall, Englewood Cliffs, NJ, 1984.

[9] Barlow, J. S. and Phan, M. Q., "Model Predictive Control of Vehicle Formations," Proceedings of the AAS/AIAA Astrodynamics Specialist Conference, Mackinac Island, MI, August 2007.

[10] 7 Barlow, J. S. and Phan, M. Q., "Adaptive Predictive Control of Vehicle Formations with Obstacle Avoidance," Proceedings of the AAS/AIAA Astrodynamics Specialist Conference, Mackinac Island, MI, August 2007.

[11] Barlow, J. S., Acosta, D. M., and Phan, M. Q., Adaptive Data-based Predictive Control for Short Take-off and Landing (STOL) Aircraft AIAA Infotech@Aerospace Conference, Seattle, WA, 2009.

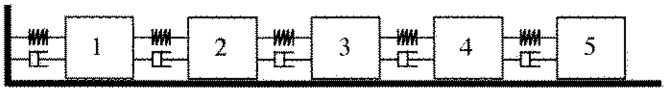

Fig. 1. A Spring-Mass-Damper Model.

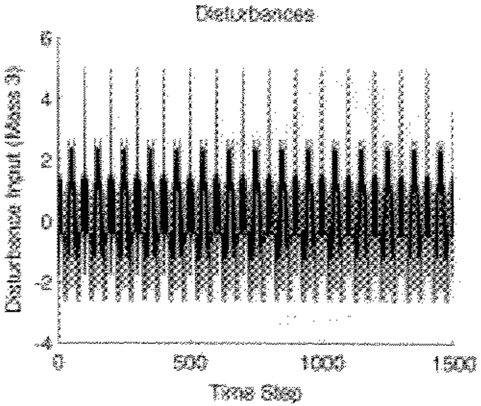

Fig. 2. "Unknown" disturbance input.

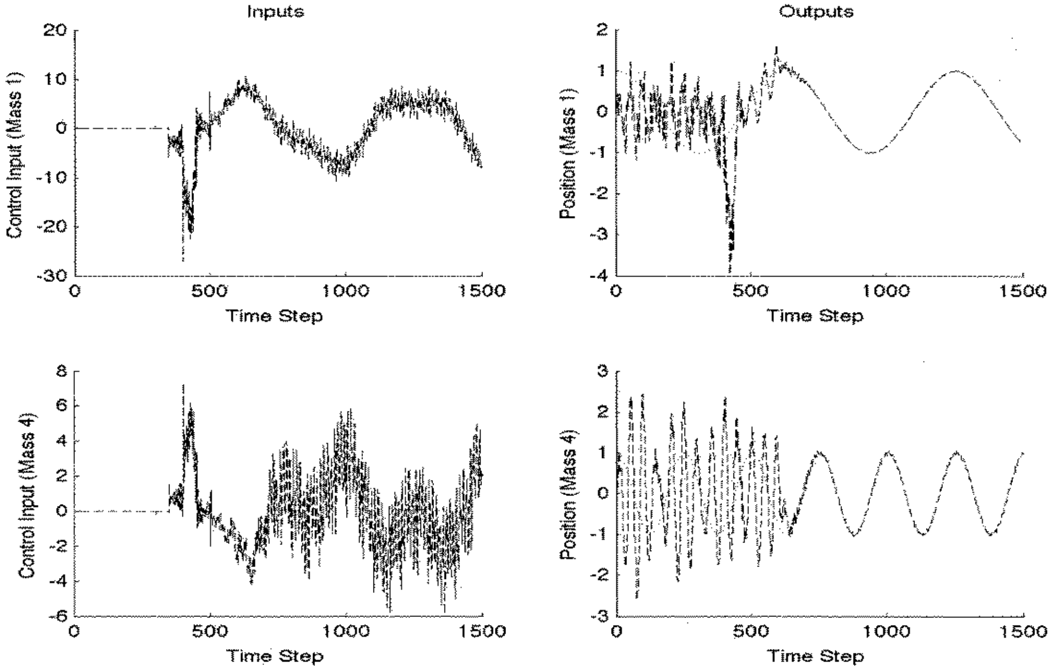

Fig. 3. An adaptive data-based predictive controller tracks two outputs using a long prediction horizon and uniform prediction step.

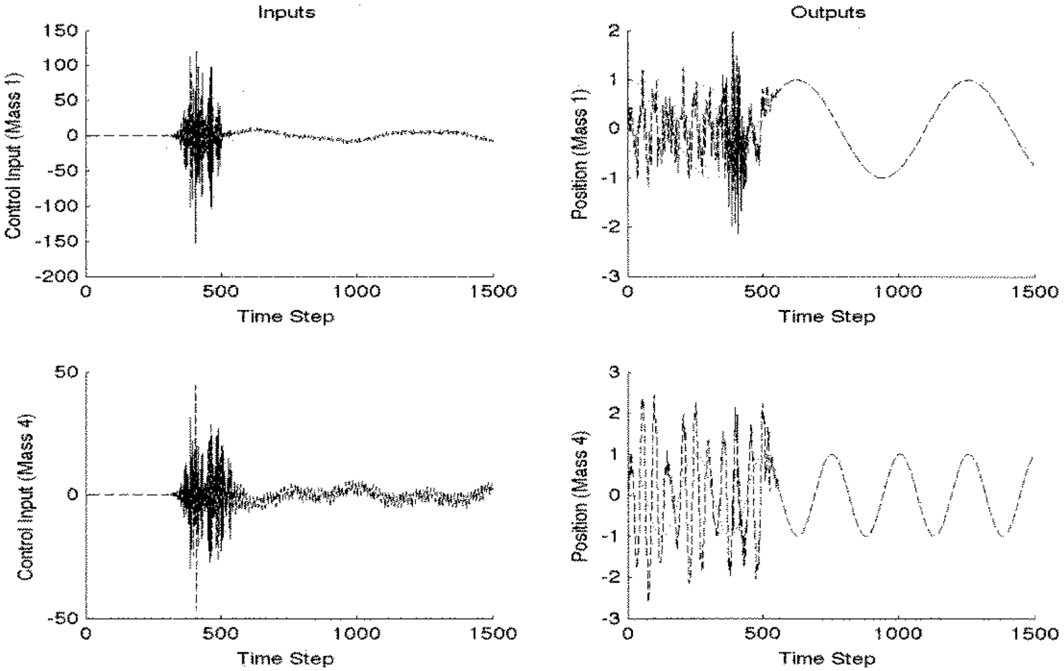

Fig. 4. An adaptive data-based predictive controller tracks two outputs using a short horizon length and uniform prediction step.

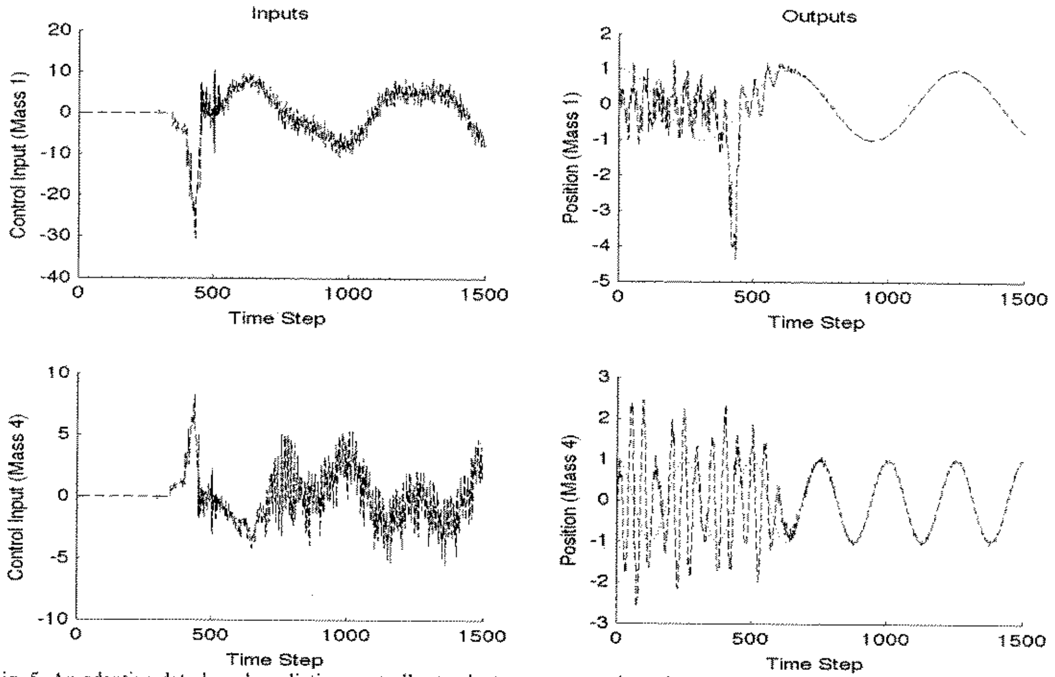

Fig. 5. An adaptive data-based predictive controller tracks two outputs using a long prediction horizon and multirate prediction step.

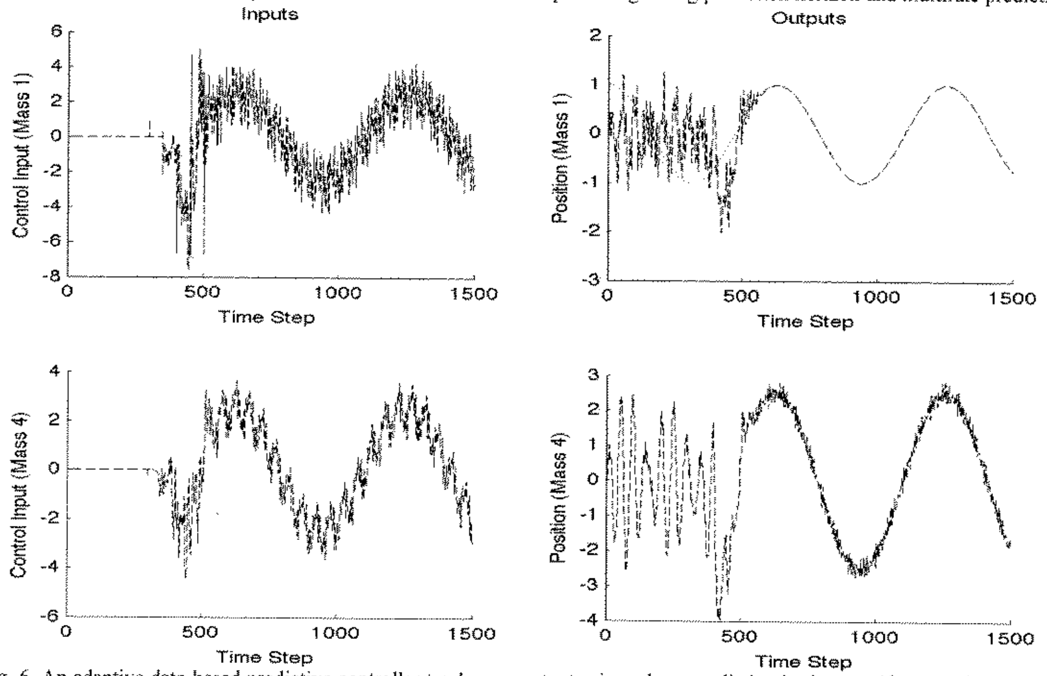

Fig. 6. An adaptive data-based predictive controller tracks one output using a long prediction horizon, uniform prediction step, and one output.