NASA Contractor Report 187561

IN-33
27175

PROGRESS IN MULTIRATE DIGITAL CONTROL SYSTEM DESIGN

Martin C. Berg and Gregory S. Mason

UNIVERSITY OF WASHINGTON
Seattle, Washington

Grant NAG1-1055
June 1991

National Aeronautics and
Space Administration

Langley Research Center
Hampton, Virginia 23665-5225

FINAL REPORT

NASA Langley Research Grant NAG-1-1055

Progress in Multirate Digital Control System Design

1 September 1989 - 31 December 1990

Martin C. Berg

Assistant Professor

Mechanical Engineering Department, FU-10

University of Washington

Seattle, Washington 98195

Gregory S. Mason

Graduate Student

Mechanical Engineering Department, FU-10

University of Washington

Seattle, Washington 98195

ABSTRACT

A new methodology for multirate sampled-data control system design based on (1) a new generalized control law structure, (2) two new parameter-optimization-based control law synthesis methods, and (3) a new singular-value-based robustness analysis method is described. The control law structure can represent multirate sampled-data control laws of arbitrary structure and dynamic order, with arbitrarily prescribed sampling rates for all sensors and update rates for all processor states and actuators. The two control law synthesis methods employ numerical optimization to determine values for the control law parameters to minimize a quadratic cost function, possibly subject to constraints on those parameters. The robustness analysis method is based on the multivariable Nyquist criterion applied to the loop transfer function for the sampling period equal to the period of repetition of the system's complete sampling/update schedule. The complete methodology is demonstrated by application to the design of a combination yaw damper and modal suppression system for a commercial aircraft.

CONTENTS

I. INTRODUCTION

The original objective for this project was to demonstrate a new algorithm for synthesizing multirate sampled-data control laws by application to a representative aircraft control problem. That algorithm, developed in connection with another research effort supervised by the Principal Investigator and based on a finite-time quadratic cost function, eventually proved unsuitable for the aircraft control problem. To complete this project we therefore developed a new multirate control law synthesis algorithm, based on an infinite-time quadratic cost function, along with a new method for analyzing the robustness of multirate systems, and applied both to the aircraft control problem

The following is a complete list of the contributions of this project:

1. A new generalized multirate sampled-data control law structure (GMCLS) was introduced. Features of this structure include an arbitrary dynamic order and structure for the processor dynamics; and sampling rates for all sensors, update rates for all processor states, and update rates for all actuators that can be selected independently. (discussed in Section II)

2. A new infinite-time-based parameter optimization multirate sampled-data control law synthesis method and solution algorithm were developed. (discussed in Section III)

3. A new singular-value-based method for determining gain and phase margins for multirate systems was developed. (discussed in Section IV)

4. The finite-time-based parameter optimization multirate sampled-data control law synthesis algorithm originally intended to be applied to the aircraft problem in this project, was instead demonstrated by application to a simpler problem involving the control of the tip position of a two-link robot arm. (discussed in Sections III and V)

5. The GMCLS, the new infinite-time-based parameter optimization multirate control law synthesis method and solution algorithm, and the new singular-value based method for determining gain and phase margins were all demonstrated by application to the aircraft control problem originally proposed for this project. (discussed in Section VI)

These five contributions are discussed in order in the following sections of this report. The first three sections are in a summary form only and the reader is referred, for details, to preprints of journal papers in the appendixes. The next two sections present applications of the parameter optimization techniques. The final two sections present our conclusions and suggest topics for future research.

II. The Generalized Multirate Sampled-Data Control Law Structure

A key point often ignored by the developers of multirate sampled-data control law synthesis methods is that, in order for any such method to be practically useful, it must provide the control law designer with the flexibility to independently choose the sampling rate for every sensor, the update rate for every processor state, and the update rate for every actuator. Such flexibility is frequently essential for efficient utilization of real-time control hardware, and for systems that include distributed processing and/or utilize sensors that provide only discrete-time signals at fixed sampling rates [1]. In this section we present a general-purpose, multirate sampled-data control law structure (GMCLS) that provides that flexibility.

To understand the GMCLS, it is necessary to establish a certain notation regarding the scheduling of sampling and update activities for a multirate system. Figure 1 shows an example of the time lines for the sampling and update activities of a multirate system. We define the *shortest time period* (STP) as the greatest common divisor of all of the sampling, update and delay periods; and we define the *basic time period* (BTP) as the least common multiple of all of the sampling, update and delay periods. We reserve the symbol T to represent the STP, and the symbol P to represent the (integer) number of STP's per BTP. Finally, we frequently make use of a doubly-indexed independent (time) variable, so that, for example, $x(m,n)$ represents x at the start of the $(n+1)th$ STP of the $(m+1)th$ BTP, for m=0,1, ... and n=0,1, ... ,P-1.

A block diagram of the GMCLS is shown in Figure 2. $\tilde{y}$ represents the incoming, noise-free, continuous-time sensor signal; v is the discrete-time sensor noise signal; and $\tilde{u}$ is the continuous-time control signal. The sampling period of the one sampler is the STP of the complete system's sampling/update schedule. The delay blocks are one-STP delays; and the ZOH block represents a zero-order hold.

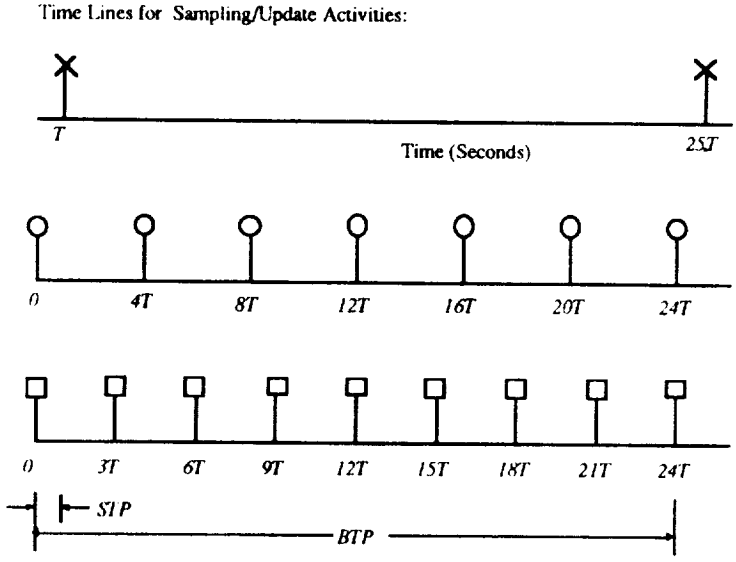


Figure 1 Example Multirate Sampling/Update Schedule

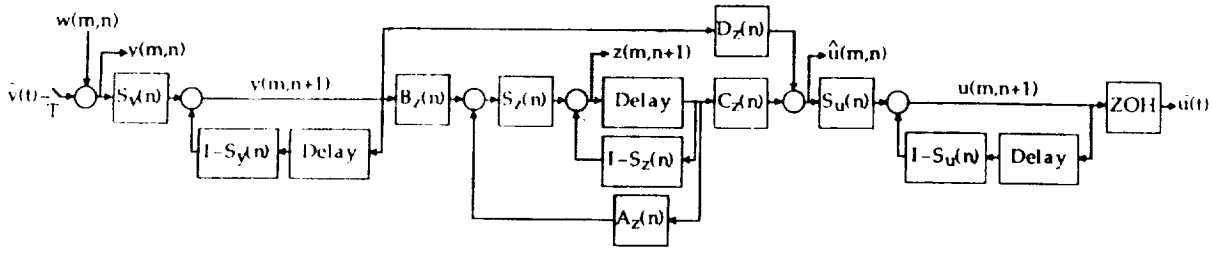


Figure 2 Generalized Multirate Sampled-Data Control Law Structure

A key feature of the GMCLS is its use of the switching matrices, $S_y(n)$, $S_z(n)$, and $S_u(n)$, for n=0,1, ... ,P-1, to represent the variations in the sensor sampling, processor state update, and control update activities, respectively. We define a switching matrix as a binary, diagonal matrix. $S_y(n)$ is the switching matrix that describes the sensor sampling activities at the start of the $(n+1)$*th* STP (of every BTP). If the *ith* diagonal element of $S_y(n)$ is 1, then the *ith* sensor's signal is sampled at the start of the $(n+1)$*th* STP of every BTP, and the sampled value, with the sensor noise v added, is immediately stored as the *ith* element of $\bar{y}$. If the *ith* diagonal element of $S_y(n)$ is 0, then the same element of $\bar{y}$ is simply held at those instants. The update activities for the processor state vector z and for the actuator hold state vector $\bar{u}$, in Figure 1, are similarly represented by the switching matrices $S_z(n)$, and $S_u(n)$, respectively, for n=0,1, ... ,P-1.

For a detailed discussion of the GMCLS see [1]. The key points are:

1. The switching matrices $S_y(n)$, $S_z(n)$, and $S_u(n)$ are completely determined by the system's sampling and update activities schedule.

2. The only unknowns are the processor matrices $A_z(n)$, $B_z(n)$, $C_z(n)$, and $D_z(n)$

3. The dynamic order of the processor dynamics (i.e., the dimension of z) is arbitrary.

For design purposes, the implications of these points are the following:

1. The GMCLS provides complete flexibility with regard to the selection of sampling rates for all sensors, update rates for all processor states, and update rates for all actuators. The single constraint is that the ratio of all sampling, update and delay rates must be rational, so that the complete sampling/update schedule is periodic.

2. The GMCLS provides complete flexibility with regard to the dynamic order and structure of the control law; i.e., the input-output dynamics of virtually any multirate sampled-data control law of practical interest can be realized with the GMCLS.

3. Apart from the (significant) problem of choosing sampling and update rates, the GMCLS reduces the control law synthesis problem to one of determining the processor matrices $A_z(n)$, $B_z(n)$, $C_z(n)$, and $D_z(n)$, for n=0,1, ... ,P-1.

For the purpose of numerically determining $A_z(n)$, $B_z(n)$, $C_z(n)$, and $D_z(n)$ it is convenient to represent the GMCLS in the following state model form (see [1] for details):

$$c(m,n+1) = A_c(n)c(m,n) + B_c(n)y(m,n) \tag{1}$$

$$u(m,n) = C_c(n)c(m,n) + D_c(n)y(m,n) \tag{2}$$

where

$$c(m,n) = [z(m,n)\ \ \bar{y}(m,n)\ \ \bar{u}(m,n)]^T \tag{3}$$

$$A_c(n) = \begin{bmatrix} [I-S_z(n)]+S_z(n)A_z(n) & S_z(n)B_z(n)[I-S_y(n)] & 0 \\ 0 & I-S_y(n) & 0 \\ S_u(n)C_z(n) & S_u(n)D_z(n)[I-S_y(n)] & I-S_u(n) \end{bmatrix} \tag{4}$$

$$B_c(n) = \begin{bmatrix} S_z(n)B_z(n)S_y(n) \\ S_y(n) \\ S_u(n)D_z(n)S_y(n) \end{bmatrix} \tag{5}$$

$$C_c(n) = [S_u(n)C_z(n)\ \ S_u(n)D_z(n)[I-S_y(n)]\ \ I-S_u(n)] \tag{6}$$

$$D_c(n) = [S_u(n)\ D_z(n)\ S_y(n)]y(m,n) \tag{7}$$

with $\bar{u}(t) = u(m,n)$ for all t on $[(mP + n)T, (mP + n + 1)T)$.

The compensator parameters, $A_z(n)$, $B_z(n)$, $C_z(n)$, and $D_z(n)$, can be separated from the sampling schedule, $S_u(n)$, $S_y(n)$, $S_z(n)$, in an output-feedback representation of the GMCLS. Assuming a discretized model of the plant dynamics of the form

$$p(m,n+1) = A_p\, p(m,n) + B_p\, u(m,n) + E_p\, v(m,n) \tag{8}$$

$$y(m,n) = C_p\, p(m,n) + F_p\, w(m,n) \tag{9}$$

where v and w represent represent process and measurement noise, respectively, we can rewrite the closed loop system in the output feedback form

$$\begin{bmatrix} p(m,n+1) \\ c(m,n+1) \end{bmatrix} = \begin{bmatrix} A_p & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} p(m,n) \\ c(m,n) \end{bmatrix} + \begin{bmatrix} B_p & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} u(m,n) \\ c(m,n+1) \end{bmatrix} + \begin{bmatrix} E_p & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} v(m,n) \\ w(m,n) \end{bmatrix} \tag{10}$$

$$\begin{bmatrix} y(m,n) \\ c(m,n) \end{bmatrix} = \begin{bmatrix} C_p & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} p(m,n) \\ c(m,n) \end{bmatrix} + \begin{bmatrix} 0 & F_p \\ 0 & 0 \end{bmatrix} \begin{bmatrix} v(m,n) \\ w(m,n) \end{bmatrix} \tag{11}$$

$$\begin{bmatrix} u(m,n) \\ c(m,n+1) \end{bmatrix} = \begin{bmatrix} D_c(n) & C_c(n) \\ B_c(n) & A_c(n) \end{bmatrix} \begin{bmatrix} p(m,n) \\ c(m,n) \end{bmatrix} \tag{12}$$

Now the compensator matrices can be factored as follows.

$$\begin{bmatrix} D_c(n) & C_c(n) \\ B_c(n) & A_c(n) \end{bmatrix} = S_1(n) \begin{bmatrix} D_z(n) & C_z(n) \\ B_z(n) & A_z(n) \end{bmatrix} S_2(n) + S_3(n) \tag{13}$$

where $S_1(n)$, $S_2(n)$, and $S_3(n)$ are functions of $S_u(n)$, $S_y(n)$, and $S_z(n)$.

Equation (13) is important because it allows us to separate the unknown compensator parameters $A_z(n)$, $B_z(n)$, $C_z(n)$, and $D_z(n)$ from the known sampling schedule.

In the following section we will introduce two synthesis algorithms that can be used to determine the optimum compensator parameters $A_z(n)$, $B_z(n)$, $C_z(n)$, and $D_z(n)$.

III. Parameter Optimization Control Law Synthesis Methods

There are five well-recognized techniques for synthesizing multirate control laws: successive loop closures, pole placement, singular-perturbation-based methods, LQG Optimal methods, and parameter optimization methods.

The advantages of successive loop closures are that it is easy to use, that it can (conceivably) be used to synthesize control laws of arbitrary dynamic order and structure, and that it is particularly effective in applications where the control loops are not strongly dynamically coupled. Its disadvantage is that its one-loop-at-a-time approach cannot fully account for all dynamic coupling between control loops.

The problem with pole placement is determining where the closed-loop poles should be placed. It is a particularly difficult problem in the multirate (as compared to the single-rate) case because the STP-to-STP dynamics of multirate systems are periodically time-varying [2]. Only the BTP-to-BTP dynamics of multirate systems are time-invariant, and it is the poles of those dynamics that are assigned by pole-placement. In applications, determining desirable BTP-to-BTP closed-loop poles for a typical multirate system is difficult because the BTP of its sampling/update schedule will typically be longer than many of its desired closed-loop characteristic times.

Singular-perturbation-based control law synthesis methods amount to successive loop closures prefaced with a coordinate transformation to separate the full control law synthesis problem into two or more dynamically decoupled control law synthesis problems of different time scales. A complete decoupling requires changes in

not just the state coordinates, but in the input and output coordinates as well. Such a decoupling is not possible in the multirate case because the input and output coordinates represent physical sensor and actuator signals destined to be sampled/updated at different rates.

The advantages of the LQG optimal control law synthesis methods are that stabilizing control laws are relatively easy to obtain and that the control laws for all control loops are synthesized simultaneously, taking full advantage of all dynamic coupling between the control loops. The disadvantages are that the dynamic order and structure of the control law is fixed, that stability robustness objectives are difficult to achieve, and that the resulting control laws are periodically time-varying [2]-[3].

We favor parameter optimization methods for control law synthesis for multirate systems because they offer the principal advantages of the successive loop closures and LQG optimal synthesis methods. These advantages are that control laws of arbitrary dynamic order and structure can be synthesized, and that control laws for all control loops can be synthesized simultaneously, taking full advantage of all dynamic coupling between control loops. The disadvantage of parameter optimization methods is that a numerical search is required to determine the control law parameters.

In this section we present two parameter optimization methods for synthesizing multirate control laws. Both utilize the GMCLS discussed in Section II. The first is based on a finite-time quadratic cost function while the second is based on an infinite-time quadratic cost function. Both methods solve the multirate compensator synthesis problem by using a gradient-type numerical search to find a set of compensator parameters that minimize a quadratic cost function.

The multirate optimization problem is as follows.

Given:

1. The plant dynamics represented by

$$\dot{\tilde{p}}(t) = \tilde{A}_p \tilde{p}(t) + \tilde{B}_{pu} \tilde{u}(t) + \tilde{B}_{pv} \tilde{v}(t) \tag{14}$$

$$\tilde{y}(t) = \tilde{C}_p \tilde{p}(t) \tag{15}$$

   Here $\tilde{p}$ is the plant state vector, $\tilde{u}$ is the control input vector, $\tilde{y}$ is the noise-free measurement output vector, and $\tilde{v}$ is the noise input vector.

2. The complete sampling and update schedule for the compensator. This amounts to specifying $S_u(n)$, $S_y(n)$, and $S_z(n)$, for n=0,1, ... , $P-1$.

3. The order for the processor dynamics (the number of elements in z in (3)).

4. The desired structure (e.g., a diagonal structure) for the processor matrices, $A_z(n)$, $B_z(n)$, $C_z(n)$, and $D_z(n)$, for n=0,1, ... , $P-1$.

5. The number of distinct sets of processor matrices and when they are active. The optimization algorithms allow $A_z(n)$, $B_z(n)$, $C_z(n)$, and $D_z(n)$ to be periodically time varying. The designer can specify equality

relations among the compensator matrices. For example, if a time invariant compensator is desired then the designer can specify that $A_z(0) = A_z(1) = \cdots = A_z(P\text{-}1)$, and similarly for B_z, C_z and D_z.

6. The power spectral density $\tilde{V}$ of the process noise $\tilde{v}$ (in (8)).

7. The covariance $W(n)$, for n=0,1, ... , $P-1$, of the sensor noise w (in Fig. 2). w is assumed to be a periodically stationary, gaussian, purely random sequence, with period equal to the BTP of the sampling/update schedule.

8. The time t_f and non-negative definite weighting matrices $\tilde{Q}$ and $\tilde{R}$ for the performance index

$$J(t_f) = E\left\{\frac{1}{2t_f}\int_0^{t_f}\begin{bmatrix}\tilde{p}(t)\\ \tilde{u}(t)\end{bmatrix}^{\mathrm{T}}\begin{bmatrix}\tilde{Q} & 0\\ 0 & \tilde{R}\end{bmatrix}\begin{bmatrix}\tilde{p}(t)\\ \tilde{u}(t)\end{bmatrix}dt\right\} \tag{16}$$

where E is the expected value operator.

In the finite time optimization problem t_f must be a multiple of the BTP of the sampling/update schedule.

In the infinite time optimization problem $t_f \to \infty$ and $J_{infinite\text{-}time} = \lim_{t_f \to \infty} J(t_f)$

Find:

A set of processor matrices, $A_z(n)$, $B_z(n)$, $C_z(n)$, and $D_z(n)$, for n=0,1, ... ,$P-1$, such that the performance index

$$J_{ss} \triangleq \lim_{t_f \to \infty} J(t_f)$$

is minimized.

This optimization problem can be solved using either the finite-time cost function or the infinite-time cost function.

Solution Method Using the Finite-Time Cost Function.

The finite-time optimization algorithm was developed in connection with another research effort supervised by the Principal Investigator. This method synthesizes the multirate compensator that minimizes $J(t_f)$ for a finite t_f. A detailed discussion of this method can be found in [1]. A summary of the solution procedure follows.

1. Determine closed-form expressions for the performance index $J(t_f)$, and for its gradients with respect to the elements of the processor matrices $A_z(n)$, $B_z(n)$, $C_z(n)$, and $D_z(n)$, for n=0,1, ... ,P-1.

2. Use a gradient-type numerical optimization algorithm to determine a set of processor matrices, $A_z(n)$, $B_z(n)$, $C_z(n)$, and $D_z(n)$, for n=0,1, ... ,$P-1$, that minimizes $J(t_f)$.

3. Obtain a steady-state solution by re-optimizing for larger and larger t_f until t_f gets to be large compared to all of the closed-loop system's characteristic times.

The advantage of this method is that with t_f finite, the cost function $J(t_f)$ remains finite even if the compensator is destabilizing. The designer does not need to find a stabilizing compensator to start the optimization process as long as t_f is small enough that $J(t_f)$ does not exceed the numerical limits of the computer performing the optimization.

The disadvantage of this method is that the closed-form expressions that have been developed thus far for the performance index $J(t_f)$ and for its gradients with respect to the elements of the processor matrices are very complex and computationally intensive. In addition, we encountered difficulties when applying this method to the aircraft control problem because our solution algorithm lacked provisions for automatic scaling of the control law parameters (i.e., the independent variables) during the numerical search. The sheer complexity of the finite-time performance index and gradient expressions prevented us from adding the automatic scaling provisions that would have allowed us to apply this method to the aircraft control problem.

Solution Method Using Infinite-Time Cost Function

Instead of modifying our existing finite-time-based algorithm to alleviate the scaling problem discussed in the previous paragraph, we chose to develop a new infinite-time-based multirate sampled-data control law synthesis method, based on corresponding developments for single-rate systems by Mukhopadhyay [4], for which much simpler performance index and gradient expressions are easy to derive. For a complete description of that method, and the solution algorithm we developed to implement it see [5]. A summary of the solution procedure follows.

1. Find an initial stabilizing guess for the processor matrices $A_z(n)$, $B_z(n)$, $C_z(n)$, and $D_z(n)$, for n=0,1, ... ,P-1. The finite-time solution algorithm requires an initial stabilizing compensator because J_{ss} is infinite when the closed loop system is unstable. From our experience, many multirate problems can be stabilized using successive loop closures. The aircraft problem was open loop stable, and so determining a stabilizing compensator was trivial.

2. Determine the necessary conditions (given in [5]) for the processor matrices, $A_z(n)$, $B_z(n)$, $C_z(n)$, and $D_z(n)$, for n=0,1, ... ,P-1 to minimize J_{ss}. These are represented by three sets of coupled matrix equations. Two sets are Lyapunov equations, one governs the steady state covariance of the plant and control states, and the other governs a Lagrange multiplier. The third represents the gradient of J_{ss} with respect to the compensator parameters.

3. Use a gradient-type numerical search to solve the necessary conditions and determine a set of processor matrices, $A_z(n)$, $B_z(n)$, $C_z(n)$, and $D_z(n)$, for n=0,1, ... ,P-1, that minimizes J_{ss}.

The advantage of this method is that the gradient of J_{ss} with respect to the compensator parameters is easy to evaluate via the necessary conditions. For a given problem, the infinite-time optimization algorithm typically requires fewer computations to find the optimum compensator parameters than does the finite-time optimization algorithm even when both algorithms are initialized with the same stabilizing compensator.

Even though the finite-time and infinite-time based solution algorithms can determine optimum compensator parameters, there is no guarantee that the design will be robust. In the following section we present a method for analyzing the robustness of a multirate control system.

IV. Gain and Phase Margins for Multirate Systems Using Singular-Values

There are many established methods for synthesizing multirate compensators, see Section III, but surprisingly few methods for analyzing the robustness of these systems. Current robustness analysis methods rely principally on the transfer function of the system. A multirate transfer function, in the traditional sense, does not exist, because multirate systems are periodically time varying. Without modification, established single-rate analysis methods cannot be applied directly to multirate systems.

As part of this project, we developed an approach for extending the nyquist criterion and singular value analysis to multirate and periodically time varying systems. For a detailed discussion of this approach, including application of structured singular value robustness analysis to multirate systems, see [6]. In this section we present a summary of the important ideas from that paper used to calculate gain and phase margins of multirate systems using singular values.

As we saw in Section II, a multirate compensator can be modeled as a linear periodically time varying system (1)-(2). Equations (1)-(2) from Section II can be written as

$$c(m,n+1) = A_c(n)c(m,n) + B_c(n)y(m,n) \tag{17}$$

$$u(m,n) = C_c(n)c(m,n) + D_c(n)y(m,n) \tag{18}$$

This system (17)-(18) can then be transformed to an *equivalent single-rate system* (ESRS) by repeated application of (17)-(18) over the BTP [7]. The ESRS has the form:

$$c(m+1,0) = A_e c(m,0) + B_e \hat{y}(m,0) \tag{19}$$

$$\hat{u}(m,0) = C_e c(m,0) + D_e \hat{y}(m,0) \tag{20}$$

$$\text{where } \hat{y}(m,0) = \begin{bmatrix} y(m,0) \\ y(m,1) \\ \vdots \\ y(m,P-1) \end{bmatrix} \quad \hat{u}(m,0) = \begin{bmatrix} u(m,0) \\ u(m,1) \\ \vdots \\ u(m,P-1) \end{bmatrix} \tag{21}$$

The transfer function for the ESRS is

$$\hat{u}(z^P) = G_P(z^P)\hat{y}(z^P) \tag{22}$$

$$\text{where } G_P(z^P) = C_e(Iz^P - A_e)^{-1}B_e + D_e \tag{23}$$

For a detailed discussion of the ESRS, see [6]. The key points are:

1. The ESRS is a time invariant single-rate system with a sampling period of one BTP and the unique property that the inputs are time correlated and the outputs are time correlated.

2. In general $G_P(z^P)$ has a very complicated form, but it can be shown that if the system is time invariant with $G(z)$ equal to a constant, then $G_P(z^P)$ will also be constant and block diagonal with $G(z)$ on the diagonal.

3. The ESRS allows us to manipulate time invariant and periodically time varying systems (e.g. multirate) as if they were both time invariant. The state space or transfer functions descriptions can be used to calculate input-output relations for systems in series or in a feedback loop just as in classical control [8]. For example, to calculate the ESRS of a multirate compensator in series with a time invariant plant, we would calculate the ESRS of the plant and compensator individually and then combine them using block diagram arithmetic.

4. Kono [9] has shown that if the ESRS is stable then the multirate system from which it was derived will be stable.

5. Single-rate robustness analysis techniques can be applied to the ESRS as long as the results are interpreted in light of the fact that some of its inputs and outputs are time correlated.

Generalized gain and phase margins for the ESRS (and equivalently the multirate system) can be calculated using singular value analysis. If we assume a plant uncertainty of the form

$$G(z)_{Actual} = G(z)_{Nominal} k e^{j\theta} \tag{24}$$

then the ESRS plant uncertainty has the form

$$G_P(z^P)_{Actual} = G_P(z^P)_{Nominal} (k e^{j\theta})_P \tag{25}$$

$$(k e^{j\theta})_P = \text{diag}[k e^{j\theta}, k e^{j\theta}, \ldots, k e^{j\theta}] \text{ with } P \text{ blocks}$$

[Recall that if $H(z)$ is constant, $H_P(z^P)$ is block diagonal with $H(z)$ on the diagonal.]

The multirate system is guaranteed to remain stable whenever

$$\bar{\sigma}((k e^{j\theta})^{-1} - 1) \leq \underline{\sigma}(I + G_P(z^P)) \text{ on the nyquist contour} \tag{26}$$

Traditional gain margins can be obtained by setting $\theta = 0$ and solving (26) for k. Phase margins can be found by setting $k = 0$ and solving (26) for θ.

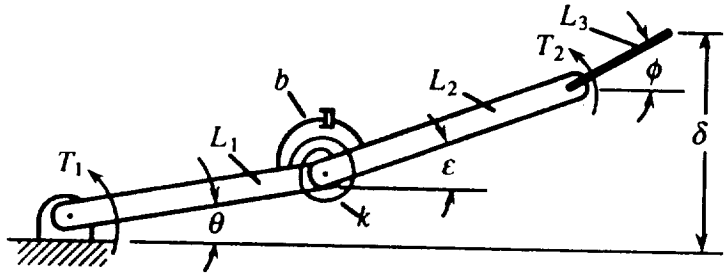
As with most singular value robustness analysis methods, the k and θ found using (26) are conservative. If, however, $k e^{j\theta}$ is diagonal, the conservativeness associated with (26) can be reduced by diagonally scaling $G_P(z^P)$. We used Osborne's method of preconditioning matrices to increase the lower bound for $G_P(z^P)$ and thus to improve our estimate of the gain and phase margins.

V. Application of the Finite-Time-Based Parameter Optimization Algorithm to a Two Link Robot Arm Control Problem

The original proposal for this project called for the finite-time-based parameter optimization multirate sampled-data control law synthesis method of Section III to be applied to an aircraft control system design problem. That method and a solution algorithm to implement it had been previously developed as part of another research effort supervised by the Principal Investigator. Due to the solution algorithm's lack of adequate provisions for automatic scaling of the control law parameters (i.e., the independent variables) during the numerical search, we were not able to apply it successfully to the aircraft control problem. We maintain, however, that the problems we encountered with it were a consequence of problems with the *solution algorithm* and are not necessarily indicative of problems with the synthesis method.

In this section we therefore present an application of the finite-time-based multirate sampled-data control law synthesis method to a two-link robot arm (TLA) control problem. The robot arm application demonstrates the utility of the method without being so poorly conditioned that automatic scaling of the control law parameters was required during the numerical search.

The two-link robot arm system we dealt with is shown in Figure 3. The first link is long and massive, for large-scale slewing motions. The second is short and lightweight so high-bandwidth control of the tip position can be achieved with a relatively small motor at the second joint. The pin joint, rotational spring, and rotational damper at the midpoint of the first link model flexibility in that link. The control inputs are the motor torques, T_1 and T_2. The measured outputs are the joint angle θ and the tip position δ. The spring constant (k) and damping coefficient (b) values (in Fig. 3) yield an open-loop vibration mode with a 10 Hz natural frequency and 1% damping.



Parameters:	Mass	Length	
L_1	0.5 kg	0.5 m	
L_2	0.5 kg	0.5 m	k = 37.33 N/rad
L_3	0.04 kg	0.2 m	b = 0.012 N· s/m

The natural frequency of the vibration mode is 10 hz.

Inputs: Torques T_1 and T_2

Outputs: θ and δ

Figure 3 Two-Link Robot Arm System

We used the finite-time-based multirate control law synthesis method of Section III to synthesize multirate sampled-data control laws for this system. Our performance objective was high-bandwidth control of the tip position δ, and it is intuitively clear that this can best be accomplished, given a fixed real-time computation capability, by trading low-bandwidth control at T_1 for high-bandwidth control at T_2. Thus, for an 8-to-1 control bandwidth ratio, we chose the sampling/update rate for δ and T_2 to be 8-times faster than that for θ and T_1. For comparison purposes, we also designed corresponding analog and single-rate sample-data control laws.

TLA Control Laws

For the TLA system, the tip position (δ) responses to a commanded step change in the tip position obtained with the analog, single-rate and multirate control laws we synthesized are shown in Figure 4. See [1] for additional results and details. A summary description of those designs follows.

LQR Analog Design The LQR Analog response was obtained with an analog LQR (full state feedback) control law that is optimal with respect to a quadratic performance index that yields 0.7071 damping ($\zeta_1 = \zeta_2 = 0.7071$) and an 8-to-1 ratio of characteristic frequencies ($\omega_{n2}/\omega_{n1} = 8$) for the two closed-loop modes.

Third-Order Analog Successive Loop Closures Design The Third-Order Analog Successive Loop Closures response was obtained with a successive loop closures control law that consisted of a single lead network from θ to T_1, and two identical cascaded lead networks from δ to T_2. The gains, and zero and pole locations were chosen to yield dominant closed-loop poles coincident with those obtained with the LQR Analog control law.

Third-Order Multirate Tustin Design The Third-Order Multirate Tustin response was obtained with a multirate sampled-data control law obtained by discretizing the lead compensators of the Third-Order Analog Successive Loop Closures design using Tustin's method [10]. The θ-to-T_1 control-loop sampling/update rate is a factor of 8 times the characteristic frequency of the lower-frequency closed-loop mode from the Third-Order Analog Successive Loop Closures design; and the δ-to-T_2 sampling/update rate is the same multiple of the characteristic frequency of the higher-frequency closed-loop mode from the Third-Order Analog Successive Loop Closures design.

Optimized Third-Order Multirate Tustin Design The Optimized Third-Order Multirate Tustin response was obtained with a control law synthesized by the finite-time-based multirate sampled-data control law synthesis method of Section III. This control law is the Third-Order Multirate Tustin control law, but with its gains and its pole and zero locations optimized to minimize the same performance index as is minimized by the LQR Analog control law.

Analog Third-Order Design The Analog Third-Order response was obtained with a third-order, generally-structured, analog control law synthesized using Ly's Sandy algorithm [11]-[12] to minimize the same performance index as is minimized by the LQR Analog control law.

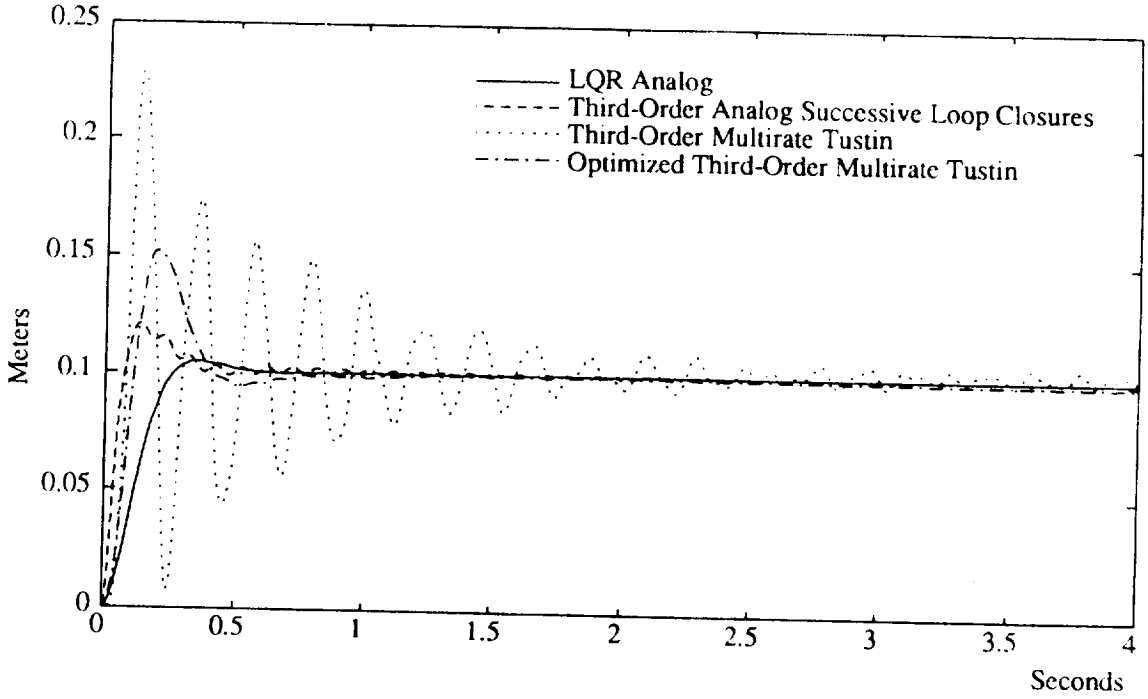


Figure 4a Robot Arm Tip Position (δ) Responses to a Tip Position Step Command

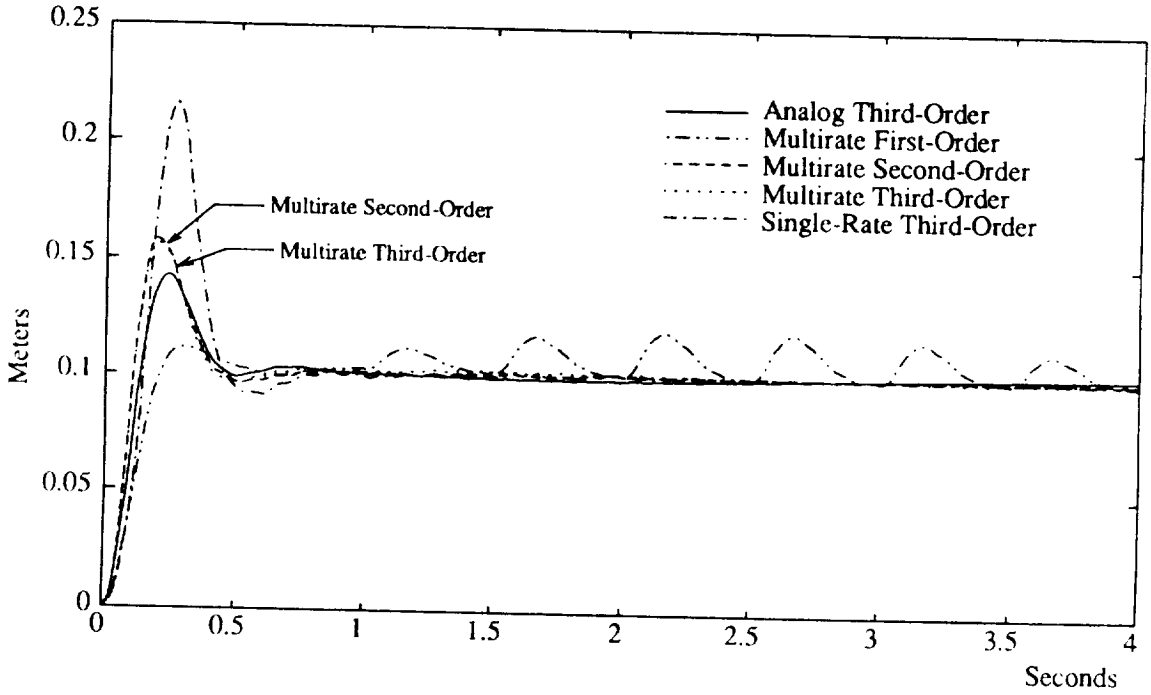


Figure 4b Robot Arm Tip Position (δ) Responses to a Tip Position Step Command

Multirate First-Order, Second-Order & Third-Order Designs The Multirate First-Order, Second-Order, and Third-Order responses were obtained with multirate, generally-structured, sampled-data control laws synthesized by the finite-time-based multirate sampled-data control law synthesis method of Section III to minimize the same performance index as is minimized by the LQR Analog control law. The sensor sampling and actuator update rates are the same as in the Third-Order Multirate Tustin control law. In the First-Order case, the update rate for the one processor state is the same as the faster sensor-sampling/actuator-update rate. In the Second-Order case, one processor state is updated at the faster rate and the other at the slower rate. In the Third-Order case, two processor states are updated at the faster rate and one is updated at the slower rate.

Single-Rate Third-Order Design Finally, the Single-Rate Third-Order control law response was obtained with a single-rate, generally-structured, sampled-data control law synthesized by the finite-time-based multirate sampled-data control law synthesis method to minimize the same performance index as is minimized by the LQR Analog control law. Its single sampling/update rate was chosen to require the same average number of computations per unit time for real-time operation as is required for real-time operation of the Multirate Third-Order control law.

Conclusions

The TLA results in Figure 4 demonstrate some of the benefits of multirate control. For example, the tip position overshoot (δ) with the multirate compensator is much less than with its equivalent single-rate counterpart. But more importantly, the results demonstrate that the finite-time-based multirate sampled-data control law synthesis method can be used to synthesize multirate control laws of arbitrary structure and dynamic order, with arbitrarily selected sampling rates for all sensors, and update rates for all processors states and actuators. The third-order multirate compensator, for example, uses two different update rates for the processor states, inputs and outputs, and a general compensator structure with full coupling between inputs, outputs and processor states of different rates.

VI. Application of the Infinite-Time-Based Parameter Optimization Algorithm to a Yaw Damper and Modal Suppression System for a Commercial Aircraft

A practical application of multirate control can be found in aircraft. The limited computational resources of aircraft dictate that their control systems must function efficiently. Multirate control allows the designer to efficiently allocate these resources by trading slow sampling and update rates in control loops associated with low-bandwidth control functions for fast sampling and update rates in control loops associated with high-bandwidth control functions. In this section we consider a particular application of multirate control: a combination yaw-damper and modal suppression system for a commercial aircraft.

In the interest of weight reduction for fuel efficiently, aircraft are being constructed with less structural rigidity. Structural vibration modes can be excited in such aircraft by wind gusts or by movements of control surfaces. These vibrations affect not only the structural integrity of the fuselage but also passenger ride quality. In the lateral direction, such vibrations are often induced by rudder activity associated with the yaw-damper. A "modal suppression system" can be added to the yaw-damper loop to suppress these vibrations. The modal suppression system would traditionally be designed by successive loop closures.

In this section we describe the design of a multirate combination yaw-damper and modal suppression system for a commercial aircraft using the infinite-time-based multirate compensator synthesis algorithm and robustness analysis technique discussed in Sections III and IV. For comparison purposes we also designed corresponding analog and single-rate sample-data systems.

The goal for each compensator design was to increase the damping of the dutch-roll mode to 0.6, and to decrease the covariance of lateral accelerations at the nose and aft of the airplane, particularly those components associated with low frequency flexible modes. The performances of the compensators were compared by comparing the closed loop dutch-roll damping, the covariances of lateral accelerations at the nose and aft of the aircraft due to a unit covariance gaussian white noise disturbance, and the PSD plots of the lateral accelerations at the nose and aft of the aircraft for either a white noise disturbance (analog designs) or a gust pulse disturbance (sampled-data designs).

Open Loop Aircraft

A block diagram of the airplane model is shown in Figure 5. The lateral dynamics model consists of 4 rigid body modes (heading, spiral, dutch roll and roll) and 11 flexible modes. Actuator/power control units for the aileron and rudder are modeled as second-order lags.

$$G(s)_{Rudder} = \frac{30(35)}{(s+30)(s+35)} \quad G(s)_{Aileron} = \frac{20(25)}{(s+20)(s+25)} \tag{27}$$

The lateral gust disturbances are filtered by a second-order Dryden gust model

$$G(s) = \frac{17.496s + 2.1617}{21.836s^2 + 9.3458s + 1} \tag{28}$$

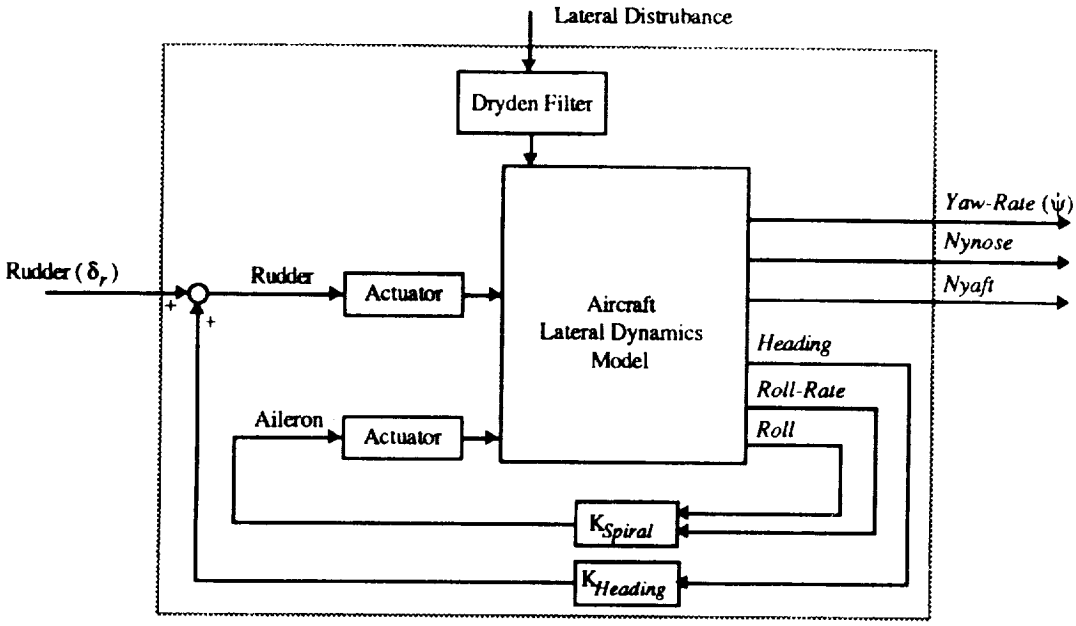


Figure 5 Block Diagram of the Open Loop Airplane Model

The poles associated with the spiral and heading modes were compensated with static gain feedback before the yaw-damper/modal suppression systems were designed, because these modes, which laid close to the origin and were controllable with the rudder, created numerical difficulties for Sandy (the optimization program used to design the Fourth-Order Analog compensator discussed in later in this section). The spiral mode was compensated by feeding back roll and roll-rate to the aileron. Heading was compensated with heading to rudder feedback. In what follows we refer to the airplane model with spiral and heading modes compensated as the uncompensated airplane (no dutch-roll compensation).

The lateral accelerations of the uncompensated airplane are measured by *Nynose* and *Nyaft*. The PSD plots of lateral accelerations for the uncompensated airplane are shown in Figure 6. A yaw-damper/modal suppression system should reduce the total area under this curve (covariance of lateral acceleration). In particular, it should reduce the peak at ≈ 0.5 Hz (near the dutch-roll mode) and the peaks between 3 Hz and 6 Hz (low frequency flexible modes). Values of the dutch-roll damping, and the *Nynose* and *Nyaft* covariances for the uncompensated airplane are given in Table 1.

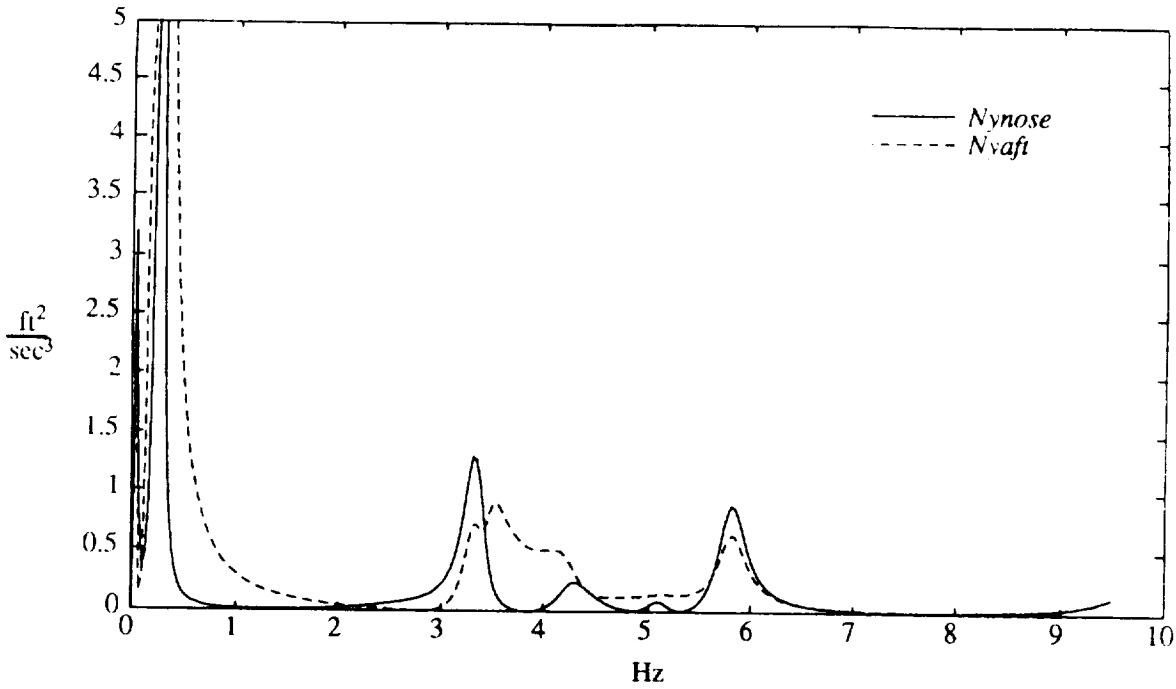


Figure 6 PSD of *Nynose* and *Nyaft* for Uncompensated Airplane with Unit Covariance Gaussian White Noise Lateral Disturbance

Table 1 Results for Analog Designs with a Unit Covariance Gaussian White Noise Lateral Disturbance

Design	Dutch-Roll Damping	*Nynose* Cov. (ft^2/sec^3)	*Nyaft* Cov. (ft^2/sec^3)
Uncompensated	0.08	5.1	21.8
Analog Yaw-Damper Only	0.6	5.0	6.1
LQR Analog	0.6	2.4	3.1
Fourth-Order Analog	0.55	2.5	2.4

Analog Yaw-Damper/Modal Suppression System Designs

Three analog compensators were designed: a yaw-damper only system, a full state feedback yaw-damper/modal suppression system, and a fourth-order yaw-damper/modal suppression system. PSD plots of *Nynose* and *Nyaft* for the analog designs are shown in Figure 8; *Nynose* and *Nyaft* covariances are summarized in Table 1. These analog designs provide a base line for comparison with the sampled-data designs and were used to determine appropriate values for cost function weighting matrices. Following is a summary of these designs.

Analog Yaw-Damper Only Design The yaw-damper only design uses static feedback from $\dot{\psi}$ to δ_r, using a gain k_{damper}. We chose k_{damper} such that the dutch-roll damping was 0.6 using classical root locus. While the peak on the PSD plot associated with the dutch-roll mode (≈0.5 Hz) has been reduced significantly from the uncompensated case, the peak near 3 Hz has increased (see Fig. 8). This is the problem with using static gain feedback. As you "press" on one peak of the PSD another "pops" up due to the input coupling between the dutch-roll and low frequency flexible modes.

LQR Analog Design The LQR design uses full state feedback to improve the dutch-roll damping and reduce the covariance of *Nynose* and *Nyafi*. The compensator was designed to minimize the following cost function.

$$J_{ss} = \lim_{t_f \to \infty} E\left\{ \begin{bmatrix} Nynose \\ Nyafi \end{bmatrix}^T \begin{bmatrix} 0.001 & 0 \\ 0 & 0.004 \end{bmatrix} \begin{bmatrix} Nynose \\ Nyafi \end{bmatrix} + 1.6\delta_r^2 \right\} \tag{29}$$

Weighting matrices for (29) were chosen such that the covariances of *Nynose* and *Nyafi* were reduced from the yaw-damper only case by the same percent, and the dutch-roll mode had a damping of 0.6. Figure 8 shows that the LQR design significantly reduces the dutch-roll peak as well as the peaks associated with the flexible modes.

Fourth-Order Analog Design The Fourth-Order Analog compensator is a yaw damper/modal suppression system designed using Sandy [11]. A block diagram of this compensator is shown in Figure 7. This design minimizes the same cost function as the LQR design (29) with the weighting on δ_r adjusted to achieve close to the desired 0.6 dutch-roll damping. An unexpected result is that the covariance at *Nyafi* for the Fourth-Order Analog design is actually better than for the LQR Analog design. This is a consequence of adjusting the cost function weighting matrices to achieve the desired the dutch-roll damping.

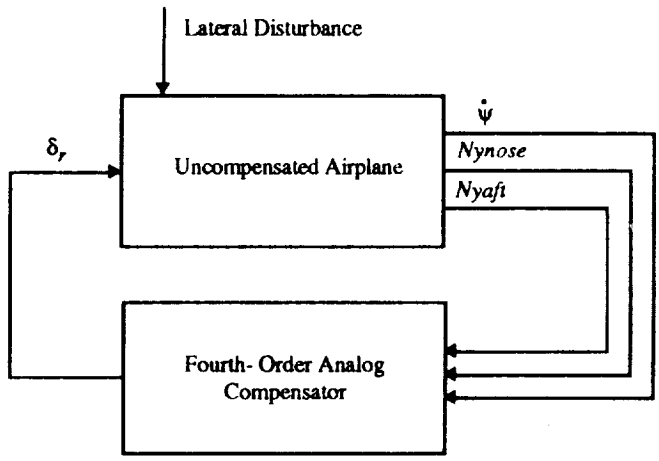


Figure 7 Block Diagram of Airplane with Fourth-Order Analog Compensator

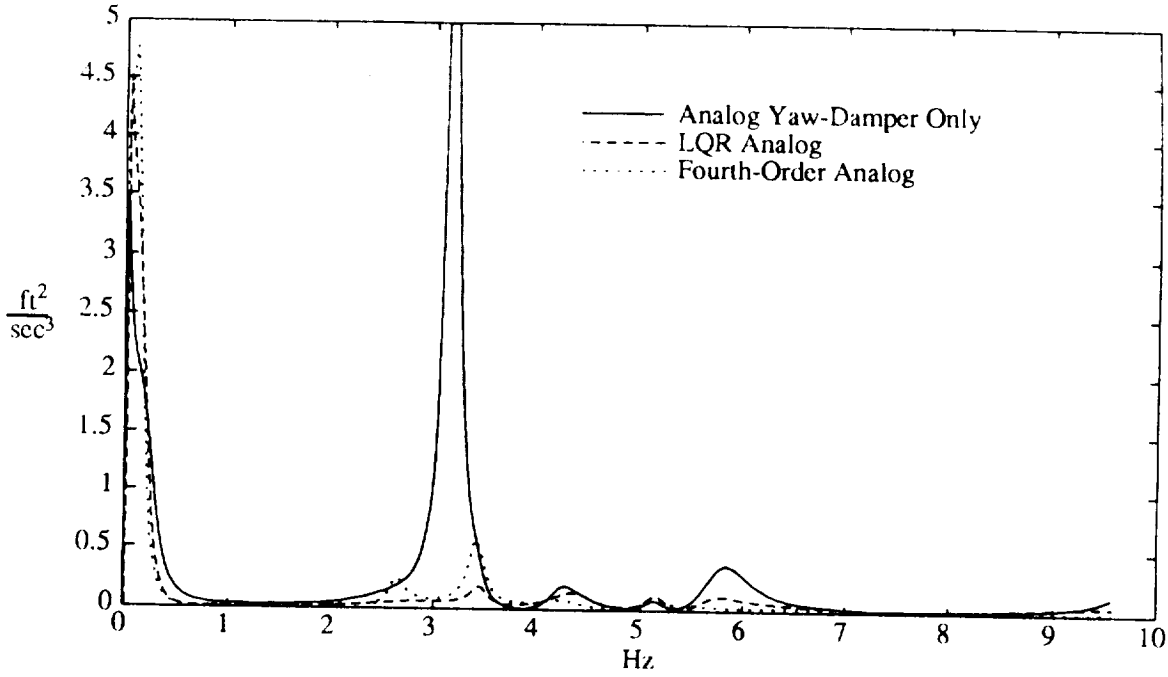


Figure 8a PSD of *Nynose* for Airplane with Analog Compensators for Unit Covariance Gaussian White Noise Lateral Disturbance

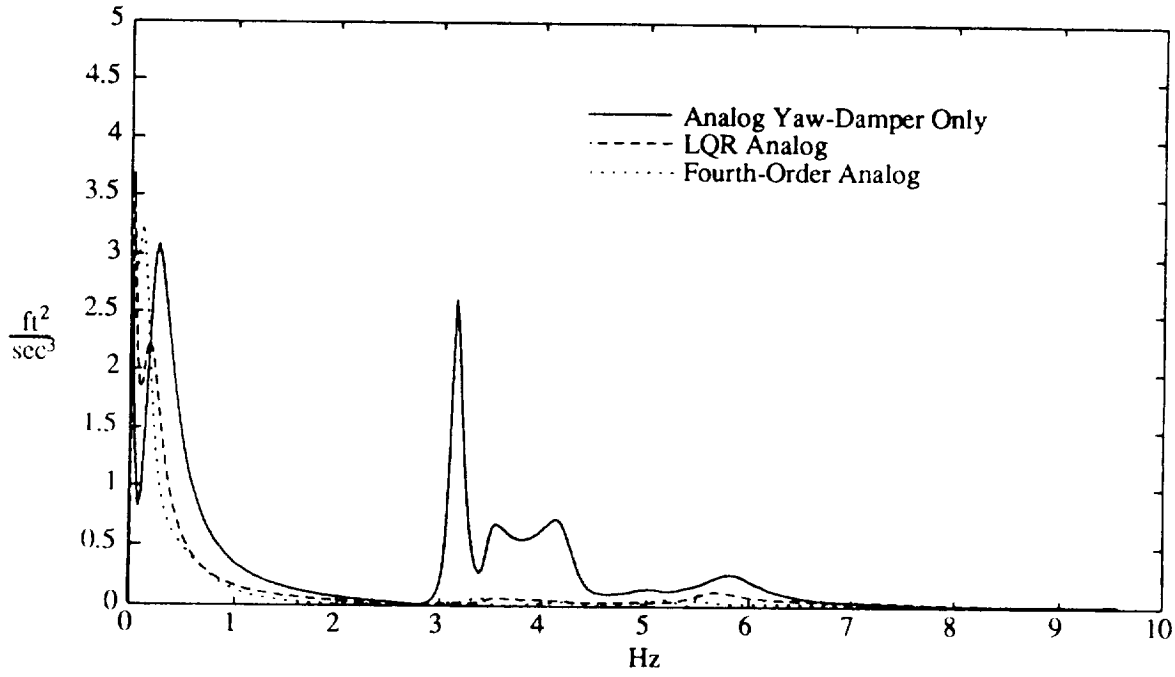


Figure 8b PSD of *Nyaft* for Airplane with Analog Compensators for Unit Covariance Gaussian White Noise Lateral Disturbance

Sampled-Data Yaw-Damper/Modal Suppression System Designs

Three sampled-data compensators were designed: a single-rate yaw-damper only system, a fourth-order multirate compensator and a fourth-order single-rate compensator. Both fourth-order compensators were synthesized using our infinite-time-based multirate control law synthesis algorithm to minimize the same cost function as the LQR Analog design.

The sampled-data compensator designs were based on a maximum sample/update rate of 50 Hz. This is 10 times the rudder actuator roll off frequency and 8 times faster than the fastest flexible mode which contributes significantly to the PSD of the lateral acceleration. This sample rate is close to the slowest practical sample rate which could be used.

PSD plots for the sampled-data designs were generated using a gust pulse (a rectangular pulse) at the disturbance input, as opposed to the gaussian white noise used for the analog designs. For the analog designs, the PSD plots were based on transfer functions from the disturbance input to *Nynose* and *Nyaft*. Multirate compensators are periodically time varying so that transfer functions for them, in the traditional sense, do not exist. For this reason, we used the gust pulse disturbance input to generate the PSD plots for the sampled-data designs.

The gust pulse input PSD has a connection to the white noise input PSD. For a time invariant continuous system, the PSD plots generated using either gaussian white noise or a *continuous* impulse input are exactly the same. This is because the Fourier transform of the impulse response is the same as the bode plot, and the PSD of gaussian white noise is a constant. If a continuous system, given by $\dot{x}(t) = Ax(t) + Bu(t)$, is such that

$$B \approx u \int_0^{T_p} e^{-A\tau} B d\tau \tag{30}$$

where u can be selected arbitrarily and T_p is much shorter than the observation time, then a continuous impulse can be approximated by a pulse of duration T_p and magnitude u. For the airplane problem addressed in this project, (30) is satisfied for

$$T_p = 0.02 \text{ seconds and } u = 50 \text{ ft/sec.} \tag{31}$$

PSD plots of *Nynose* and *Nyaft* for the sampled-data designs are shown in Figure 11. *Nynose* and *Nyaft* covariances for these designs are summarized in Table 2. Following is a summary of the sampled-data designs.

Single-Rate Yaw-Damper Only Design The Single-Rate Yaw-Damper Only design is similar to the Analog Yaw-Damper design except that a sampler is used at the output $\dot{\psi}$ and a zero order hold is used at the input δ_r. Both the sampler and zero order hold operate at 50 Hz. The performance of the sampled-data yaw damper is very close to that of the analog damper (Figs. 8 and 11).

Multirate Fourth-Order Design The multirate compensator is shown in Figure 9. It was designed to minimize the same cost function as the LQR Analog design with the weighting on δ_r adjusted to achieve the desired dutch-roll damping. The compensator uses two sampling/update rates. The rudder is updated and the lateral accelerations are sampled at 50 Hz; $\dot{\psi}$ is sampled at a slower rate, 12.5 Hz, because it is composed primarily of the slow dutch-roll mode.

Two of the processor states for this multirate compensator are updated at the fast rate, 50 Hz, and two are updated at the slow rate, 12.5 Hz. Initially a compensator was designed in which all of the processor states were updated at 50 Hz, but we found that there was no noticeable performance degradation if two of the processor states were updated at the slower rate. Slowing the update rate of these states reduces the number of computations required per unit time for real time implementation of the multirate compensator.

Table 2 Results for Sampled-Data Designs with a Unit Covariance Gaussian White Noise Lateral Disturbance

Design	Dutch-Roll Damping	*Nynose* Cov. (ft^2/sec^3)	*Nyafi* Cov. (ft^2/sec^3)
Uncompensated	0.08	5.1	21.8
Single-Rate Yaw-Damper Only	0.6	4.3	5.4
Multirate Fourth-Order	0.6	3.6	4.7
Single-Rate Fourth-Order	0.6	3.5	4.7

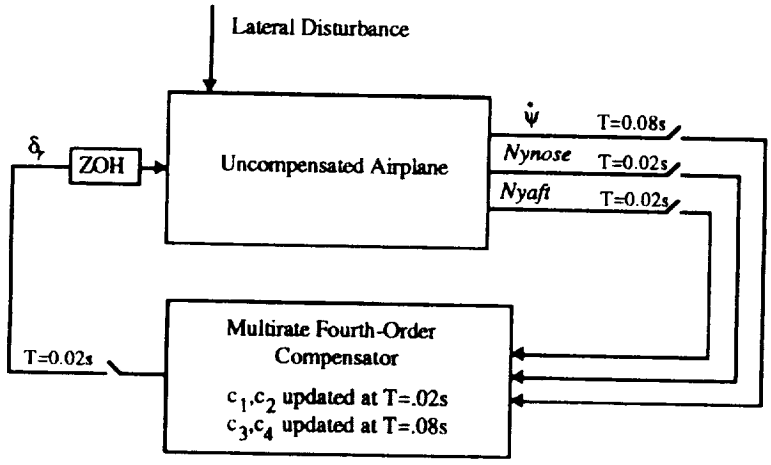


Figure 9 Block Diagram of Airplane with Multirate Fourth-Order Compensator

Single-Rate Fourth-Order Design The Single-Rate Fourth-Order compensator is shown in Figure 10. The sampling rate for this compensator is 28.6 Hz. That rate was chosen such that the number of multiplications required per unit time for its real time operation is the same for the multirate compensators. The cost function used to design the single-rate compensator was the same as was used to design the Multirate Fourth-Order compensator.

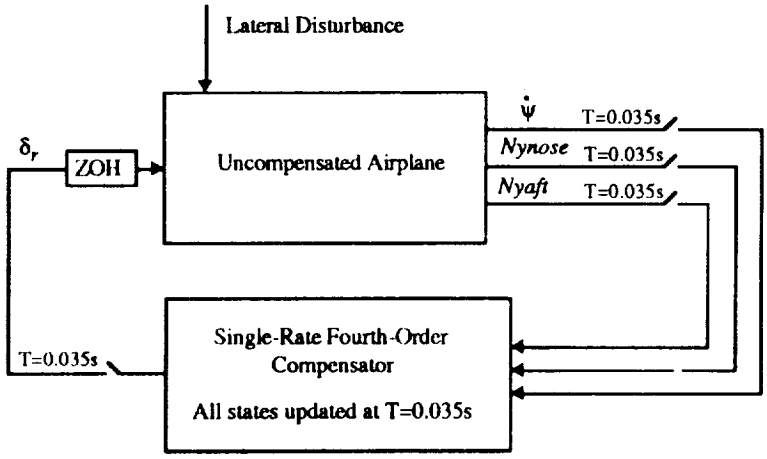


Figure 10 Block Diagram of Airplane with Single-Rate Fourth-Order Compensator

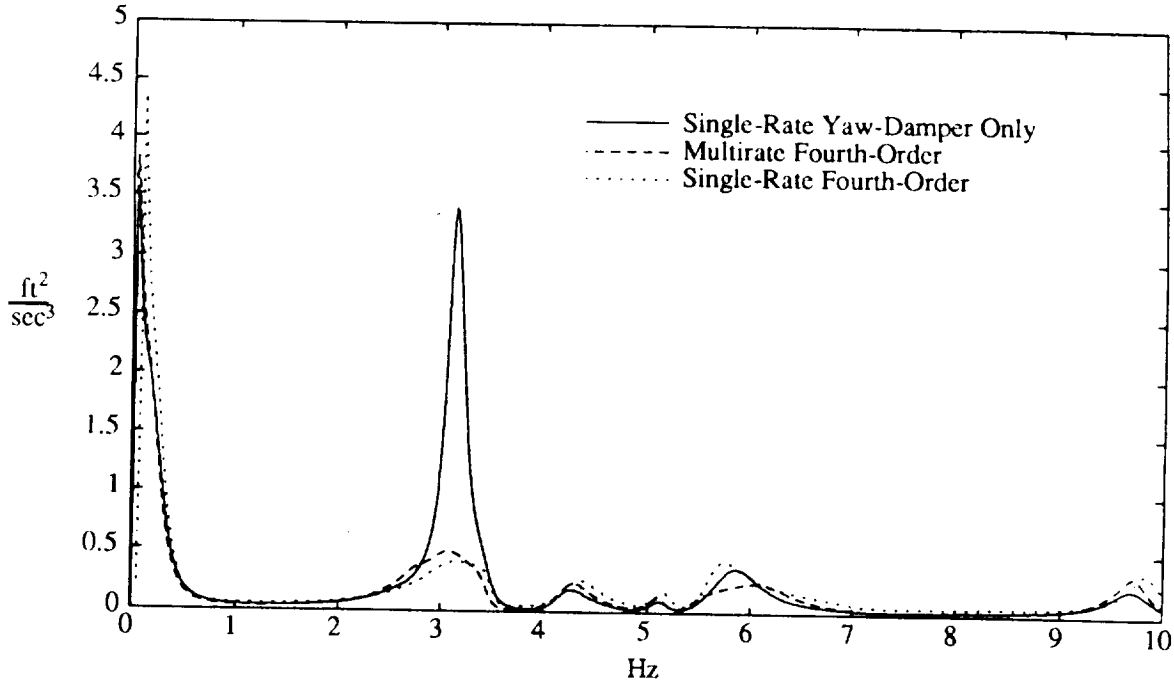


Figure 11a PSD of *Nynose* for Airplane with Sampled-Data Compensators for Gust Pulse Lateral Disturbance

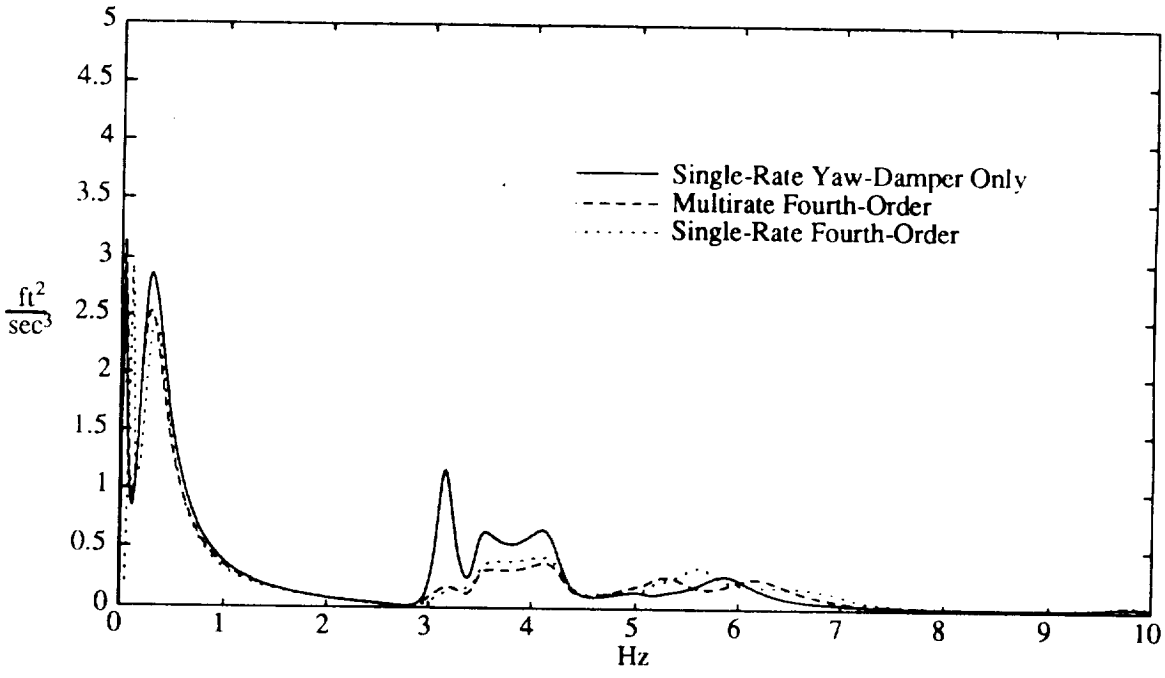


Figure 11b PSD of *Nyaft* for Airplane with Sampled-Data Compensators for Gust Pulse Lateral Disturbance

Gain and Phase Margins for Sampled-Data Designs

Gain and phase margins at the control input (δ_r) were evaluated for the Multirate Fourth-Order and Single-Rate Fourth-Order compensators using the robustness analysis methods of Section IV. Table 3 summarizes the traditional gain and phase margins for the these compensators. Figure 12 shows the region of guaranteed stability for simultaneous changes in k and θ for both compensators.

Table 3 Traditional Gain and Phase Margins

Design	Gain Margin (db) [$\theta = 0$]	Phase Margin (Deg) [$k = 0$db]
Multirate Fourth-Order	[-3.8, 7.1]	$\pm 32^{\circ}$
Single-Rate Fourth-Order	[-3.3, 5.5]	$\pm 27^{\circ}$

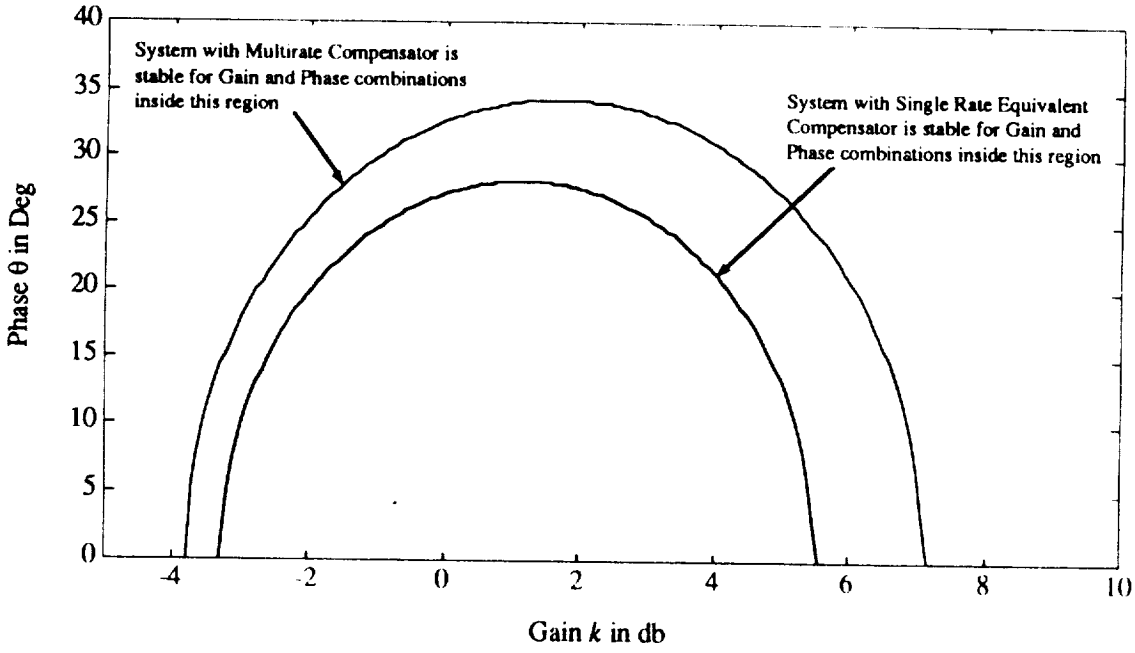


Figure 12 Stability Region for Simultaneous Gain and Phase Uncertainty for Fourth-Order Sampled-Data Compensators

Conclusions

Figures 8 and 11 show that the yaw-damper/modal suppression systems significantly decrease the covariance of the lateral acceleration at the nose and aft of the airplane while attaining the desired 0.6 dutch-roll damping. It should be no surprise that the analog compensators out performed the sampled-data compensators because the sampled-data compensators were designed using a slow sampling rate. Still, both fourth-order sampled-data compensators reduce the peak accelerations of *Nynose* and *Nyaft* by 175% and 50% respectively over the yaw-damper only systems. The performance of the Single-Rate Fourth-Order compensator is nearly as good as that of the multirate compensator, but, for input gain and phase uncertainty, the multirate compensator is more robust than the single-rate compensator.

VII. Summary and Conclusions

In this report we have presented a methodology for designing multirate control systems. We have introduced the Generalized Multirate Control Law Structure (GMCLS) which allows complete flexibility with regard to the dynamic order and structure of the control law, and with regard to the sampling rates for all sensors and the update rates for all processor states and actuators. We have presented two parameter optimization multirate control law synthesis algorithms, one based on an infinite-time cost function and the other based on a finite-time cost function, which can be used to find optimum values for the GMCLS parameters. We have presented a technique for determining gain and phase margins for multirate systems. Finally, we have demonstrated our methodology by applying it to the design of a two link robot arm control system and to the design of a combination yaw-damper and modal suppression system for a commercial aircraft. The application to the aircraft control problem, in particular, demonstrates that the methodology can be applied to design problems of a scale that one might expect to encounter in practice.

VIII. Suggestions for Future Research

The results presented here demonstrate a methodology for multirate digital control system design that is applicable to practical problems. Before this methodology can be routinely applied in practice, however, the following need to be developed:

1. A means for directly synthesizing *robust* multirate control laws.

2. Numerical optimization algorithms incorporating auto-scaling of the independent variables and other features that more effectively deal with the practical difficulties of parameter optimization applied to multirate control law synthesis.

With regard to direct synthesis for robustness, there are several possibilities. One would add the multiple-plant-condition design for robustness ideas of Ly [10]-[11]. A second would add direct nonlinear robustness constraints on the control law parameters during the numerical optimization. The latter approach has been

successfully applied by Mukhopadhyay [4] to synthesize robust single-rate control laws by parameter optimization.

In addition to theoretical work, a second major research effort in multirate control needs to be directed toward experimental research. Now that a bonafide multirate control system design methodology has been developed, we strongly believe that further substantive progress in the field can best be made in conjunction with bonafide hardware applications of that methodology in the laboratory.

REFERENCES

1. Berg, M.C., Mason, G.S., and Yang, G.S., "A New Multirate Sampled-Data Control Law Structure and Synthesis Algorithm," to be presented at the 1991 American Control Conference. **Preprint in Appendix A.**
2. Berg, M.C., Amit N. and Powell J.D., "Multirate Digital Control system Design," *IEEE Trans. Auto. Contr.*, Vol AC-33, Dec 1988, pp.1139-1150.
3. Glasson, D.P., "A New Technique for Multirate digital control Design and Sample Rate Selection," *AIAA Jour. Guid., Contr and Dynamics*, Vol. 5, Aug 1982, pp. 379-382.
4. Mukhopadhyay, V., "Digital Robust Control Law Synthesis Using Constrained Optimization," *AIAA Jour. Guid., Contr. and Dynamics*, Vol. 12, March-April 1989, pp. 175-181.
5. Mason, G.S. and Berg, M.C., "Reduced Order Multirate Compensator Synthesis," to be presented at the 1991 American Control Conference and accepted for publication in the *AIAA Jour. Guid., Contr. and Dynamics*. **Preprint in Appendix B.**
6. Mason, G.S., and Berg, M.C., "Robustness Analysis of Multirate and Periodically Time Varying Systems with Structured and Unstructured Uncertainty." to be presented at the 1991 AIAA Guidance and Control Conference. **Preprint in Appendix C.**
7. Meyer, R.A., and Burrus, C.S., "A Unified Analysis of Multirate and Periodically Time-Varying Digital Filters," *IEEE Trans. Circuits in Systems*, Vol. CAS-22, No. 3, March 1975, pp. 162-168.
8. Khargonekar, P.P, Poolla, K., and Tannenbaum, A., "Robust Control of Linear Time-Invariant Plants Using Periodic Compensation," *IEEE Trans. Auto. Contr.*, Vol. AC-30, No. 11, Nov 1985, pp. 1088-1096
9. Kono, M., "Eigenvalue Assignment in Linear Periodic Discrete-Time Systems," *Int. J. Control*, Vol. 32, No. 1, 1980, pp. 149-158.
10. Franklin, G.F., Powell, J. David, and Workman, M.L., *Digital Contol of Dynamic Systems*, Addison-Wesley Pub. Co. 1990.
11. Ly, U.L., "A Design Algorithm for Robust Low-Order Controllers." PhD Thesis, Stanford Univ., Stanford, CA, 1982.
12. Ly, U.L., "Robust Control Design Using Nonlinear Constrained Optimization," *Proc. Amer. Contr. Conf.*, San Diego, CA, May 1990, pp. 968-969.

APPENDIX A

PREPRINT OF REFERENCE 1

Submitted to the AIAA Journal of Guidance, Control and Dynamics

A NEW MULTIRATE SAMPLED-DATA CONTROL LAW STRUCTURE AND SYNTHESIS ALGORITHM

Martin C.Berg
Assistant Professor
Mechanical Engineering Department, FU-10
University of Washington, Seattle, Washington, 98195
Telephone: 206-543-5288

Gregory S. Mason
Graduate Student
Mechanical Engineering Department, FU-10
University of Washington, Seattle, Washington, 98195
Telephone: 206-685-2429

Gen-Sheng Yang
Associate Scientist
Chung-Shan Institute of Science and Technology
P.O. Box 90008-15-15
Lung-Tan, Tao-Yuan
Taiwan, R.O.C.
Telephone: 886-2-712-8529

ABSTRACT

A new multirate sampled-data control law structure is defined and a new parameter-optimization-based synthesis algorithm for that structure is introduced. The synthesis algorithm can be applied to multirate, multiple-input multiple-output, sampled-data control laws having a prescribed dynamic order and structure, and apriori-specified sampling/update rates for all sensors, processor states, and control inputs. The synthesis algorithm is applied to design two-input, two-output tip position controllers of various dynamic orders for a sixth-order, two-link robot arm model.

KEY WORDS
Control, Multirate, Sampled-Data, Digital, Synthesis

RESEARCH SUPPORTED BY
NASA Langley Research Grant NAG-1-1055
The Government of Taiwan

PREFERRED ADDRESS FOR CORRESPONDENCE
Martin C. Berg
Mechanical Engineering Department, FU-10
University of Washington
Seattle, Washington 98195
Telephone: 206-543-5288

I. INTRODUCTION

Even in this age of fast, low-cost microprocessors there remain several important motivations for multirate sampling in sampled-data control systems. One need only consider large space structure control problems to realize that the cost, bulk, and weight of real-time computing hardware continues to be an important control system design issue. Multirate sampling provides the opportunity to allocate sampling rates, and thus real-time computing power, more efficiently. In two-time-scale control problems, for example, multirate sampling allows slow sampling in control loops associated with low-bandwidth control functions to be traded for fast sampling in those associated with high-bandwidth control functions.

As with microprocessors, the costs of analog-to-digital and digital-to-analog converters are also computation-rate dependent. Multirate sampling thus provides another opportunity to reduce hardware costs because the computation rates required of analog-to-digital and digital-to-analog converters frequently depend upon their sampling rates. Multirate sampling can even be used to reduce the total number analog-to-digital and/or digital-to-analog converters required by a system, by sample-dependent scheduling of multiple conversion tasks to a lesser number of conversion devices.

A third "motivation" for multirate sampling is becoming increasingly important: sometimes multirate sampling is the *only* choice. This situation can arise when an apriori decision has been made to include in a system a sensor that provides a discrete-time signal at a fixed sampling rate. A head position control system for a computer disk drive is a good example of such a system. The disk head, which is suspended atop the rotating disk, includes a sensor that reads the head position directly from certain diametrically-spaced segments on the disk. The sensor's sampling rate is thus *fixed* by the disk's rotation speed. To increase the control bandwidth beyond that dictated by that sampling rate, a second, faster-rate sensor must be added.

A key point often ignored by developers of multirate control law synthesis methods is that these motivations for multirate sampling dictate also certain flexibilities required to meet the needs of engineering practice. Specifically, multirate control law synthesis methods, to meet the needs of engineering practice, must allow the sampling rates for all sensors, the update rates for all processor

states, and the update rates for all actuators to be specified independently. The one generally accepted restriction, with regard to these rates, is that the ratio of all combinations of sampling and update rates must be rational, so that the complete sampling/update schedule will always be periodic. (We assume that all sampling and update events are synchronized to the same clock. The asynchronous case is treated elsewhere [1].)

Time lines representing such a periodic sampling schedule are shown in Fig. 1. We define the *Basic Time Period* (BTP) of such a schedule as the least common multiple of all of its sampling and update periods. The BTP is the period of repetition of the sampling/update schedule. We define the *Shortest Time Period* (STP) as the greatest common divisor of all of its sampling and update periods. We reserve the symbol P to represent the (integer) number of STP's per BTP, and we shall frequently use a double-indexing scheme for the independent variable so that, for example, $x(m,n)$ represents x at start of the $(n+1)$th STP of the $(m+1)$th BTP, for $m = 0,1, \ldots,$ and $n = 0, \ldots, P-1$.

There are five well-recognized methods for synthesizing multirate sampled-data control laws: successive loop closures, pole placement, the singular perturbation method, the LQG (linear quadratic Gaussian) method, and parameter optimization methods. Successive loop closures [2] is arguably the most important because it is the single one of the five that is widely used in industry. The advantages of successive loop closures are that its one-loop-at-a-time approach requires no new multirate synthesis techniques, and that the sampling/update rate for each control loop can be specified independently. The problem with successive loop closures is that its one-loop-at-a-time approach cannot fully account for all dynamic coupling between control loops.

Pole placement [3,4,5,6,7] for multirate systems has received considerable recent attention in the wake of reports on the capacity for periodically time-varying output feedback controllers to place closed-loop poles. In Ref. 3, for example, it is shown that given any controllable and observable continuous-time plant with m inputs, it is always possible to construct a periodically time-varying, pure-gain, output feedback control law that places the closed-loop poles arbitrarily, provided that the outputs are all sampled at a suitably chosen single sampling rate $1/T_0$, and that the inputs are updated at the rates $N_1/T_0, \ldots, N_m/T_0$, where the N_i are certain positive integers.

The problem with pole placement for multirate systems is the same as with pole placement for single-rate systems: how to determine where the closed-loop poles should be placed? It is a particularly difficult problem in the multirate case because multirate systems are periodically time varying [2,8]. The periodicity of multirate systems implies that their eigenstructure can only be defined based on their (time-invariant) BTP-to-BTP dynamics. Determining desirable closed-loop poles for a multirate system is typically difficult because the BTP of a multirate system is typically much longer than the characteristic times of many of its faster dynamics.

Singular perturbation control law synthesis methods [9,10,11,12,13,14,15] were first developed for continuous-time control systems to take advantage of the multiple-time-scale dynamics that often occur in control systems. It would seem that an extension to multirate sampled-data systems should follow naturally, given that a principal motivation for multirate sampling has always been to take advantage of those same multiple time scales, but that has not been the case in practice.

The problem is the singular perturbation method's inherant dependence on a coordinate transformation to separate the full control law synthesis problem into two (or more) dynamically decoupled control law synthesis problems of different time scales. Such a coordinate transformation is the first step in control law synthesis by the singular perturbation method. The state coordinates are easily decoupled because they represent only the plant's *internal* dynamics. The input and ouput coordinates cannot be so manipulated because they represent the plant's *external* sensor and actuator signals. Consequently, during the second control law synthesis step, when the control laws for the different-time-scale state vector components are synthesized separately, every control input vector element and every sensor ouput vector element remains coupled to every state coordinate so that, just as with successive loop closures, all dynamic coupling between control loops cannot be accounted for.

Various schemes have been developed to circumvent this difficulty. None have been completely successful. In Ref. 13, for example, a state feedback control law is synthesized by the singular perturbation method, and the lack of a completely decoupling transformation gives rise to a requirement for the slow component of the plant state vector to be estimated between slow-sampler updates, and a requirement for every control input to be updated at every sampling/update instant.

The advantage of the LQG method [2,16,17,18] for multirate sampled-data control law synthesis is that the control laws for all control loops are synthesized simultaneously, taking into account all dynamic coupling between control loops. The disadvantages are the same as with the LQG method for continuous-time control law synthesis: that practical performance and stability robustness objectives are often difficult to achieve via the minimization of a quadratic performance index, and that the resulting control laws are often unnecessarily complex. LQG control laws are even less desirable in the multirate as compared to the single-rate case because multirate Kalman filter and LQR state feedback gains are periodically time-varying [2]. In short, LQG multirate sampled-data control laws can provide a useful benchmark for performance comparisions, but they are not practical for applications.

Parameter optimization methods [2,19] for multirate sampled-data control law synthesis combine the principal advantages of the LQG and successive loop closures synthesis methods. They allow the synthesis of multirate sampled-data control laws of practical structure, and simultaneously account for all dynamic coupling between control loops. The typical parameter optimization method requires that the control law structure and its parameters to be optimized be prescribed. A numerical search is used to determine values for those parameters such that a performance index is minimized, possibly subject to constraints on those parameters. The disadvantage of parameter optimization methods is that they inevitably require a numerical search to determine the control law parameters.

A new parameter optimization method for synthesizing multirate sampled-data control laws is described in Sec. III of this paper. It is the second generation of the method described in Refs. 2 and 20. Unlike its predecessor, which accomodates only partial state feedback control laws, this new method accomodates a general, dynamic, multiple-input multiple-output control law structure. This new control lawstructure is described in Sec. II of this paper. Section IV describes an application of this new method to a design problem involving a two-link robot arm model. Conclusions are given in Sec. V.

II. CONTROL LAW STRUCTURE

This section describes the multirate sampled-data control law structure in Fig. 2. In Fig. 2, $\tilde{y}$ is the noise-free, continuous-time sensor signal, v is the discrete-time sensor noise signal, and $\tilde{u}$ is the continuous-time control signal. The one sampler in Fig. 2 operates at the sampling rate 1/T, where T is the STP of the system's complete sampling/update schedule. The Delay blocks are one-STP delays. The ZOH block is a zero-order hold.

The sensor sample-and-hold dynamics are represented by

$$\bar{y}(m,n+1) = [I - S_y(n)]\,\bar{y}(m,n) + S_y(n)\,y(m,n) \tag{1}$$

where $\bar{y}$ is the sensor signal hold state vector. The matrix $S_y(n)$ is the sensor switching matrix for the (n+ 1)th STP. We define a switching matrix as a diagonal matrix with 1 or 0 at every diagonal position. If the ith diagonal element of $S_y(n)$ is 1, the continuous-time signal from the ith sensor is sampled at the start of the (n+ 1)th STP of every BTP and that sampled value is immediately stored as the ith element of $\bar{y}$; otherwise, the same element of $\bar{y}$ is held at those instants. The key point is that $\bar{y}$ always contains the most recent sampled sensor data.

The processor dynamics are represented by

$$\begin{aligned} z(m,n+1) = {} & [I - S_z(n)]z(m,n) + S_z(n)\,\{A_z(n)\,z(m,n) \\ & + B_z(n)\,\{[I - S_y(n)]\bar{y}(m,n) + S_y(n)\,y(m,n)\}\} \end{aligned} \tag{2}$$

$$\begin{aligned} \hat{u}(m,n) = {} & C_z(n)\,z(m,n) \\ & + D_z(n)\,\{[I - S_y(n)]\,\bar{y}(m,n) + S_y(n)\,y(m,n)\} \end{aligned} \tag{3}$$

where z is the processor state vector, and $\hat{u}$ is the processor output vector. The matrix $S_z(n)$ is the processor state switching matrix. If the ith diagonal element of $S_z(n)$ is 1, the ith processor state is updated at the start of the (n+1)th STP of every BTP; otherwise, the same element of z is held at those instants. The matrices $A_z(n)$, $B_z(n)$, $C_z(n)$, and $D_z(n)$ are the processor state model matrices,

whose determination constitutes the control law synthesis problem. Note that a nonzero $D_z(n)$ results in direct feedthrough of sensor data to $\hat{u}(m,n)$.

The control signal update-and-hold dyamics are represented by

$$\bar{u}(m,n+1) = [I - S_u(n)]\,\bar{u}(m,n) + S_u(n)\,\hat{u}\,(m,n) \tag{4}$$

where $\bar{u}$ is the control signal hold state vector. The matrix $S_u(n)$ is the control signal switching matrix. If the ith diagonal element of $S_u(n)$ is 1, the ith element of $\bar{u}$ is updated at the start of the (n+ 1)th STP of every BTP; otherwise the same element of $\bar{u}$ is held at those instants.

Finally, the continuous-time control signal $\tilde{u}$ is generated by

$$\tilde{u}(t) = [I - S_u(n)]\,\bar{u}(m,n) + S_u(n)\,\hat{u}(m,n) \tag{5}$$

for all t on $[(mP + n)T, (mP + n + 1)T)$.

The advantage of the control law structure of (1) through (5) is that it can be used to represent virtually any sampled-data control law structure of practical interest. Its form, however, is not standard. Straightforward algebra, applied to (1) through (5), yields the following more standard form:

$$c(m,n+1) = A_c(n)\,c(m,n) + B_c(n)\,y(m,n) \tag{6}$$

$$u(m,n) = C_c(n)\,c(m,n) + D_c(n)\,y(m,n) \tag{7}$$

where

$$c(m,n) = [z(m,n)\;\; \bar{u}(m,n)\;\; \bar{y}(m,n)]^T \tag{8}$$

$$A_c(n) = \begin{bmatrix} [I - S_z(n)] + S_z(n)\, A_z(n) & S_z(n)\, B_z(n)\, [I - S_y(n)] & 0 \\ 0 & I - S_y(n) & 0 \\ S_u(n)\, C_z(n) & S_u(n)\, D_z(n)\, [I - S_y(n)] & I - S_u(n) \end{bmatrix} \tag{9}$$

$$B_c(n) = \begin{bmatrix} S_z(n)\, B_z(n)\, S_y(n) \\ S_y(n) \\ S_u(n)\, D_z(n)\, S_y(n) \end{bmatrix} \tag{10}$$

$$C_c(n) = [S_u(n)\, C_z(n) \quad S_u(n)\, D_z(n)\, [I - S_y(n)] \quad I - S_u(n)] \tag{11}$$

$$D_c(n) = [S_u(n)\, D_z(n)\, S_y(n)] \tag{12}$$

with

$$\tilde{u}(t) = u(m,n) \tag{13}$$

for all t on $[(mP + n)T, (mP + n + 1)T)$.

III. PARAMETER OPTIMIZATION METHOD

This section describes a parameter optimization control law synthesis method for the control law structure of Sec. II. It is a generalization of the similar method for state feedback control laws described in Refs. 2 and 20, and incorporates also the multiple-plant-condition design for robustness ideas of Ref. 21. The approach involves a numerical search to determine the processor matrices, $A_z(n)$, $B_z(n)$, $C_z(n)$, and $D_z z(n)$, for n=0, . . . , P–1, such that a quadratic performance index is minimized. That approach has been criticized in the past because of (1) the difficulties of achieving practical performance and stability robustness objectives via the minimization of a quadratic performance index, and (2) difficulties related to the convergence of the numerical search.

The proposed method addresses those criticisms in several ways. First, to enable synthesis for robustness to plant parameter variations, the performance index is defined over multiple plant conditions. This simple idea has been a key to the success of the popular Sandy [21,22,23,24,25,26]

algorithm for synthesizing robust continuous-time control laws. Second, to improve the convergence of the numerical search, the performance index and its gradients with respect to the control law parameters are calculated exactly, at every iteration, using closed-form expressions. Third, so that a stabilizing initial guess for the control law is not required, and to eliminate problems with destabilizing control laws encountered during the search, a finite-time performance index is used. Finally, to lessen the difficulties of achieving practical performance and stability robustness objectives via the minimization of a quadratic performance index, linear and nonlinear constraints can be imposed on the control law parameters.

The continuous-time plant dynamics at plant condition i are assumed to be represented by:

$$\dot{\tilde{p}}^{(i)}(t) = \tilde{A}_p^{(i)} \tilde{p}^{(i)}(t) + \tilde{B}_{pu}^{(i)} \tilde{u}^{(i)}(t) + \tilde{B}_{pw}^{(i)} \tilde{w}^{(i)}(t) \tag{14}$$

$$\tilde{y}^{(i)}(t) = \tilde{C}_p^{(i)} \tilde{p}^{(i)}(t) \tag{15}$$

where $\tilde{p}^{(i)}$ is the plant state vector, $\tilde{u}^{(i)}$ is the control input vector, $\tilde{y}^{(i)}$ is the sensor output vector, and $\tilde{w}^{(i)}$ is a stationary, zero mean, gaussian white noise input vector of known power spectral density.

The performance index is assumed to be

$$J = \sum_{i=1}^{N_p} E\left\{ \frac{1}{2t_f} \int_0^{t_f} \begin{bmatrix} \tilde{p}^{(i)}(t) \\ \tilde{u}^{(i)}(t) \end{bmatrix}^T \begin{bmatrix} \tilde{Q}^{(i)} & 0 \\ 0 & \tilde{R}^{(i)} \end{bmatrix} \begin{bmatrix} \tilde{p}^{(i)}(t) \\ \tilde{u}^{(i)}(t) \end{bmatrix} dt \right\} \tag{16}$$

where N_p is the number of plant conditions; E is the expected value operator; t_f is the final time and is a multiple of the BTP of the system's complete sampling/update schedule; and $\tilde{Q}^{(i)}$ and $\tilde{R}^{(i)}$ are the state and control weighting matrices for the ith plant condition and are non-negative definite matrices.

Based upon the description of the continuous-time plant dynamics in (14) and (15), a complete description of the complete system's sampling/update schedule, the performance index in (16), and the control law in (6) through (13), closed-form expressions for the performance index J and for its gradients with respect to the processor matrices $A_z(n)$, $B_z(n)$, $C_z(n)$, and $D_z(n)$, for n=0, . . . , P–1, are derived in Refs. 19 and 27. Those derivations and the resulting closed-form expressions are lengthy, and will not be repeated here. The key points are that the resulting expressions are *closed-form*, and that the number of computations required for their evaluation is ***independent*** of t_f. The single restriction for those expressions to be valid is that the state transition matrix for the BTP-to-BTP closed-loop system must be diagonalizable [19]. That is not a serious restriction because that matrix is rarely nondiagonalizable in practice.

Thus far nothing has been said about synthesizing other than periodically time-varying control laws. To that end, the performance index and gradient derivations in Refs. 19 and 27 assume that the processor matrices are constrained to satisfy

$$\begin{bmatrix} D_z(n) & C_z(n) \\ B_z(n) & A_z(n) \end{bmatrix} = \sum_{r=0}^{M-1} \alpha(n,r) \begin{bmatrix} \bar{D}_z(r) & \bar{C}_z(r) \\ \bar{B}_z(r) & \bar{A}_z(r) \end{bmatrix} \tag{17}$$

with $M \in \{1, \ldots, P\}$, and with the α functions constrained to satisfy

$$\alpha(n,p)\,\alpha(n,q) = \begin{cases} 1 & \text{if } p = q \\ 0 & \text{if } p \neq q \end{cases} \tag{18}$$

Equations (17) and (18) constrain the number of different sets of processor matrices to M. The function $\alpha(r,n)$ determines which set of processor matrices is active at the (n+ 1)th STP. Equation (18) guarantees that only one set of processor matrices is active per STP.

Based on the description of the continuous-time plant dynamics in (14) and (15), a complete description of the complete system's sampling/update schedule, the performance index in (16), the control law in (6) through (13), the constraint relations in (17) and (18), and the closed-form expressions for the performance index J, and for its gradients with respect to the processor matrices $\bar{A}_z(r)$, $\bar{B}_z(r)$, $\bar{C}_z(r)$, and $\bar{D}_z(r)$, for r=0, . . . , M–1, in Refs. 19 and 27, we have developed a computer

algorithm to numerically determine a set of processor matrices that minimizes J. A numerical search is used to determine the processor matrices $\bar{A}_z(r)$, $\bar{B}_z(r)$, $\bar{C}_z(r)$, and $\bar{D}_z(r)$, for $r = 0, \ldots, M-1$, given an initial guess for those matrices. The NPSOL nonlinear programming algorithm is used for the numerical search. NPSOL [28] is a powerful nonlinear programming package with good convergence properties as a result of its use of exact performance index and gradient evaluations at every iteration. In addition, NPSOL accomodates linear and/or nonlinear constraints on the independent variables. This means that linear and/or nonlinear constraints on the control law parameters can be combined with the usual performance index minimization objectives to achieve practical performance and stability robustness objectives.

Additional important features of this new synthesis algorithm include automatic discretization of the continuous-time plant model and of the continuous-time performance index [27]. These are important design features because they effectively decouple the sampling/update rates selection problem from the problem of determining a suitable performance index. This means that the performance index can be determined first, based on a continuous-time design, and that this new algorithm can then be used to determine a multirate sampled-data design that minimizes the same performance index.

In summary, the inputs required to apply this new synthesis algorithm are the following:

- A state model description of the continuous-time plant dynamics at each of the N_p plant conditions.

- State and control weighting matrices for the performance index at each of the N_p plant conditions.

- The final time t_f for the performance index.

- The power spectral density of the continuous-time white process noise at each of the N_p plant conditions.

- A complete description of the complete system's sampling/update schedule.

- The integer M and the α functions that constrain the periodicity of the processor matrices via (17) and (18).

- The desired dynamic order and structure for the processor matrices.

- The covariance matrix for the discrete-time sensor noise at each of the N_p plant conditions.

- A complete description of all linear and/or nonlinear constraints to be imposed on the elements of the processor matrices.

- An initial guess for the processor matrices.

A disadvantage of most parameter optimization control law synthesis methods is that they require a stabilizing initial guess for the control law. That is not the case with this method because of its finite-time performance index. The finite time ensures that the performance index and its gradients will be finite whether or not the closed-loop system is stable. A disadvantage of the finite time is that a steady-state solution, i.e., for

$$J_{ss} \triangleq \lim_{t_f \to \infty} J \tag{19}$$

cannot be obtained directly. A steady-state solution is easily obtained, in practice, however, by choosing a finite time t_f that is large compared to the characteristic times of all of the closed-loop system's poles. Because the number of computations required to evaluate the performance index and gradient expressions of Refs. 19 and 27 does not depend upon t_f, this can be done without penalty in terms of the computation time for the numerical search.

In practice, because digital computers cannot store arbitrarily large finite numbers, a steady-state solution usually cannot be obtained by simply initially setting t_f to a large value. Instead, it is usually necessary to complete first (i.e., when the current best guess for the control law parameters

is poor) an optimization for a small t_f, and to then re-optimize, for larger and larger t_f, until t_f gets to be large compared to the characteristic times of all closed-loop poles.

The final key issue regarding this new synthesis algorithm concerns the requirement that the structure of the processor matrices be specified. The key point is that the imposed structure should guarantee that the free parameters for the numerical search constitute an independent set with respect to the control law's input-output dynamics. When the processor dynamics in (2) and (3) are considered, with the constraints in (17) and (18) in effect, it is straightforward to see that the complete set of the elements of $\bar{A}_z(r)$, $\bar{B}_z(r)$, $\bar{C}_z(r)$ and $\bar{D}_z(r)$, for r = 0, . . . , M−1, do not constitute such an independent set because, for example, an arbitrary change in one element of $\bar{B}_z(0)$ can be compensated for by changes to the elemets of $\bar{C}_z(0)$, and to the other elements of $\bar{B}_z(0)$, such that the processor's input-output dynamics are unchanged.

Thus, additional structure, or, equivalently, additional constraints, must be imposed on the elements of the processor matrices to guarantee that the free parameters for the numerical search constitute an independent set with respect to the control law's input-output dynamics. In practice, a suitable set of such constraints can frequently be determined based on "classical" control law structures (e.g., combinations of lead and lag compensators and notch filters).

More generally structured control law can, of course, also be accomodated. What constitutes an optimal structure for the processor matrices for the general case is a topic of current research. We have successfully applied the following structure (shown for the n-is-even case) for the particular case where the constraints in (17) and (18) are applied with M = 1 (the time-invariant case):

$$\bar{A}_z(0) = \text{block diag} \left\{ \begin{bmatrix} 0 & 1 \\ -\sigma_i^2-\omega_i^2 & -2\sigma_i \end{bmatrix} \right\}, \; i = 1, \ldots, n/2 \tag{20}$$

$$\bar{B}_z(0) = \begin{bmatrix} b_{11} & \cdots & b_{1m} \\ & & \\ b_{n1} & \cdots & b_{nm} \end{bmatrix} \tag{21}$$

$$\bar{C}_z(0) = \begin{bmatrix} 1 & \cdots & 1 \\ c_{21} & & c_{2n} \\ & & \\ c_{p1} & \cdots & c_{pm} \end{bmatrix} \tag{22}$$

$$\bar{D}_z(0) = \begin{bmatrix} d_{11} & \cdots & d_{1m} \\ & & \\ d_{p1} & \cdots & d_{pm} \end{bmatrix} \tag{23}$$

The appendix shows that the $(m+p)n+mp$ σ_i, ω_i, b_{ij}, c_{ij} and d_{ij} parameters of this structure constitute an independent set with respect to the control law's input-output dynamics (for any non-trivial sampling/update schedule) provided that no eigenvalues $\sigma_i \pm j\omega_i$ of the processor dynamics are repeated.

IV. EXAMPLES

This section describes the design of a tip position control system for a planar two-link robot arm. The robot arm system is shown in Fig. 3. The first link is long and massive, for large-scale slewing motions. The second is relatively short and lightweight, so that high-bandwidth control of the arm's tip position can be achieved using a relatively small motor at the second joint. The pin joint, rotational spring and rotational damper at the midpoint of the first link models flexibility in that link. The second link is assumed to be rigid. The motor torques T_1 and T_2 are the control inputs, and it is assumed that only the joint angle θ and the tip position δ are measured. The linearized dynamical equations for this system for small $\varepsilon - \theta$ and small $\phi - \varepsilon$ are easily derived. The spring constant (k) and damping coefficient (b) values (in Fig. 3) were chosen based on that model to achieve 1 percent damping and a 10 Hz natural frequency for the open-loop vibration mode.

Figures 4 through 6 show the closed-loop arm responses, based on the linearized arm dynamics, to a step change in the commanded tip position with nine different control laws. The tip position (δ) responses are shown in Fig. 4. The simultaneous control torque (T_1 and T_2) responses are shown in Figs. 5 and 6. The nine different control laws are briefly described as follows:

LQR Analog: The continuous-time LQR (full-state-feedback) control law that minimizes

$$J = \lim_{t_f \to \infty} \frac{1}{2t_f} \int_0^{t_f} \beta^2[\theta^2 + (\rho\delta)^2] + \left(\frac{T_1}{T_{1max}}\right)^2 + \left(\frac{T_2}{T_{2max}}\right)^2 dt \tag{24}$$

with

$$\beta = 32 \text{ rad}^{-1} \tag{25}$$

$$\rho = 14 \text{ rad/m} \tag{26}$$

$$T_{2max} = 0.00335 \text{ N·m} \tag{27}$$

$$T_{1max}/T_{2max} = 8 \tag{28}$$

The T_{2max} value is the T_2 torque that achieves $\dot{\phi} = 2\pi$ rad/sec in 1 sec with $\theta(t) \equiv \varepsilon(t) \equiv 0$ and $\dot{\phi}(0) = 0$. The T_{1max}/T_{2max} value represents a typical ratio of peak motor torques at the respective joints. The β and ρ values were chosen by trial and error to achieve the closed-loop poles in Table 1. Note that the ratio of the characteristic frequencies of the rigid-body closed-loop pole pairs is eight; and that the characteristic frequency of the faster closed-loop rigid-body pole pair is a factor of five less than the characteristic frequency of the closed-loop vibration mode.

Third-Order Analog Successive Loop Closures: The third-order, continuous-time, successive loop closures control law in Fig. 7, which consists of a single lead compensator in the θ-to-T_1 loop and twin, cascaded lead compensators in the δ-to-T_2 loop. The closed-loop poles for this design are in Table 2. Note that the rigid body and vibration mode closed-loop poles match those of the LQR Analog design.

Third-Order Multirate Tustin: A multirate sampled-data approximation to the Third-Order Analog Successive Loop Closures design obtained via Tustin's approximations of the continuous-time transfer functions in Fig. 7. The sampling/update rates (in samples/updates per second) of the

θ-to-T_1 and δ-to-T_2 loops are eight times the characteristic frequencies (in cycles per second) of the slow and fast, respectively, rigid body closed-loop pole pairs from the Third-Order Analog Successive Loop Closures design.

Optimized Third-Order Multirate Tustin: The same as the Third-Order Multirate Tustin design, but with the lead compensator gain, zero, and pole locations optimized, by the parameter optimization control law synthesis method of Sec. III, to minimize the same performance index as in the LQR Analog design. To synthesize this control law, continuous-time process noise and discrete-time sensor noise inputs were added to the robot arm model. The former were taken to be white noise disturbance torques w_1 and w_2, coincident with the respective control torques, with

$$E\left\{\begin{bmatrix} w_1(t) \\ w_2(t) \end{bmatrix} [w_1(\tau) \;\; w_2(\tau)]\right\} = \begin{bmatrix} 4.9\text{x}10^{-5} & 0 \\ 0 & 1.6\text{x}10^{-5} \end{bmatrix} \delta(t-\tau) \tag{29}$$

where δ is the Dirac delta function. The latter were taken to be stationary, purely random sequences, v_1 and v_2, for the θ and δ measurements, respectively, with

$$E\left\{\begin{bmatrix} v_1(m,n) \\ v_2(m,n) \end{bmatrix} [v_1(m,n) \;\; v_2(m,n)]\right\} = \begin{bmatrix} 8.1\text{x}10^{-5} & 0 \\ 0 & 1\text{x}10^{-4} \end{bmatrix} \tag{30}$$

Multirate Third-Order: The same as the Optimized Third-Order Multirate Tustin design, but using the third-order, generalized, time-invariant structure in (20) through (23) for the processor matrices. Just as with the Optimized Third-Order Multirate Tustin design, two of the processor states are updated at the faster sampling/update rate, and the third is updated at the slower sampling/update rate.

Multirate Second-Order: The same as the Multirate Third-Order design, but using the second-order, generalized, time-invariant structure in (20) through (23) for the processor matrices. One of the processor states is updated at the faster sampling/update rate, and the other is updated at the slower sampling/update rate.

Multirate First-Order: The same as the Multirate Third-Order design, but using the first-order, generalized, time-invariant structure of (20) through (23) for the processor matrices. The one processor state is updated at the faster sampling/update rate.

Single-Rate Third Order: The same as the Multirate Third-Order design, but single-rate, with the single sampling/update rate chosen to yield the same number of real-time computations per unit time as the Multirate Third-Order design.

Analog Third-Order: The continuous-time equivalent to the Multirate and Single-Rate Third-Order designs. The processor matrices have the same structure as in the Multirate and Single-Rate Third-Order designs. The control law was synthesized using the Sandy algorithm [21] to minimize the same performance index as in the Multirate and Single-Rate Third-Order designs.

The LQR Analog responses in Figs. 4a, 5a and 6a constitute the optimal responses for the performance index in (24), assuming full state feedback, no process or sensor noise, and infinitely fast sampling. The Third-Order Analog Successive Loop Closures responses in the same figures have low tip position overshoot, but include also a relatively large contribution from the vibration mode (see especially Figs. 5a and 6a).

The Third-Order Multirate Tustin responses in Figs. 4a, 5a and 6 are unacceptable. This is somewhat suprising, but not totally unexpected given the (low) factor-of-eight sampling/update rate-to-characteristic frequency ratio for this design.

The Optimized Third-Order Multirate Tustin responses in Figs. 4a, 5a and 6a are acceptable, and demonstrate that the parameter optimization control law synthesis algorithm of Sec. III can be used to optimize the parameters of classically-structured control laws.

The Multirate Third-Order, Second-Order and First-Order responses in Figs. 4b, 5b and 6b demonstrate that the same parameter optimization control law synthesis algorithm can be used to synthesize multirate sampled-data control laws having a prescribed dynamic order and a prescribed, but general, structure, with apriori specified sampling/update rates for all sensors, processor states, and control inputs.

The Single-Rate Third-Order and Analog Third-Order responses in the same figures put the multirate responses in perspective. The Single-Rate Third-Order control law is the single-rate equivalent to the Multirate Third-Order control law because it (1) was synthesized to minimize the same performance index, using the same process and sensor noise characteristics; and (2) requires the same number of computations per unit time for real-time operation.

The Analog Third-Order responses in the same figures are the responses that would have been obtained with the Multirate Third-Order control law and Singe-Rate Third-Order control laws if sampling and update rates were not an issue, and very fast sampling and update rates were everywhere used.

V. CONCLUSIONS

With the possible exception of successive loop closures, the multirate sampled-data control law synthesis methods available today fail to provide the designer with sufficient flexibility to prescribe sensor sampling rates and processor state and control input update rates. A new parameter-optimization-based method for synthesizing multirate sampled-data control laws of arbitrary dynamic order that provides that flexibility is described in this paper. This new method, described in Sec. III, determines, by numerical optimization, the free parameters of the general purpose multiple-input, multiple-output, sampled-data control law structure in Fig. 2, to minimize a quadratic performance index, possibly subject to linear and/or nonlinear constraints on those parameters. A stabilizing initial guess for the control law is not required because the performance index is finite-time. To enable the synthesis of robust control laws, the performance index can be defined over multiple plant conditions.

An application of this new method to the design of a tip position control system for a sixth-order, two-link robot arm was described. Multirate sampled-data control laws of various dynamic orders synthesized by various methods were compared to confirm that the new synthesis method can be used to synthesize multirate sampled-data control laws having a prescribed dynamic order and structure, with apriori specified sampling/update rates for all sensors, processor states, and control inputs.

REFERENCES

1. Ritchey, V.S. and Franklin, G.F., "A Stability Criterion for Asynchronous Multirate Linear Systems," *IEEE Trans. Auto. Contr.*, Vol. 34, May 1989, pp. 529-535.

2. Berg, M.C., Amit N. and Powell, J.D., "Multirate Digital Control System Design," *IEEE Trans. Auto. Contr.*, Vol. AC-33, Dec. 1988, pp. 1139-1150.

3. Araki, M. and Hagiwara, T., "Pole Assignment by Multirate Sampled-Data Output Feedback," *Int. Jour. Contr.*, Vol. 44, 1986, pp. 1661-1673.

4. Chammas, A. and Leondes, C, "Pole Assignment by Piecewise Constant Output Feedback," *Int. Jour. Contr.*, Vol. 29, 1979, pp. 31-38.

5. Hagiwara, T. and Araki, M, "Design of a Stable State Feedback Controller Based on the Multirate Sampling of the Plant Output," *IEEE Trans. Auto. Contr.*, Vol. 33, Sep. 1988, pp. 812-819.

6. Hernandez, V. and Urbano A., "Pole-Assignment for Discrete-Time Linear Periodic Systems," *Int. Jour. Contr.*, Vol. 46, 1987, pp. 687-697.

7. Patel, Y. and Patton, R.J., "A Robust Approach to Multirate Controller Design Using Eigenstructure Assignment," *Proc. Amer. Contr. Conf.*, San Diego, CA, May 1990, pp. 945-951.

8. Araki, M. and Yamamoto, K., "Multivariable Multirate Sampled-Data Systems: State-Space Description, Transfer Characteristics, and Nyquist Criterion," *IEEE Trans. Auto. Contr.*, Vol. AC-31, 1986, pp. 145-154.

9. Saksena, V.R., O'Reilly, J. and Kokotovic, P.V., "Singular Perturbations and Time-Scale Methods in Control Theory: Survey 1976-1983," *Automatica*, Vol. 20, 1984, pp. 273-293.

10. Calise, A.J., Prasad, J.V.R. and Siciliano, B., "Design of Optimal Output Feedback Compensators in Two-Time Scale Systems," *IEEE Trans. Auto. Contr.*, Vol. 35, Apr 1990, pp. 488-492.

11. Kando, H. and Iwazumi, T., "Multirate Digital Control Design of an Optimal Regulator via Singular Perturbation Theory," *Int. Jour. Contr.*, Vol. 44, 1986, pp. 1555-1578.

12. Lennartson, B., "Multirate Sampled-Data Control of Two-Time-Scale Systems," *IEEE Trans. Aut. Contr.*, Vol. 34, 1989, pp. 642-644.

13. Litkouhi, B. and Khalil, H., "Multirate and Composite Control of Two-Time-Scale Discrete-Time Systems," *IEEE Trans. Aut. Contr.*, Vol. AC-30, 1985, pp. 645-651.

14. Litkouhi, B. and Khalil, H., "Infinite-Time Regulators for Singularly Perturbed Difference Equations," *Int. Jour. Contr.*, Vol. 39, 1984, pp. 587-598.

15. Naidu, D.S. and Price, D.B., "Time-Scale Synthesis of a Closed-Loop Discrete Optimal Control System," *AIAA Jour. Guid.* Contr., Vol. 10, 1987, pp. 417-421.

16. Amit, N., "Optimal Control of Multirate Digital Control Systems," PhD Thesis, Stanford Univ., Stanford, CA, 1980.

17. Glasson, D.P., "A New Technique for Multirate Digital Control Design and Sample Rate Selection," *AIAA Jour. Guid. Contr.*, Vol. 5, Aug. 1982, pp. 379-382.

18. Al-Rahmani, H.M. and Franklin, G.F., "A New Optimal Multirate Control of Linear Periodic and Time-Invariant Systems," *IEEE Trans. Auto. Contr.*, Vol. 35, Apr 1990, pp. 406-415.

19. Berg, M.C. and Yang, G.S., "A New Algorithm for Multirate Digital Control Law Synthesis," *Proc. IEEE Conf. Decision Contr.*, Dec. 1988, Austin, TX, pp. 1685-1690.

20. Berg, M.C., "Design of Multirate Digital Control Systems," PhD Thesis, Stanford Univ., Stanford, CA, 1986.

21. Ly, U.L., "A Design Algorithm for Robust Low-Order Controllers," PhD Thesis, Stanford Univ., Stanford, CA, 1982.

22. Ly, U.L., "Robust Control Design Using Nonlinear Constrained Optimization," *Proc. Amer. Contr. Conf.*, San Diego, CA, May 1990, pp. 968-969.

23. Voth, C.T. and Ly, U.L., "Total Energy Control System Autopilot Design with Constrained Optimization," *Proc. Amer. Contr. Conf.*, San Diego, CA, May 1990, pp. 1332-1337.

24. Gangsaas, D., Bruce, K.R., Blight, J.D., and Ly, U.L., "Application of Modern Synthesis to Aircraft Control: Three Case Studies," *IEEE Trans. Auto. Contr.*, Vol. AC-31, Nov. 1986, pp. 995-1014.

25. Bossi, J.A., Jones, R.D. and Ly, U.L., "Multivariable Regulator Design for Robustness and Performance: A Realistic Example," *Proc. Amer. Contr. Conf.*, Seattle, WA, 1986, pp. 285-288.

26. Ly, U.L., "Optimal Low-Order Flight Critical Pitch Augmentation Control Law for a Transport Airplane," *Proc. AIAA Guid. Contr. Conf.*, 1984, pp. 743-757.

27. Yang, G.S., "A Generalized Synthesis Method for Multirate Feedback Control Systems," PhD Thesis, Univ. of Washington, Seattle, WA, 1988.

28. Gill, P.E., Saunders, M.A. and Wright, M.H., "User's Guide for NPSOL," Tech. Rep. SOL 86-2, Dept. Operations Research, Stanford Univ., Stanford, CA, 1986.

APPENDIX A. SUPPORTING MATHEMATICS

Consider the control law in (1) through (5). Suppose that the constraints in (17) and (18), with M=1, are in effect, so that the processor matrices are constrained to be time-invariant. Suppose that the processor matrices, $\bar{A}_z(0)$, $\bar{B}_z(0)$, $\bar{C}_z(0)$ and $\bar{D}_z(0)$, are further constrained to have the forms in (20) through (23). Finally, to guarantee a nontrivial sampling/update schedule, suppose that the sensor, processor state, and actuator switching matrices satisfy

$$\det\left[\sum_{n=0}^{P-1} S_y(n)\right] \neq 0 \tag{31}$$

$$\det\left[\sum_{n=0}^{P-1} S_z(n)\right] \neq 0 \tag{32}$$

$$\det\left[\sum_{n=0}^{P-1} S_u(n)\right] \neq 0 \tag{33}$$

We will show that the (m+p)n+pm σ_i, ω, b_{ij}, c_{ij} and d_{ij} elements of the control law then constitute an independent set with respect to that control law's input-output dynamics if and only if $\bar{A}_z(0)$ has no repeated eigenvalues.

We begin by noting that, with (31), (32) and (33) in effect, it is straightforward to see that the independence in question does not depend whatsoever on the sensor, processor state, or actuator switching matrices. Therefore we consider only the special case where $S_y(n)$, $S_z(n)$ and $S_u(n)$, for n=0, . . . P–1, are identity matrices. The control law then reduces to

$$z(m,n+1) = \bar{A}_z(0)\, z(m,n) + \bar{B}_z(0)\, y(m,n) \tag{34}$$

$$u(m,n) = \bar{C}_z(0)\, z(m,n) + \bar{D}_z(0)\, y(m,n) \tag{35}$$

where u(m,n) is defined in (13).

Consider first the control law

$$z(k+1) = A\, z(k) + B y(k) \tag{36}$$

$$u(k) = C\, z(k) + D\, y(k) \tag{37}$$

where

$$z = \begin{bmatrix} z_1 \\ \vdots \\ z_n \end{bmatrix} \qquad y = \begin{bmatrix} y_1 \\ \vdots \\ y_m \end{bmatrix} \qquad u = \begin{bmatrix} u_1 \\ \vdots \\ u_p \end{bmatrix} \tag{38}$$

$$A = \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \lambda_n \end{bmatrix} \tag{39}$$

and B, C and D have the forms in (21) through (23). So that the control law's impulse response will be purely real, the λ_i's and the corresponding columns of C and rows of B must be either real, or must occur in complex-conjugate pairs, and D must be purely real.

Lemma: The $(m+p)n+pm$ λ_i, b_{ij}, c_{ij} and d_{ij} parameters (counting a real element as one parameter, and a complex conjugate pair of elements as two parameters) of the control law in (36) and (37) constitute an independent set with respect to that control law's input-output dynamics if and only if $\lambda_i \neq \lambda_j$ for $i \neq j$.

Proof: Consider the related control law

$$z(k+1) = A\, z(k) + B\, y(k) \tag{40}$$

$$u(k) = \bar{C}\, z(k) \tag{41}$$

with

$$\bar{C} = \begin{bmatrix} \bar{c}_{11} & \cdots & \bar{c}_{1n} \\ & & \\ \bar{c}_{p1} & \cdots & \bar{c}_{pn} \end{bmatrix} \tag{42}$$

Its input-output dynamics are represented by

$$\bar{H}(z) = \bar{C}\,(z\,I - A)^{-1}\,B = \sum_{i=1}^{n} \frac{\bar{C}_i\,B_i}{z-\lambda_i} \tag{43}$$

where $\bar{C}_i$ is the i th column of $\bar{C}$, and B_i is the *i*th row of B. The λ_i, b_{ij} and $\bar{c}_{ij}$ parameters of this control law are dependent with respect the control law's input-output dynamics if and only if, for an arbitrary change in one, the others can be changed so that $\bar{H}(z)$ is unchanged.

Case 1: No repeated λ_i 's.

From (41), for the case of no repeated λ_i 's, it is straightforward to see that when one of the λ_i's is changed by an arbitrary amount, it will not be possible to change the remaining λ_i, b_{ij} and $\bar{c}_{ij}$ elements so that $\bar{H}(z)$ is unchanged. But when b_{ij} is multiplied by a nonzero but otherwise arbitrary α, we can multiply the remaining elements of B_i by α, and divide $\bar{C}_j$ by α, so that $\bar{H}(z)$ is unchanged. Thus, the λ_i, b_{ij} and $\bar{c}_{ij}$ parameters of the control law in (40) and (41) are dependent with respect to that control law's input-output dynamics.

If, however, one element of every column of $\bar{C}$ is fixed, as is the case in the C matrix of (22), it is similarly straightforward to see that it will not be possible to compensate for an arbitrary change in any λ_i, b_{ij} or $\bar{c}_{ij}$ element by changing the remaining λ_i, b_{ij} and $\bar{c}_{ij}$ elements so that $\bar{H}(z)$ is unchanged. Thus, for the case of no repeated λ_i's, with one element of every column of $\bar{C}$ fixed, the λ_i, b_{ij} and $\bar{c}_{ij}$ parameters of the control law in (40) and (41) constitute an independent set with respect to that control law's input-output dynamics.

Case 2: Repeated λ_i's.

Consider again the control law in (40) and (41), but this time suppose that $\lambda_1=\lambda_2$. That control law's input-output dynamics are represented by

$$\bar{H}(z) = \bar{C}\,(z\,I - A)^{-1}\,B = \frac{\bar{C}_1\,B_1 + \bar{C}_2\,B_2}{z-\lambda_1} + \sum_{i=3}^{n} \frac{\bar{C}_i\,B_i}{z-\lambda_i} \tag{44}$$

From (42), it is straightforward to see that, with or without one element of every column of $\bar{C}$ fixed, the remaining b_{ij} and $\bar{c}_{ij}$ elements of $\bar{C}_1$, B_1, $\bar{C}_2$, and B_2 can be changed to compensate for an arbitrary change in any one element of $\bar{C}_1$, B_1, $\bar{C}_2$, or B_2 so that $\bar{H}(z)$ is unchanged. Thus, for the case of repeated λ_i's, with or without one element of every column of $\bar{C}$ fixed, the λ_i, b_{ij} and $\bar{c}_{ij}$ parameters of the control law of (40) and (41) are dependent with respect to that control law's input-output dynamics.

General Case: We conclude that the (m+p)n λ_i, b_{ij}, and c_{ij} parameters of the control law in (36) and (37), with D= 0, constitute an independent set with respect to that control law's input-output dynamics if and only if $\lambda_i \neq \lambda_j$, for i≠j. A nonzero D matrix simply adds *pm* parameters to that set.

Theorem: For the control law in (1) through (5); with the constraints in (17) and (18), with M=1, in effect, so that the processor matrices are constrained to be time-invariant; with $\bar{A}_z(0)$, $\bar{B}_z(0)$, $\bar{C}_z(0)$ and $\bar{D}_z(0)$ further constrained to have the forms in (20) through (23); and assuming that the sensor, processor state, and actuator switching matrices satisfy (31) through (33); the (m+p)n+pm σ_i, ω_i, b_{ij}, c_{ij} and d_{ij} elements of that control law constitute an independent set with respect to that control law's input-output dynamics if and only if $\bar{A}_z(0)$ has no repeated eigenvalues.

Proof: The control law in (34) and (35) has the same number of free parameters as the control law in (36) and (37), and the two are related by the similarity transformation x= M z , where

$$M = \text{block diag}\ \{\ \frac{1}{2j\,\omega_i}\begin{bmatrix} \sigma_i+j\omega_i & 1 \\ -\sigma_i+j\omega_i & -1 \end{bmatrix},\ i = 1, \ldots, n/2\ \} \tag{45}$$

with $\sigma_i = \text{Re}(\lambda_i)$ and $\omega_i = \text{Im}(\lambda_i)$.

	Closed-Loop Pole	Damping Ratio	Characteristic Frequency
Rigid Body	$-1.10 \pm j\,1.10$	0.71	0.25 Hz
Rigid Body	$-8.81 \pm j\,8.83$	0.71	2.0 Hz
Vibration Mode	$-0.649 \pm j\,62.8$	0.01	10 Hz

Table 1 LQR Analog Design Closed-Loop Poles

	Closed-Loop Pole	Damping Ratio	Characteristic Frequency
Rigid Body	$-1.10 \pm j\,1.11$	0.71	0.25 Hz
Rigid Body	$-8.88 \pm j\,8.84$	0.71	2.0 Hz
Vibration Mode	$-1.35 \pm j\,63.9$	0.02	10 Hz
Compensator	-10.5	–	1.7 Hz
Compensator	$-33.2 \pm j\,34.0$	0.70	7.6 Hz

Table 2 Third-Order Analog Successive Loop Closures Design Closed-Loop Poles

Time Lines for Sampling/Update Activities:

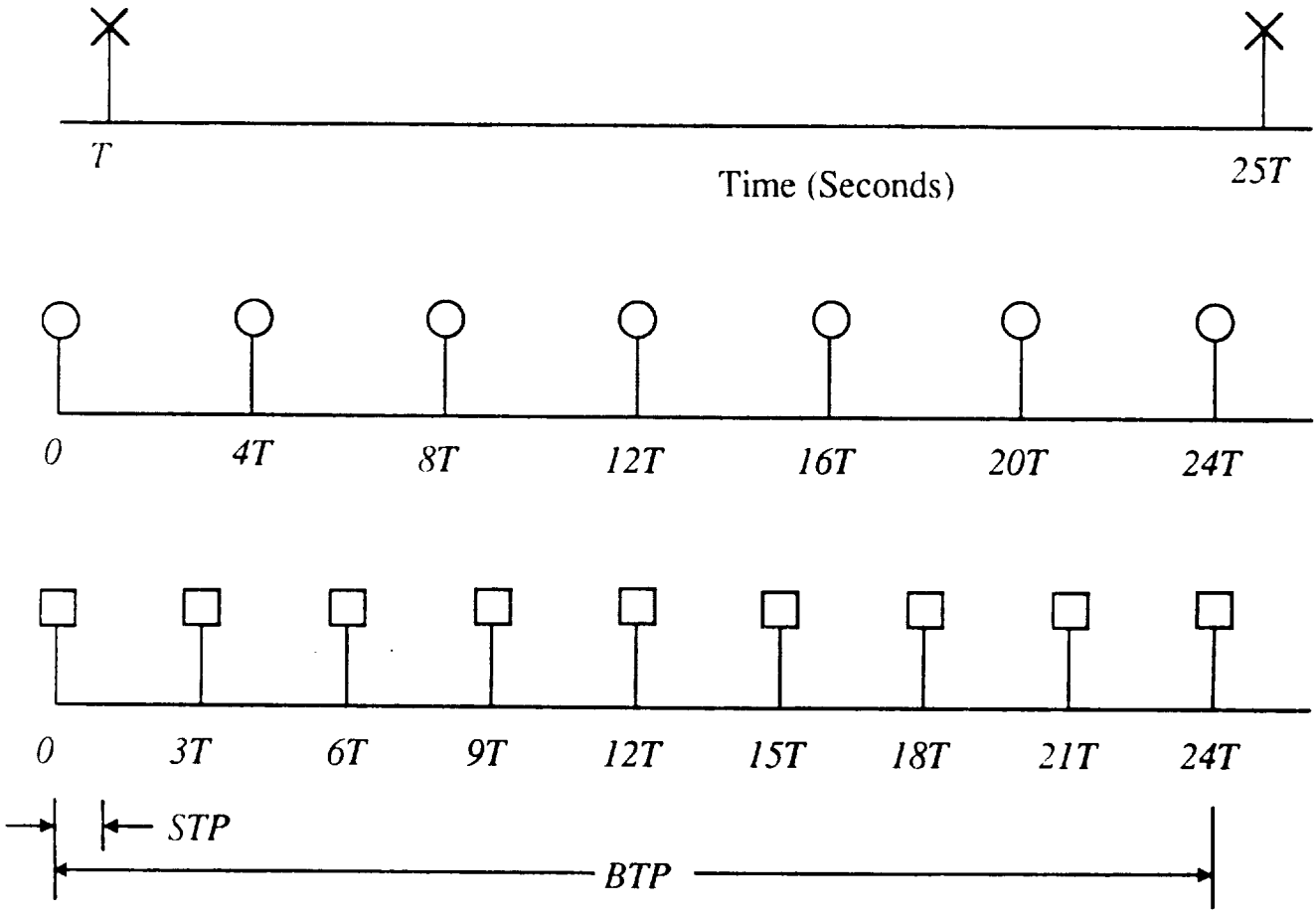


Fig. 1 Example Multirate Sampling/Update Schedule

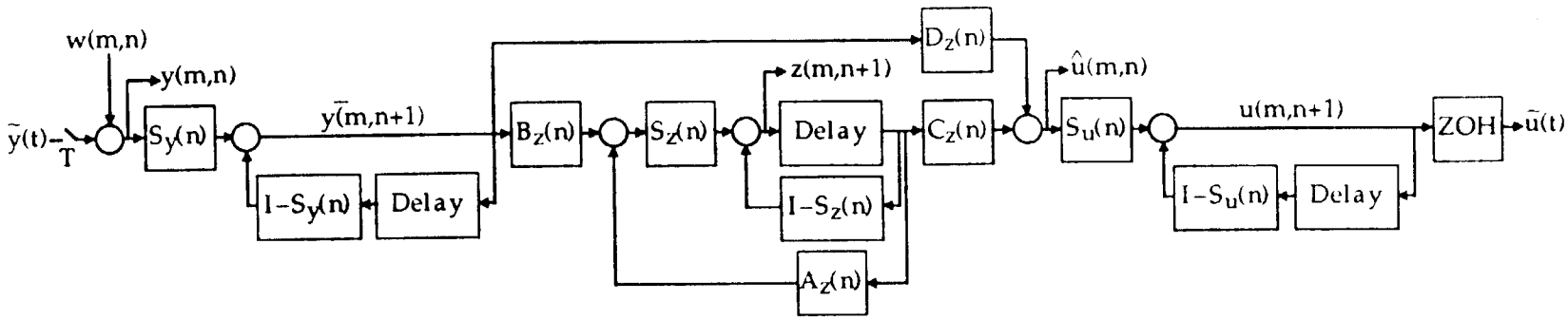
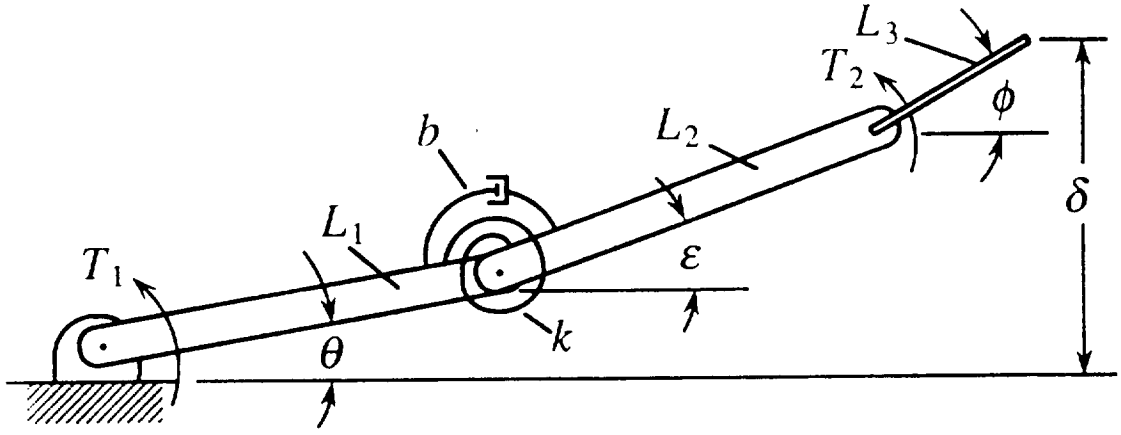


Fig. 2 Multirate Sampled-Data Control Law Structure



Parameters:	Mass	Length	
L_1	0.5 kg	0.5 m	
L_2	0.5 kg	0.5 m	$k = 37.33$ N/rad
L_3	0.04 kg	0.2 m	$b = 0.012$ N· s/m

The natural frequency of the vibration mode is 10 hz.

Inputs: Torques T_1 and T_2

Outputs: θ and δ

Fig. 3 Two-Link Robot Arm

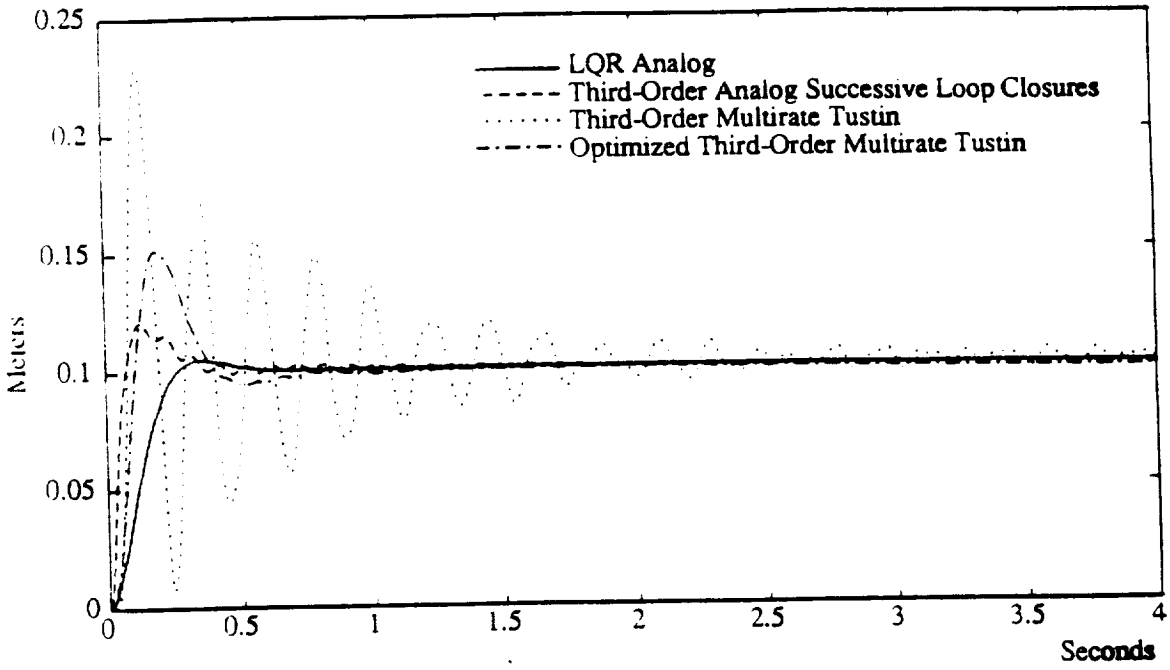


Fig. 4a Tip Position (δ) Responses to Tip Position Step Command

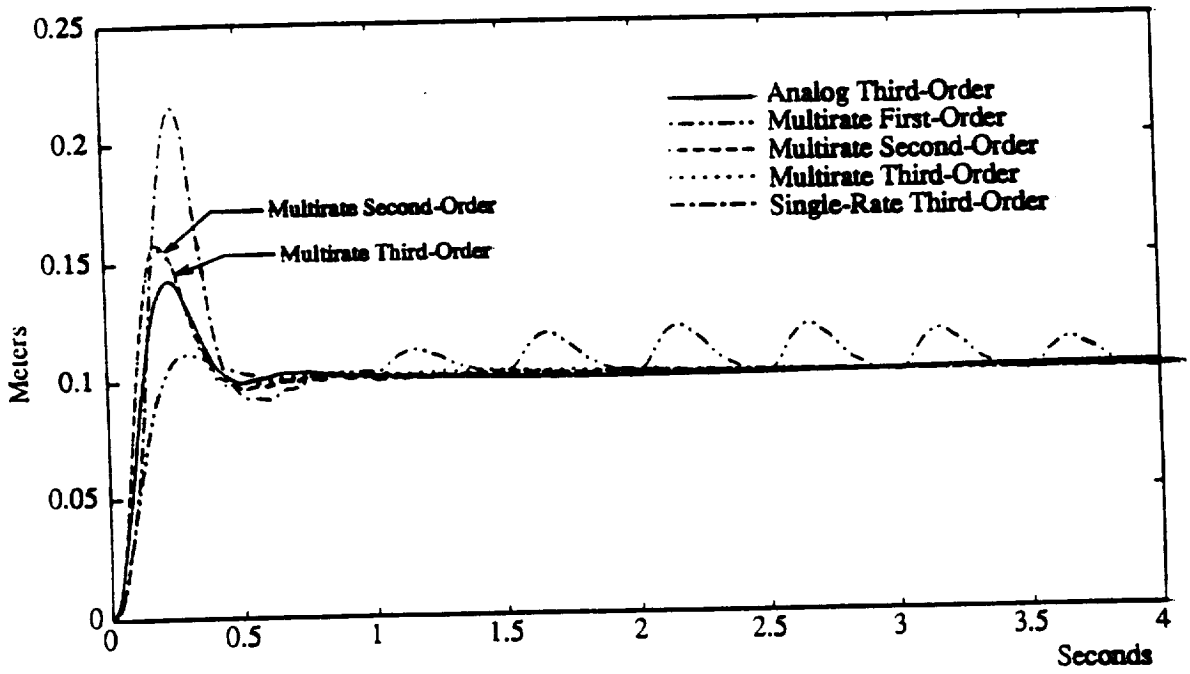


Fig. 4b Tip Position (δ) Responses to Tip Position Step Command

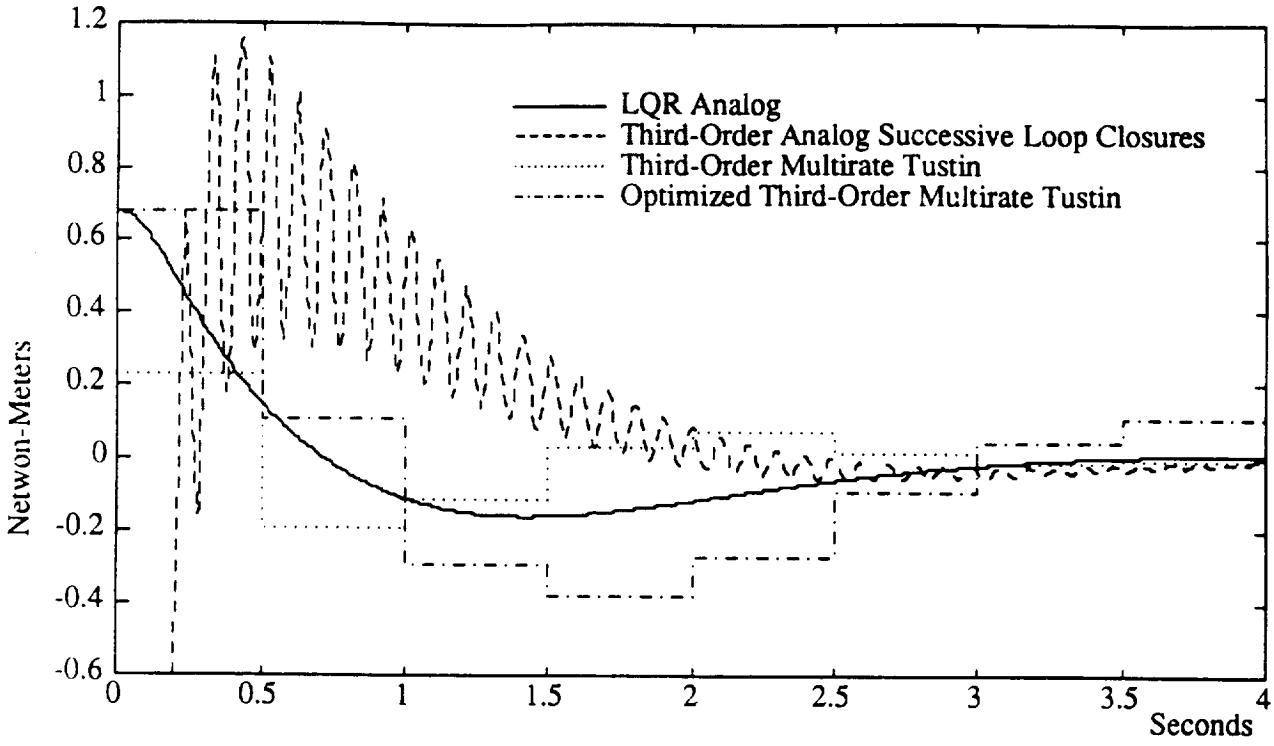


Fig. 5a Control Torque T_1 Responses to Tip Position Step Command

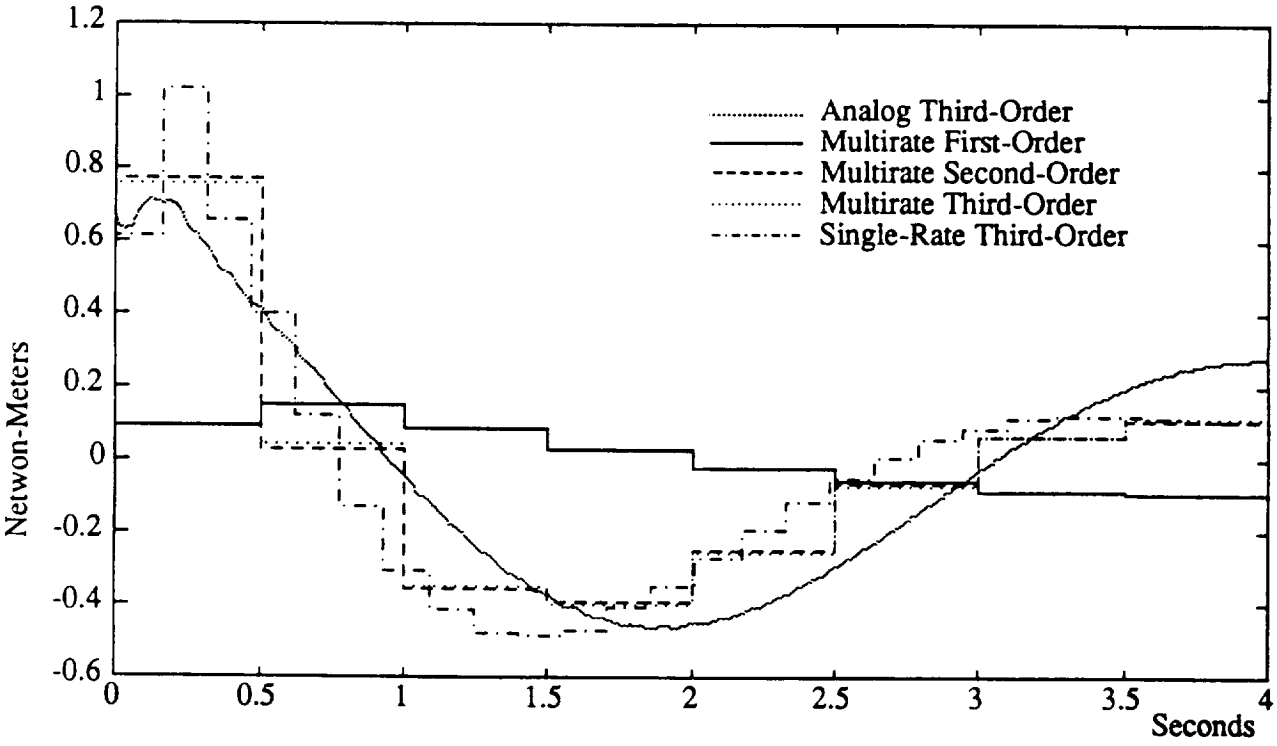


Fig. 5b Control Torque T_1 Responses to Tip Position Step Command

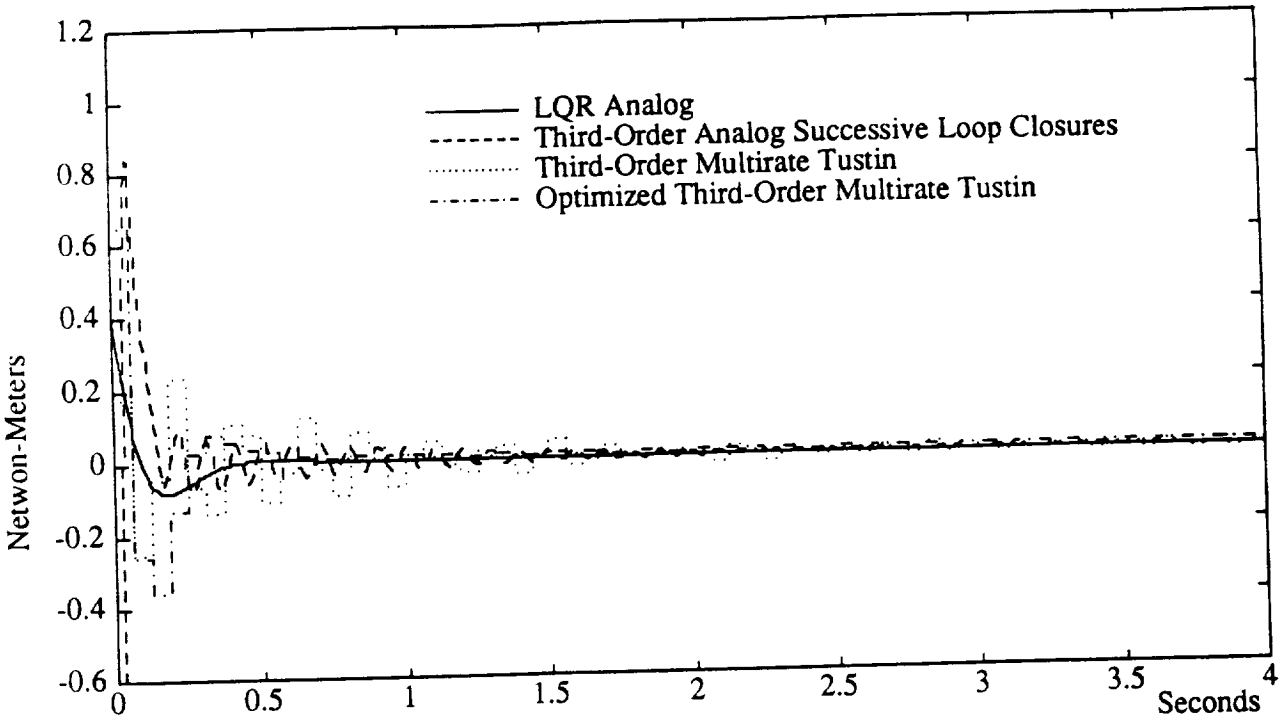


Fig. 6a Control Torque T_2 Responses to Tip Position Step Command

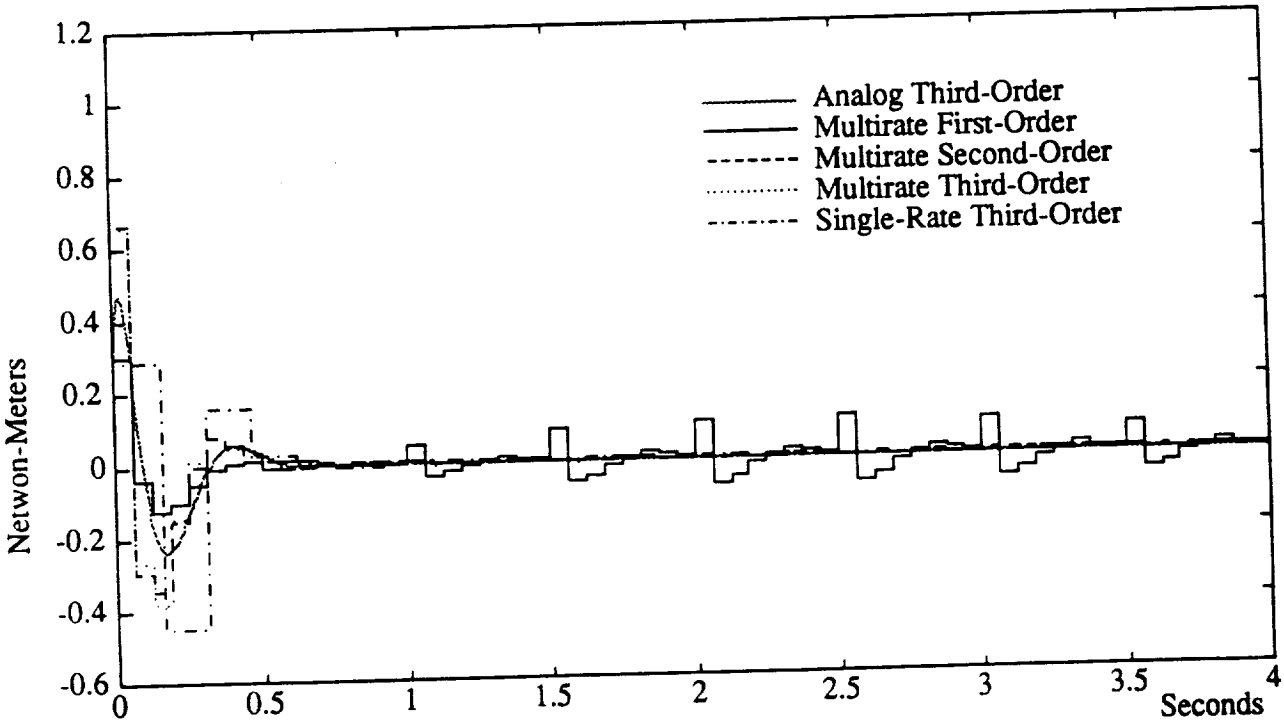


Fig. 6b Control Torque T_2 Responses to Tip Position Step Command

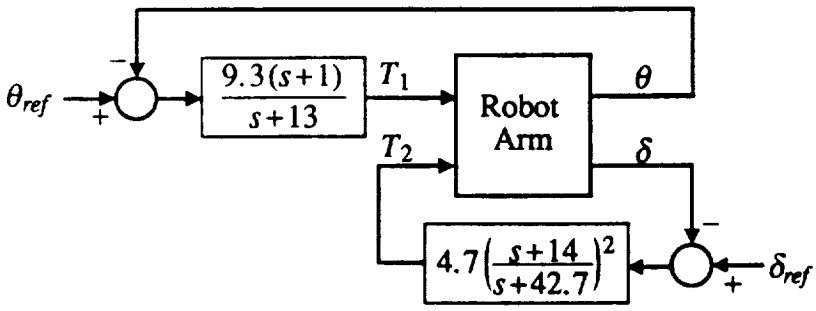


Fig. 7 Third-Order Analog Successive Loop Closures Design

APPENDIX B

REPRINT OF REFERENCE 5

Reduced Order Multirate Compensator Synthesis

Gregory S. Mason* and Martin C. Berg**

University of Washington

ABSTRACT

A method for synthesizing reduced order multirate compensators is presented. Necessary conditions for which the compensator parameter values minimize an infinite time quadratic cost function are derived. An algorithm for finding compensator parameter values which satisfy the necessary conditions is described. This algorithm is then used to design several tip position controllers for a two link robot arm.

INTRODUCTION

In many cases a multirate compensator can provide better performance than a single rate compensator requiring the same number of real-time computations. Berg, for example, was able to reduce the steady state RMS response of states and controls to a disturbance for a simple mass-spring-mass system nearly 20% by using a multirate compensator over a single rate compensator [1]. Numerous other examples have been provided in the literature by Berg [1]-[3], Amit [4]-[5], and Yang [6]. While multirate compensators can provide improved performance over single rate compensators, they are also, in general, more complicated to design.

The complexity of multirate compensators stems from the fact that they are by nature time varying, periodically time varying for most practical applications. Not only must designers choose multiple sampling/update rates for the compensator, but they must also determine the parameter values for a time varying compensator.

One method for designing multirate compensators is multirate LQG [4]. Multirate LQG is the multirate equivalent of single rate LQG and is straightforward to solve because the equations governing the solution are similar to those for the single rate case. Multirate LQG, however, results in a full order compensator which

---

* Graduate Student, Department of Mechanical Engineering FU-10, University of Washington, Seattle WA 98195

** AIAA Member, Assistant Professor, Department of Mechanical Engineering FU-10, University of Washington, Seattle WA 98195

has periodically time varying gains. For many applications full order time varying compensators are not practical.

A Generalized Algorithm for Multirate Synthesis (GAMS) [6] was developed by Yang to overcome many of the shortcomings of multirate LQG. Yang's algorithm can synthesize reduced order multirate compensators with or without time varying gains by using a numerical gradient type search to find optimum compensator parameter values. His algorithm uses a finite time cost function in its problem formulation, unlike multirate and single rate LQG which use an infinite time cost function. By using a finite time cost function Yang's algorithm eliminates the numerical problem that arises when a destabilizing compensator is encountered during the numerical search. Even though Yang's algorithm uses a closed form expression for the gradient, the calculations necessary to perform the gradient-type search are extremely cumbersome.

In this paper we present a new algorithm for synthesizing reduced order multirate compensators with or without time varying gains. The algorithm utilizes the compensator structure of Yang's algorithm, but the problem is formulated using an infinite time, instead of a finite time, cost function. This allows us to derive necessary conditions for which the multirate compensator minimizes the cost function. The equations for the necessary conditions are fairly simple and can be solved directly using a standard nonlinear equation solver, eliminating many of the numerical complexities of Yang's algorithm.

THE GENERAL MULTIRATE COMPENSATOR

Before deriving the equations governing a reduced order multirate compensator, we will first present the structure for a general multirate compensator. We restrict our discussion for now to compensators with time invariant gains and sampling/update rates whose ratios are rational numbers.

A general multirate compensator is shown in Figure 1. Each input (y), output (u), and state ($\tilde{x}$) is sampled/updated at a rate which, in general, represents the desired bandwidth of the input or output with which it is associated. $\tilde{y}$ is the value of y currently available to the digital processor from the zero order hold; while $\bar{u}$ is the current output from the digital processor which is held with a zero order hold to form the output u. When the sampling/update rates have ratios which are rational numbers the sampling/update schedule is periodically time varying. We define the greatest common divisor of all the sampling/update periods as the *shortest time period* (STP) and the least common multiple of all the sampling/update periods as the *basic time period* (BTP) (see Figure 2).

The state equations for the multirate compensator pictured in Figure 1 are:

$$\begin{Bmatrix} \tilde{z} \\ \tilde{y} \\ \tilde{u} \end{Bmatrix}_{k+1} = \begin{bmatrix} [I-s_{z,k}]+s_{z,k}\bar{A} & s_{z,k}\bar{B}[I-s_{y,k}] & 0 \\ 0 & [I-s_{y,k}] & 0 \\ s_{u,k}\bar{C} & s_{u,k}\bar{D}[I-s_{y,k}] & [I-s_{u,k}] \end{bmatrix} \begin{Bmatrix} \tilde{z} \\ \tilde{y} \\ \tilde{u} \end{Bmatrix}_k + \begin{bmatrix} s_{z,k}\bar{B}s_{y,k} \\ s_{y,k} \\ s_{u,k}\bar{D}s_{y,k} \end{bmatrix} y_k \qquad (1)$$

$$u_k = \begin{bmatrix} s_{u,k}\bar{C} & s_{u,k}\bar{D}[I-s_{y,k}] & [I-s_{u,k}] \end{bmatrix} \begin{Bmatrix} \tilde{z} \\ \tilde{y} \\ \tilde{u} \end{Bmatrix}_k + \begin{bmatrix} s_{u,k}\bar{D}s_{y,k} \end{bmatrix} y_k \qquad (2)$$

$\tilde{u}$ is a hold state used to model the sampler and zero order hold between $\bar{u}$ and u. $s_{y,k}$, $s_{z,k}$, and $s_{u,k}$ are switching matrices for y, $\tilde{z}$, and u respectively that model the system's sampling/update activity at the start of the k^{th} STP. $s_{*,k}$ has the form:

$$s_{*,k} = \begin{bmatrix} r_1 & 0 & 0 & \cdots & 0 \\ 0 & r_2 & 0 & \cdots & 0 \\ \vdots & & & & \vdots \\ 0 & \cdots & 0 & r_{m_*-1} & 0 \\ 0 & \cdots & 0 & 0 & r_{m_*} \end{bmatrix}$$

where $$r_j = \begin{cases} 1 \text{ if the } j^{th} \text{ "*" } (\tilde{z},\ y, \text{ or } u) \text{ is sampled/updated} \\ \qquad\qquad \text{at the start of the } k^{th} \text{ STP} \\ 0 \text{ otherwise} \end{cases}$$

m_z = the number of states ($\tilde{z}$)

m_y = the number of inputs (y)

m_u = the number of outputs (u)

A more complete discussion of this compensator structure can be found in [6]-[7].

Equations (1) and (2) can be written more compactly as:

$$z_{k+1} = A_k z_k + B_k y_k \qquad (3)$$

$$u_k = C_k z_k + D_k y_k \qquad (4)$$

$$\text{where } z_k \equiv \begin{Bmatrix} \tilde{z} \\ \tilde{y} \\ \tilde{u} \end{Bmatrix}_k$$

Equations (3) and (4) form a single rate periodically time varying system with a sampling rate of one STP and a period of one BTP. If N=BTP/STP, then $A_k = A_{k+N}$, $B_k = B_{k+N}$, $C_k = C_{k+N}$, and $D_k = D_{k+N}$.

Even though A_k, B_k, C_k and D_k are periodically time varying, the multirate compensator gains, $\bar{A}$, $\bar{B}$, $\bar{C}$, and $\bar{D}$, are time invariant. The periodicity of the multirate compensator is due to multirate sampling/updating, not the compensator gains. In the remainder of this section we will demonstrate how the time invariant compensator gains, $\bar{A}$, $\bar{B}$, $\bar{C}$, and $\bar{D}$, can be separated from the periodic compensator matrices A_k, B_k, C_k and D_k.

Define the composite compensator matrix as

$$P_k = \begin{bmatrix} D_k & C_k \\ B_k & A_k \end{bmatrix} \tag{5}$$

and factor P_k as follows:

$$P_k = S_{1k}\tilde{P}S_{2k} + S_{3k} \tag{6}$$

where

$$\tilde{P} = \begin{bmatrix} \bar{D} & \bar{C} \\ \bar{B} & \bar{A} \end{bmatrix} \tag{7}$$

$$S_{1k} = \begin{bmatrix} s_{u,k} & 0 \\ 0 & s_{z,k} \\ 0 & 0 \\ s_{u,k} & 0 \end{bmatrix} \tag{8}$$

$$S_{2k} = \begin{bmatrix} s_{y,k} & 0 & I\text{-}s_{y,k} & 0 \\ 0 & I & 0 & 0 \end{bmatrix} \tag{9}$$

$$S_{3k} = \begin{bmatrix} 0 & 0 & 0 & I\text{-}s_{u,k} \\ 0 & I\text{-}s_{z,k} & 0 & 0 \\ s_{y,k} & 0 & I\text{-}s_{y,k} & 0 \\ 0 & 0 & 0 & I\text{-}s_{u,k} \end{bmatrix} \tag{10}$$

Equation (6) is a key result. It allows us to factor the time invariant compensator gains, the unknown parameters we will solve for in the next section, out of the time varying compensator.

It is important to note the difference between P_k and $\tilde{P}$ in (6). P_k (with a subscript) is a periodically time varying matrix defined by (5). It includes all the information about the compensator gains *and* the

sampling/update schedule. $\tilde{P}$ is a constant matrix which contains *only* the gains for the compensator. P_k can be written in terms of $\tilde{P}$ and S_{1k}, S_{2k}, and S_{3k} using equation (6). S_{1k}, S_{2k}, and S_{3k} are periodically time varying matrices which contain a description of the sampling/update scheme.

DERIVATION OF THE NECESSARY CONDITIONS

In this section we will use the results of the previous section to derive the necessary conditions for the reduced order multirate compensator. The multirate problem to be solved is as follows:

Given: the discretized plant model

$$\hat{x}_{k+1} = \hat{F}\hat{x}_k + \hat{G}\hat{u}_k + \hat{W}w_k \tag{11}$$

$$\hat{y}_k = \hat{H}\hat{x}_k + v_k \tag{12}$$

where $\hat{F}$, $\hat{G}$, $\hat{W}$ and $\hat{H}$ are obtained by discretizing the analog plant matrices at one STP; w_k and v_k are discrete-time Gaussian white noise inputs; $\hat{u}$ is the control input from the compensator; and $\hat{y}$ is the sampled sensor output.

Find: the multirate control law with a prescribed dynamic order and sampling schedule, of the form of (1)-(2), which minimizes a quadratic cost function of the form:

$$J = \lim_{i \to \infty} \sum_{k=1}^{N} E\left\{ \begin{Bmatrix} x_k \\ u_k \end{Bmatrix}^T \begin{bmatrix} Q_1 & M \\ M^T & Q_2 \end{bmatrix} \begin{Bmatrix} x_k \\ u_k \end{Bmatrix} \right\} \tag{13}$$

where E is the expected value operator, and the summation from 1 to N accounts for the fact that the closed loop system is periodically time varying. A prescribed sampling schedule implies that the values of $s_{z,k}$, $s_{y,k}$, and $s_{u,k}$ are known.

Using (3)-(4) it is easy to see that this problem is essentially a time varying feedback problem - a time invariant plant with a periodically time varying compensator. One thing that makes this problem difficult is that the compensator has an explicit form, that of (1)-(2), in which only certain parameters, $\bar{A}$, $\bar{B}$, $\bar{C}$, and $\bar{D}$, can be adjusted to minimize J.

To solve the multirate control problem we cast it into output feedback form and follow a derivation similar to Mukhopadhyay's for the single rate case [8]-[9]. Using (3)-(4), and (11)-(12) we write the output feedback equations:

$$\begin{Bmatrix} \hat{x}_{k+1} \\ z_{k+1} \end{Bmatrix} = \begin{bmatrix} \hat{F} & 0 \\ 0 & 0 \end{bmatrix} \begin{Bmatrix} \hat{x}_k \\ z_k \end{Bmatrix} + \begin{bmatrix} \hat{G} & 0 \\ 0 & I \end{bmatrix} \begin{Bmatrix} \hat{u}_k \\ z_{k+1} \end{Bmatrix} + \begin{bmatrix} \hat{W} & 0 \\ 0 & 0 \end{bmatrix} \begin{Bmatrix} w_k \\ v_k \end{Bmatrix} \tag{14}$$

$$\begin{Bmatrix} \hat{y}_k \\ z_k \end{Bmatrix} = \begin{bmatrix} \hat{H} & 0 \\ 0 & I \end{bmatrix} \begin{Bmatrix} \hat{x}_k \\ z_k \end{Bmatrix} + \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix} \begin{Bmatrix} w_k \\ v_k \end{Bmatrix} \tag{15}$$

$$\begin{Bmatrix} \hat{u}_k \\ z_{k+1} \end{Bmatrix} = \begin{bmatrix} D_k & C_k \\ B_k & A_k \end{bmatrix} \begin{Bmatrix} \hat{y}_k \\ z_k \end{Bmatrix} \tag{16}$$

Equations (14)-(16) can be written more compactly as

$$x_{k+1} = Fx_k + Gu_k + W\eta_k \tag{17}$$

$$y_k = Hx_k + V\eta_k \tag{18}$$

$$u_k = P_k y_k \tag{19}$$

It is important to keep in mind that P_k in equation (19) corresponds to the P_k in equation (5), a periodically time varying matrix which contains all the information about the multirate compensator gains and sampling/update rates.

The closed loop system is

$$x_{k+1} = F_{ck}x_k + G_{ck}\eta_k \tag{20}$$

where

$$F_{ck} = F + GP_kH \tag{21}$$

$$G_{ck} = W + GP_kH \tag{22}$$

The state covariance propagation for this system obeys:

$$X_{k+1} = F_{ck}X_kF_{ck}^T + G_{ck}RG_{ck}^T \tag{23}$$

where

$$X_k = E\{x_k x_k^T\}$$
$$R = E\{\eta_k \eta_k^T\}$$

Equations (20)-(22) represent a periodically time varying system with a period of one BTP. We can generate a single rate system by repeated application of equation (20) over one BTP [10]. The single rate system can be written as

$$x_{k+N} = F_{bk} x_k + G_{bk} \eta_{bk} \tag{24}$$

where

$$F_{bk} = F_{c(k+N-1)} F_{c(k+N-2)} F_{c(k+N-3)} \cdots F_{ck} \tag{25}$$

$$G_{bk} = [F_{c(k+N-1)} F_{c(k+N-2)} \cdots F_{c(k+1)} G_{ck} \mid F_{c(k+N-1)} F_{c(k+N-2)} \cdots F_{c(k+2)} G_{c(k+1)} \mid \cdots \mid G_{c(k+N-1)}] \tag{26}$$

$$\eta_{bk} = \begin{bmatrix} \eta_k \\ \eta_{k+1} \\ \vdots \\ \eta_{k+N-1} \end{bmatrix}$$

This single rate system has exactly the same values for x as the periodically time varying closed loop system at each BTP. However, the values of x at the intermediate STP's are lost because x is incremented by N in (24) but only by 1 in (20). There are N such single rate systems associated with (20). They can be written as:

$$x_{k+N+i} = F_{b(k+i)} x_{k+i} + G_{b(k+i)} \eta_{k+i} \quad \text{for } i=1,2...,N \tag{27}$$

If F_{bk} is stable, then the periodically time varying system (20) is stable [11]. We can calculate the steady state covariance for x using the following Lyapunov equations:

$$X_k = F_{bk} X_k F_{bk}^T + G_{bk} R_b G_{bk}^T \quad \text{for } k=1,2,...,N \tag{28}$$

$$R_b = \begin{bmatrix} R & 0 & \cdots & 0 \\ 0 & R & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & 0 & R \end{bmatrix}$$

Note that X_k is periodic, that is it varies within one BTP, but from BTP to BTP $X_k = X_{k+N}$. Once we have calculated X_k at any k using (28), we can use (23) to propagate it over the BTP. This eliminates the need to solve equation (28) N times.

Now, using (23), (13), and the properties of the Trace (Tr) operator we can write the cost function for the stabilized system as (see [8]-[9])

$$J = \sum_{k=1}^{N} Tr\left\{\left[Q_1 + MP_kH + (MP_kH)^T + (P_kH)^TQ_2P_kH\right]X_k + (P_kV)^TQ_2P_kVR\right\} \tag{29}$$

Adjoin the covariance constraints (23) to the cost J using Lagrange multipliers, Λ_k, to obtain:

$$\begin{aligned}\bar{J} = \sum_{k=1}^{N} Tr\{&\left[Q_1 + MP_kH + (MP_kH)^T + (P_kH)^TQ_2P_kH\right]X_k + (P_kV)^TQ_2P_kVR \\ &+ \Lambda_{k+1}^T\left[F_{ck}X_kF_{ck}^T + G_{ck}RG_{ck}^T - X_{k+1}\right]\}\end{aligned} \tag{30}$$

with $X_1 = X_{N+1}$.

Necessary conditions for minimum J are

$$\frac{\partial \bar{J}}{\partial X_k} = 0, \quad \frac{\partial \bar{J}}{\partial \Lambda_{k+1}} = 0, \text{ and } \frac{\partial \bar{J}}{\partial \tilde{P}} = 0 \tag{31}$$

In addition, $\dfrac{\partial^2 \bar{J}}{\partial \tilde{P}^2}$ must be positive definite for a minimum J.

Substituting (30) into (31) and replacing P_k with $P_k = S_{1k}\tilde{P}S_{2k} + S_{3k}$ from (6) we obtain:

$$\frac{\partial \bar{J}}{\partial X_k} = 0 = Q_1 + MP_kH + (MP_kH)^T + (P_kH)^TQ_2P_kH + F_{ck}^T\Lambda_{k+1}F_{ck} - \Lambda_k \tag{32}$$

$$\text{for } k=1,2,\ldots,N \text{ with } \Lambda_k = \Lambda_{k+N}$$

$$\frac{\partial \bar{J}}{\partial \Lambda_{k+1}} = 0 = F_{ck}X_kF_{ck}^T + G_{ck}RG_{ck}^T - X_{k+1} \tag{33}$$

$$\text{for } k = 1,2,\ldots,N \text{ with } X_k = X_{k+N}$$

$$\begin{aligned}\frac{\partial \bar{J}}{\partial \tilde{P}} = 0 = 2\sum_{k=1}^{N} S_{1k}^T\{&\left[Q_2 + G^T\Lambda_{k+1}G\right]P_k\left[HX_kH^T + VRV^T\right] \\ &+ \left[M^T + G^T\Lambda_{k+1}F\right]X_kH^T\}S_{2k}^T\end{aligned} \tag{34}$$

Equations (32)-(34) are a set of coupled matrix equations. They make up necessary conditions for $\tilde{P}$, which is comprised of the multirate compensator gain matrices $\bar{A}$, $\bar{B}$, $\bar{C}$, and $\bar{D}$, in equation (1)-(2), to minimize the cost function J. Values of $\bar{A}$, $\bar{B}$, $\bar{C}$, and $\bar{D}$, found by solving (32)-(34) can be substituted into (1)-(2), along with the definition of the sampling schedule, $s_{z,k}$, $s_{u,k}$, and $s_{y,k}$, to form the complete time varying multirate compensator.

To ensure that the compensator gains satisfying (32)-(34) minimize J, we should also check that the Hessian of $\bar{J}$ with respect to $\tilde{P}$ is positive definite. Our present algorithm does not calculate the Hessian explicitly, but uses an approximate value calculated by the numerical search algorithm discussed in the next section.

Equations (32)-(34) were derived assuming time invariant compensator gains. We can easily derive the corresponding equations for periodically time varying gains. Let

$$\bar{A} = \bar{A}_k,\ \bar{B} = \bar{B}_k,\ \bar{C} = \bar{C}_k, \text{ and } \bar{D} = \bar{D}_k \tag{35}$$

with the restriction that $\bar{A}_{k+N} = \bar{A}_k$, $\bar{B}_{k+N} = \bar{B}_k$, $\bar{C}_{k+N} = \bar{C}_k$, and $\bar{D}_{k+N} = \bar{D}_k$

Define the composite periodically time varying compensator matrix:

$$\tilde{P}_k = \begin{bmatrix} \bar{D}_k & \bar{C}_k \\ \bar{B}_k & \bar{A}_k \end{bmatrix} \tag{36}$$

Then replace $\tilde{P}$ with $\tilde{P}_k$ in (30) and differentiate with respect to $\tilde{P}_k$ to obtain

$$\begin{aligned} \frac{\partial \bar{J}}{\partial \tilde{P}_k} = 0 = S_{1k}^T \{ &[Q_2 + G^T \Lambda_{k+1} G] P_k [H X_k H^T + V R V^T] \\ &+ [M^T + G^T \Lambda_{k+1} F] X_k H^T \} S_{2k}^T \quad \text{for } k = 1,2,\ldots N \end{aligned} \tag{37}$$

Thus for every new set of compensator gains we obtain one new equation of the form of (37).

Equations (32)-(34) are very similar to the single rate equations. In fact, if we set S_{1k}, S_{2k} and S_{3k} so they correspond to a single rate system, and N=1, we obtain the exact results derived by Mukhopadhyay for the single rate case [8].

IMPLEMENTATION

In order to find a reduced order multirate compensator that minimizes the cost function J, we need to solve (32)-(34) for the compensator gains $\tilde{P}$. A flow chart of the algorithm used to determine the compensator gains is shown in Figure 3. Using the prescribed sampling schedule the algorithm first discretizes the analog plant model, analog cost function, and analog process noise model. See [2] for a discussion of the relevant discretization procedures. Equations (32)-(34) are then solved for the compensator gains using a gradient type search in MATLAB [12]. We chose a gradient type search to solve (32)-(34) because it allows us to easily add constraints on the parameters values - simple equality constraints were used to find the optimized compensators in the next section. The equations necessary to solve for the Lagrange multipliers are located in the Appendix, (A.3)–(A.4). To ensure that the solution represents a minimum J, the algorithm checks that the Hessian of J with respect to the free parameters in $\tilde{P}$ is positive definite at the solution point.

Because (32)-(34) are not valid when the closed loop system is unstable, the algorithm 1) must be provided with an initial stabilizing compensator, and 2) must result in a stabilizing compensator at every iteration. From our experience, finding an initial stabilizing compensator is generally not a problem. Many systems suitable for multirate control can be stabilized using successive loop closure with minimal cross coupling between the control loops. A stabilizing multirate compensator can then be obtained by discretizing the individual continuous control loops at the desired sampling rates. When there are no constraints on its structure, a stabilizing compensator can also be obtained using the boot strapping method of Boussard [13]-[14]. For difficult multirate control problems, where a stabilizing compensator cannot be found using either of the preceeding two methods, one can always use Yang's algorithm to find a stabilizing compensator and then to switch to our algorithm to complete the optimization. In our experience Yang's algorithm usually converges to a stabilizing solution quickly - it is the optimization of the compensator parameters that is time consuming.

To avoid the problem of destabilizing compensators during the iteration process we included a check in the algorithm which systematically reduces the step size to ensure that the compensator is stabilizing. Because the gradient of the cost function with respect to the compensator parameters becomes very large near the stability boundary, the algorithm is always forced away from a destabilizing solution as long as it never steps over the stability boundary into an unstable region.

Even though our algorithm was programmed as an interpreted Matlab M-File we found that it still performed better than Yang's algorithm which runs as compiled FORTRAN. The primary difference between the two algorithms is in the complexity of the expression for the gradient of J with respect to the compensator

parameters. Calculation of the gradient expression for Yang's problem involves diagonalization of the closed loop system and evaluation of several matrix equations with nested summations. Compare equations (32)-(34) with equations (112-115) in [3] to see the difference in the complexity of the two gradient expressions.

TWO LINK ROBOT ARM EXAMPLE

We used a math model of a planar two link robot arm (TLA) to demonstrate the capabilities of our algorithm. This is the same model used by Yang [6], and so we were able to verify our results by direct comparison. A diagram of the TLA is shown in Figure 4.

The goal of our design was to control the tip position (δ) of the arm via a multirate compensator. We used the following analog cost function and process noise covariance matrices from [6].

$$J = \lim_{t \to \infty} E\left\{ x^T \begin{bmatrix} 0.21 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 18.5 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} x + u^T \begin{bmatrix} 0.01 & 0 \\ 0 & 0.69444 \end{bmatrix} u \right\} \tag{38}$$

$$\text{where } x = \begin{Bmatrix} \theta \\ \dot{\theta} \\ \delta \\ \dot{\delta} \end{Bmatrix} \text{ and } u = \begin{Bmatrix} T_1 \\ T_2 \end{Bmatrix}$$

$$E\{w w^T\} = \begin{bmatrix} 0.69444 & 0 \\ 0 & 0.01 \end{bmatrix} \tag{39}$$

We assumed perfect measurement and that plant disturbances enter the system coincident with the control torques. The sampling/update rates are given in Table 1.

Table 1: Sampling/Update Rates for TLA

	Sample/Update Rate
θ	0.225 s
δ	0.028125 s
T_1	0.225 s
T_2	0.028125 s

Five different compensators were designed: an analog LQR, a multirate lead/lead, an optimized multirate lead/lead, an optimized multirate general 2^{nd} order, and an optimized single rate general 2^{nd} order. We used a smooth step input to δ_{ref} and θ_{ref} defined as follows:

$$\delta_{ref}(t) = \begin{cases} 0.005\left[1 - \cos\left(\frac{\pi t}{T_c}\right)\right] \text{ m} & t \le T_c \\ 0.01 \text{ m} & t \ge T_c \end{cases}$$

$$\theta_{ref}(t) = \frac{\delta_{ref}(t)}{L_1 + L_2}, \qquad T_c = 0.125 \text{ sec} \tag{40}$$

and the servo configuration shown in Figure 5 to measure the performance of the different compensators. The response of the TLA for the five compensators is shown in Figures 6a-6c.

The analog LQR compensator used full state feed back. We provided this compensator as an example of the response possible using the cost function weighting matrices of (38).

The multirate lead/lead was found using successive loop closures. We designed the control loops in the discrete domain so that the eigenvalues of the closed loop system matched those we obtained using LQR transformed to discrete time. This compensator consists of two simple lead loops: one from δ to T_2 operating at the fast sampling/update rate, and one from θ to T_1 operating at the slow sampling/update rate.

The final three compensators were synthesized using our new algorithm and the cost weighting matrices used to design the analog LQR compensator. The optimized multirate lead/lead was found by optimizing the pole/zero locations and gains of the lead/lead compensator found by successive loop closures.

The optimized multirate general 2^{nd} order compensator uses the same sampling/update scheme as the lead/lead compensators but has the compensator structure of (41), where a_{ij}, b_{ij}, c_{ij}, and d_{ij} are the parameters which were optimized. This compensator has the maximum number of independent free parameters possible for a second order system [7].

$$\bar{A} = \begin{bmatrix} a_{11} & 0 \\ 0 & a_{22} \end{bmatrix} \quad \bar{B} = \begin{bmatrix} 1 & b_{12} \\ b_{21} & 1 \end{bmatrix} \quad \bar{C} = \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix} \quad \bar{D} = \begin{bmatrix} d_{11} & d_{12} \\ d_{21} & d_{22} \end{bmatrix} \tag{41}$$

The optimized single rate general 2^{nd} order compensator is a single rate equivalent of the multirate general 2^{nd} order compensator. It has the same structure as the multirate general 2^{nd} order compensator, (41), but uses a single sampling rate. This sampling rate was chosen such that the number of computations required to implement either the multirate or single rate compensators during real-time operation are the same.

Our results are the same as those obtained using Yang's algorithm. Thev demonstrate how multirate compensators can provide better performance than single rate compensators by trading lower bandwidth control of the slow modes for higher bandwidth control of the fast modes. In this example we were able to reduce the tip response over shoot 40% and the peak control torque 25% by using a multirate controller over a single rate controller.

CONCLUSIONS

In this paper we have presented a new algorithm for synthesizing reduced order multirate compensators. It can be use to design compensators of arbitrary structure and dynamic order, with independent sampling/update rates for the compensator inputs, outputs and states. This algorithm provides the versitility of Yang's alorithm with out the numerical complexities associated with the finite time cost function.

Finally, we do not want to discount Yang's algorithm altogether because, while our algorithm requires an initial stabilizing compensator, Yang's does not. For those problems where finding an initial stabilizing compensator is difficult, we can always use Yang's algorithm to find a stabilizing compensator and then quickly optimize the compensator parameter values with our algorithm.

ACKNOWLEDGMENT

This research was supported by NASA Langley Research Grant NAG-1-1055.

REFERENCES

[1] Berg, M.C., Amit N. and Powell J.D., "Multirate Digital Control system Design," *IEEE Trans. Auto. Conr.*, Vol AC-33, Dec 1988, pp.1139-1150.

[2] Berg, M.C., "Design of Multirate Digital Control Systems," Ph.D. Thesis, Stanford Univ., Stanford, CA, 1986.

[3] Berg, M.C., and Yang, G.S., "A New Algorithm for Multirate Digital Control Law Synthesis," *Proc. IEEE Conf. Decision Contr.*, Dec 1988, Austin, TX, pp. 1685-1690.

[4] Amit, N., "Optimal Control of Multirate Digital Control Systems," Ph.D. Thesis, Dep. Aero. Astro. Stanford Univ., Stanford, CA, Rep. 523, 1980.

[5] Amit, N., and Powell J.D., "Optimal Control of Multirate Systems," *Proc. AIAA Guid. Contr. Conf.*, Albuquerque, NM, 1981.

[6] Yang, G.S., "A Generalized Synthesis Method for Multirate Feedback Control Systems," Ph.D. Thesis, Univ. of Washington, Seattle, WA, 1988.

[7] Berg, M.C., Mason, G.S., and Yang, G.S., "A New Multirate Sampled-Data Control Law Structure and Synthesis Algorithm," *1991 American Control Conference*, Boston, MA, June 1991.

[8] Mukhopadhyay, V., "Digital Robust Control Law Synthesis Using Constrained Optimization," *AIAA Jour. Guid., Contr. and Dynamics*, Vol. 12, March-April 1989, pp. 175-181.

[9] Mukhopadhyay, V., Newsom, J.R., and Abel, I., "A Method for Obtaining Reduced-Order Control Laws for High-Order Systems Using Optimization," NASA Tech. Paper 1876, August 1981.

[10] Meyer, R.A., and Burrus, C.S., "A Unified Analysis of Multirate and Periodically Time-Varying Digital Filters," *IEEE Trans. Circuits in Systems*, Vol. CAS-22, No. 3, March 1975, pp. 162-168.

[11] Kono, M., "Eigenvalue assignment in Linear Periodic Discrete-Time Systems," *Int. J. Control*, Vol. 32, No. 1, 1980, pp. 149-158.

[12] *Pro-Matlab Users Guide*, The Mathworks Inc., 1989.

[13] Broussard, J.R., and Halyo, N., "Optimal Multirate Output Feedback," *Proceed. 23rd Conf. Decis. Contr.*, Las Vegas NV, Dec 1984, pp. 926-929.

[14] Halyo, N., and Broussard, J.R., "A Convergent Algorithm for the Stocastic Infinite-Time Discrete Optimal Output Feedback Problem," *Proceed. Joint Autom. Contr. Conf.*, Charlottesville VA, 1981, Vol. 1, Section WA-1E.

APPENDIX

Given a P_k which stabilizes the multirate system we can calculate the steady state values of Λ_k where Λ_k is defined by equation (32) rewritten here as (A.1).

$$0 = Q_1 + MP_kH + (MP_kH)^T + (P_kH)^TQ_2P_kH + F_{ck}^T\Lambda_{k+1}F_{ck} - \Lambda_k \qquad \text{(A.1)}$$

$$\text{for } k=1,2,\ldots,N \text{ with } \Lambda_k = \Lambda_{k+N}$$

First simplify (A.1) by defining

$$Q_3 \equiv \begin{bmatrix} Q_1 & M \\ M^T & Q_2 \end{bmatrix} \text{ and } J_k \equiv \begin{bmatrix} I \\ P_kH \end{bmatrix} \qquad \text{(A.2)}$$

I is an identity matrix

Then (A.1) can be written as

$$\Lambda_k = J_k^TQ_3J_k + F_{ck}^T\Lambda_{k+1}F_{ck} \quad \text{for } k=1,2,\ldots,N \text{ with } \Lambda_k = \Lambda_{k+N} \qquad \text{(A.3)}$$

Equation (A.3) represents a periodically time varying Lyapunov equation. We can create an equivalent single rate system by repeated application of (A.3).

$$\Lambda_k = J_{dk}^TQ_dJ_{dk} + F_{dk}^T\Lambda_kF_{dk} \quad \text{for } k=1,2,\ldots,N \text{ with } \Lambda_k = \Lambda_{k+N} \qquad \text{(A.4)}$$

$$F_{dk} = F_{c(k+N-1)}F_{c(k+N-2)}F_{c(k+N-3)}\cdots F_{ck} \qquad \text{(A.5)}$$

$$J_{dk} = \begin{bmatrix} J_{(k+N-1)}F_{c(k+N-2}F_{c(k+N-3)}\cdots F_{ck} \\ J_{(k+N-2)}F_{c(k+N-3)}\cdots F_{ck} \\ \vdots \\ J_k \end{bmatrix} \qquad \text{(A.6)}$$

$$Q_d = \begin{bmatrix} Q_3 & 0 & \cdots & 0 \\ 0 & Q_3 & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & 0 & Q_3 \end{bmatrix}$$

Equation (A.4) is a time invariant Lyapunov equation which can be solved for Λ_k. Once any Λ_k has been found, the propagation equation (A.3) can be used to find the remaining Λ_k.

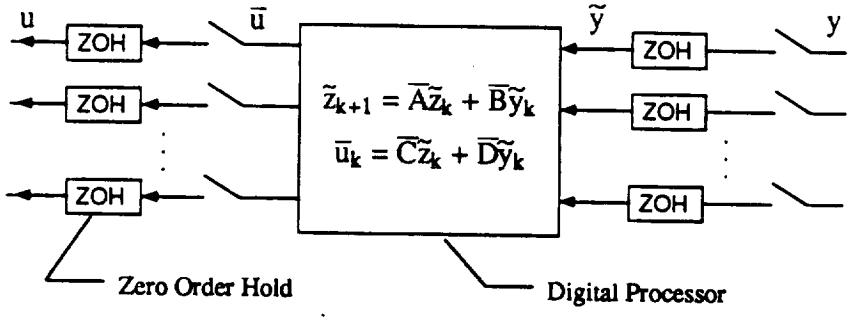


Figure 1. A General Multirate Compensator

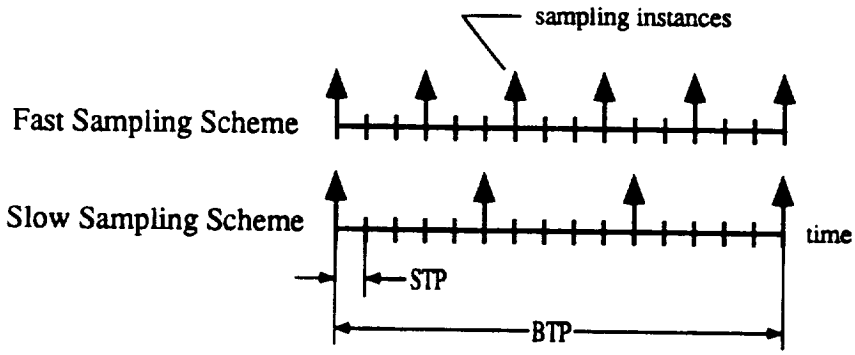


Figure 2. Example of a Multirate Sampling Scheme

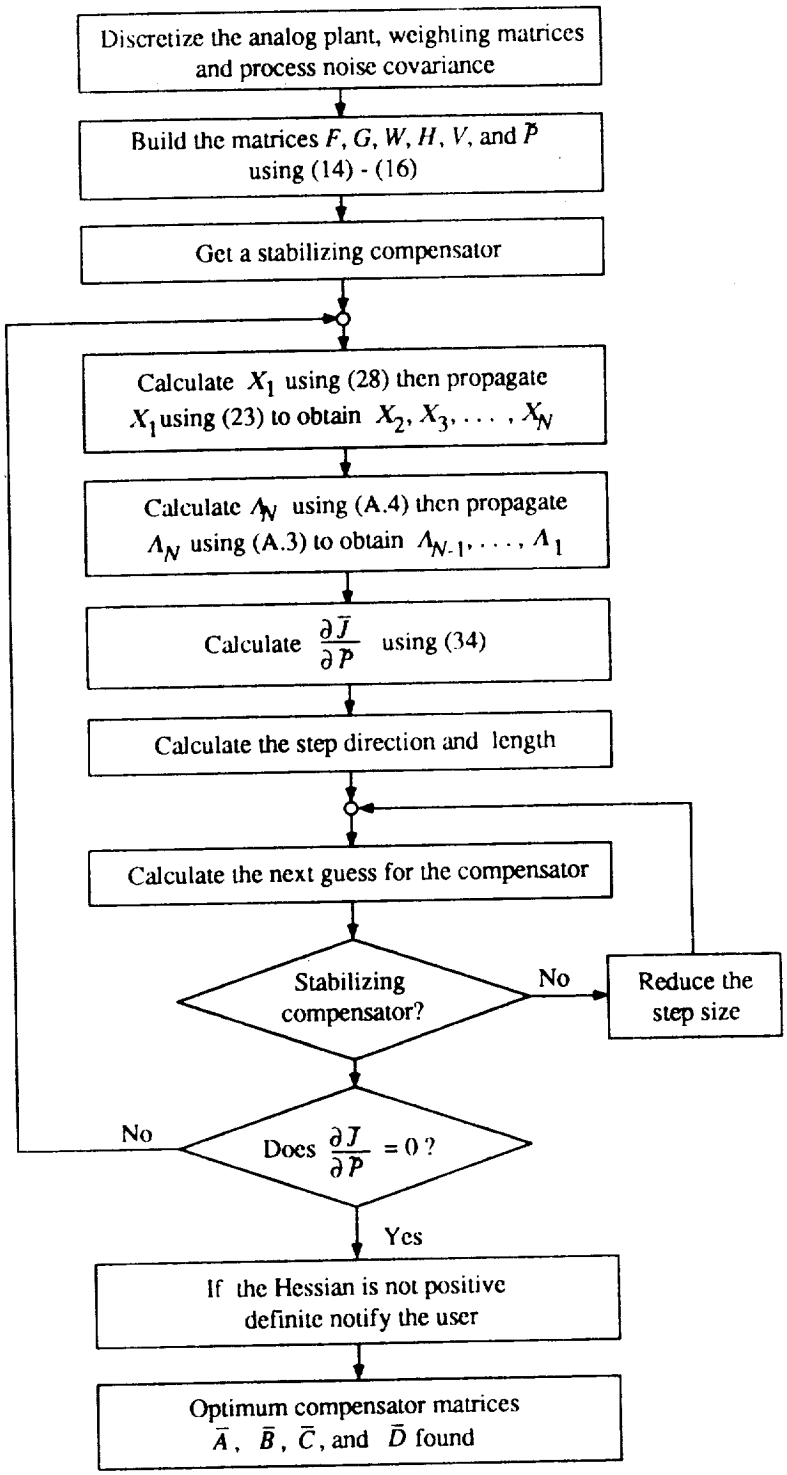
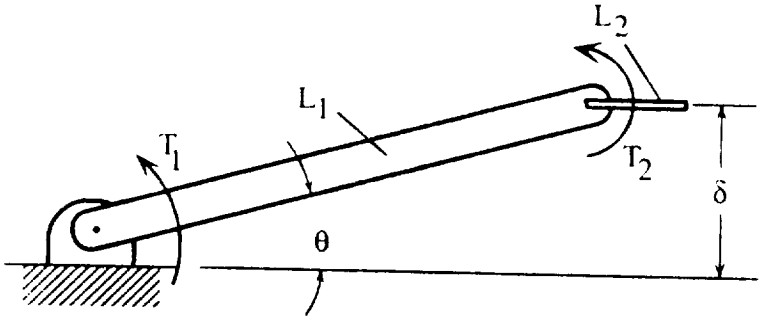


Figure 3. Flow Chart of Optimization Algorithm



Parameters:

	Mass	Length
L_1	1.235 kg	0.965 m
L_2	0.163 kg	0.167 m

Inputs: Torque T_1 and T_2

Outputs: θ and δ

Figure 4. Planar Two Link Robot Arm

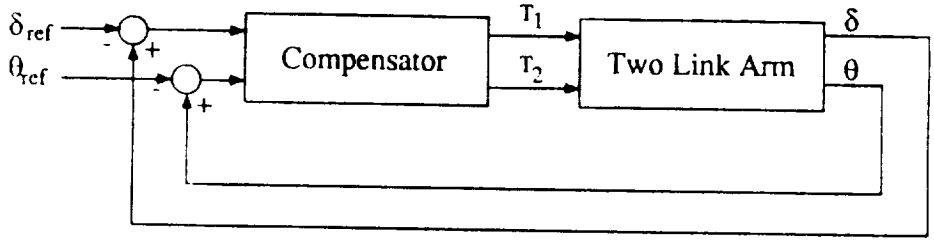


Figure 5. TLA Plant/Compensator Configuration

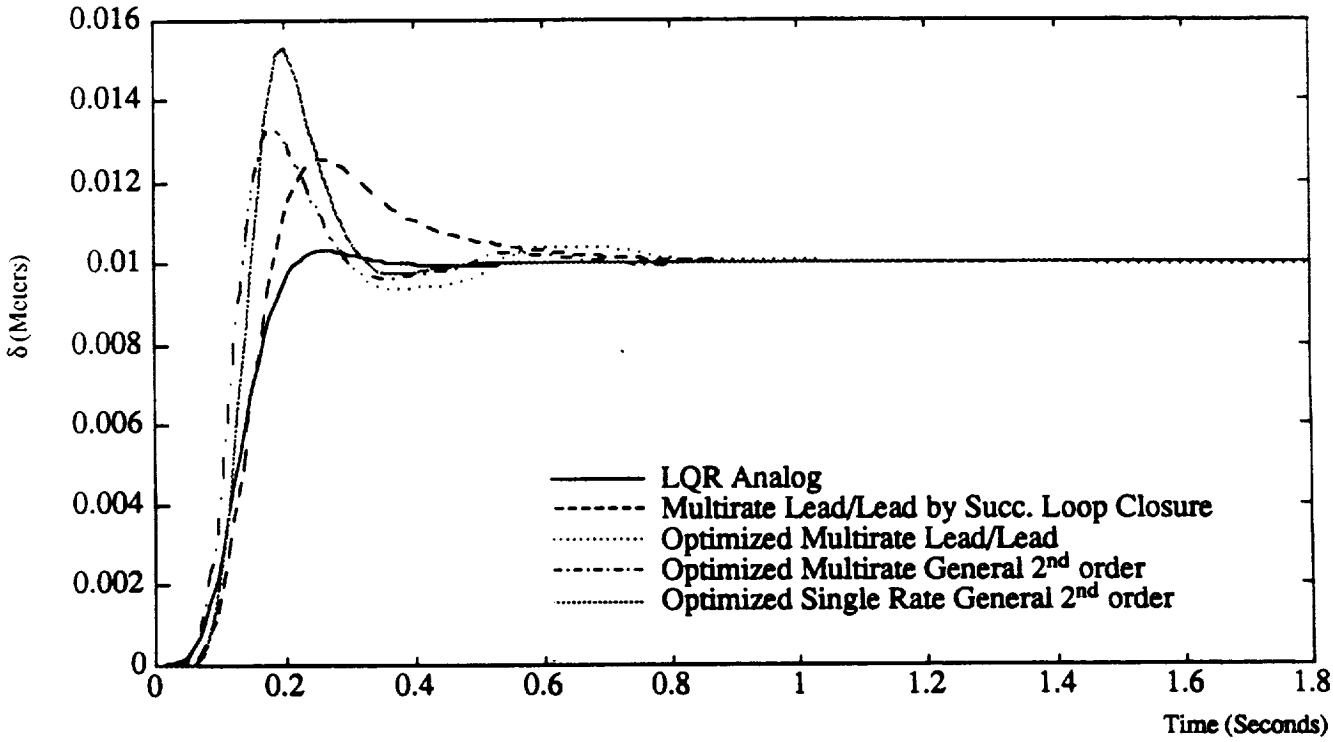


Figure 6a. Tip (δ) Response to a Smooth Step Command to Tip Position

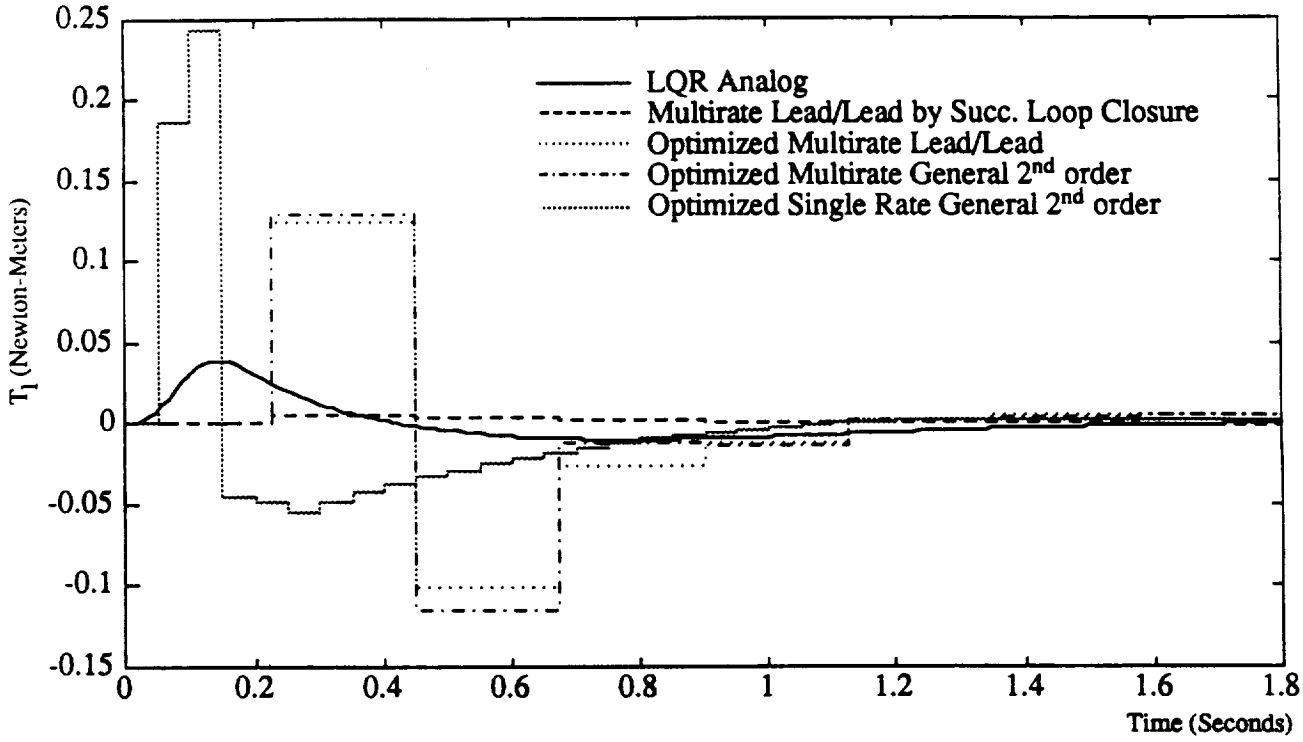


Figure 6b. Control Torque T_1 Response to a Smooth Step Command to Tip Position

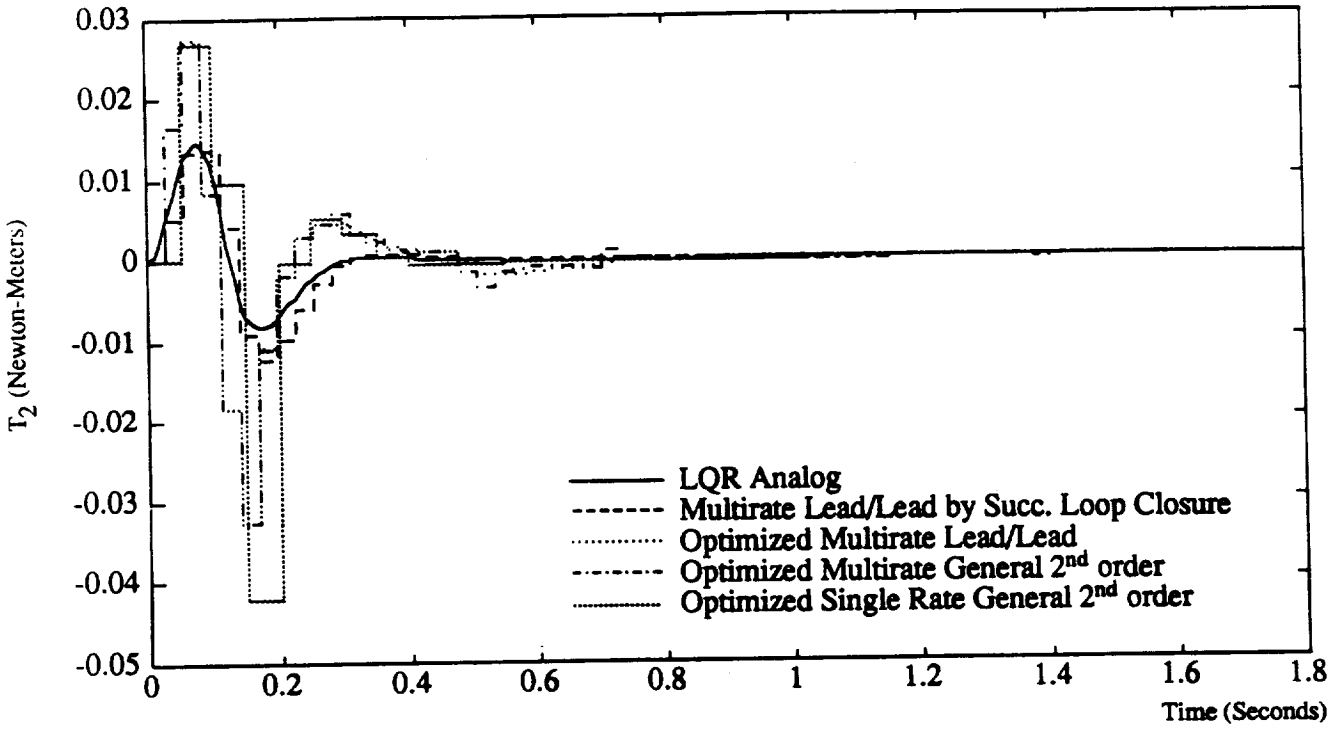


Figure 6c. Control Torque T_2 Response to a Smooth Step Command to Tip Position

APPENDIX C

PREPRINT OF REFERENCE 6

Draft Version

Robustness Analysis of Multirate and Periodically Time Varying Systems

Gregory S. Mason
Graduate Student
Mechanical Engineering Department, FU-10
University of Washington, Seattle, Washington, 98195
Telephone: 206-685-2429

Martin C. Berg
Assistant Professor
Mechanical Engineering Department, FU-10
University of Washington, Seattle, Washington, 98195
Telephone: 206-543-5288

I. INTRODUCTION

There are many established methods for synthesizing multirate compensators [Berg, Yang, Mason, Glasson, Amit] but surprisingly few methods for analyzing the robustness of these systems. Current robustness analysis methods rely principally on the transfer function of the system. A multirate transfer function, in the traditional sense, does not exist, because most multirate systems are periodically time varying. Without some modification, established analysis methods cannot be applied directly to multirate systems.

[Thompson] and [Apostolakis] have both proposed ways to extend existing robustness analysis techniques to multirate systems. Thompson used "Kranc" operators to transform a special class of multirate systems, derived from sampled continuous systems, into MIMO single rate systems. Apostolakis transformed the general multirate system into a discrete time single rate MIMO system and then used impulse modulation to produce a continuous MIMO system [Boykin's]. In both cases, the inputs and outputs of the new MIMO system were comprised of delayed samples of the inputs and outputs of the multirate system. Thompson and Apostolakis then used multivariable nyquist criterion and/or unstructured singular value analysis to calculate the gain and phase margins for the multirate system.

In this paper we will present an alternative approach for extending nyquist criterion and singular value analysis to multirate and periodically time varying systems. Like Apostolakis, we transform the original system into an equivalent time invariant single rate system. However, we perform the robustness analysis in the "z" domain. By working in the "z" domain we can establish relationships between a multirate/periodically time varying system and its time invariant single rate equivalent. These relationships clarify the limitations of nyquist and singular value analysis using single rate equivalent systems.

The paper is divided into five section. Section I provides some back ground information about multirate systems and discusses transfer functions for multirate and periodically time varying systems. Section II discusses the application of the nyquist stability criterion to these systems; Section III discusses the application of structured and unstructured singular values analysis to these systems. Section IV contains a example of robustness analysis using structured singular values for a multirate system. Concluding remarks follow in Section V.

I. MULTIRATE AND PERIODICALLY TIME VARYING SYSTEMS

Before discussing robustness analysis we will first establish the relationship between multirate and periodically time varying systems. Then we will define an equivalent single rate system which will allow us to combine periodically time varying, multirate and time invariant systems using traditional block diagram techniques.

A general multirate compensator is shown in Figure 1. Each input (y), output (u), and state ($\tilde{z}$) is sampled/updated at a rate which, in general, represents the desired bandwidth of the input or output with which it is associated. $\tilde{y}$ is the value of y currently available to the digital processor from the zero order hold; while $\bar{u}$ is the current output from the digital processor which is held with a zero order hold to form the output u. A discussion of this compensator structure can be found in [Berg & Mason, and Yang].

Associated with this multirate compensator is a multirate sampling schedule which specifies the sampling/update rate for each input, output and state. We define the greatest common divisor of all the sampling/update periods as the *shortest time period* (*STP*) and the least common multiple of all the sampling/update periods as the *basic time period* (*BTP*). The integer N is defined as:

$$N = \frac{BTP}{STP} \tag{1}$$

When the sampling/update rates have ratios which are rational numbers, the sampling/update schedule is periodically time varying and the multirate compensator can be modeled as a linear periodically time varying system of the form [Mason & Berg]:

$$x(k+1) = A(k)x(k) + B(k)u(k) \tag{2}$$
$$yk) = C(k)x(k) + Dk)u(k) \tag{3}$$

where $A(k) = A(k+N)$, $B(k) = B(k+N)$, $C(k) = C(k+N)$, and $D(k) = D(k+N)$

The sampling period for (2)-(3) is one *STP* and the period of repetition is one *BTP*.

Any practical multirate system can be modeled as linear periodically time varying system of the form of (2)-(3). Therefore, we will focus the remainder of the discussion on linear periodically time varying systems, of which multirate and single rate are a special case.

Given a periodically time varying system of the form of (2)-(3), we can create an equivalent time invariant system by repeated application of (2)-(3) over the *BTP* [Meyer & Burrus]. The equivalent time invariant system is

$$x(N(k+1)) = Ax(Nk) + B\hat{u}(Nk) \tag{4}$$
$$\hat{y}(Nk) = Cx(Nk) + D\hat{u}(Nk) \tag{5}$$

Where

$$A = A(N\text{-}1)A(N\text{-}2)\cdots A(0) \tag{6}$$

$$B = [A(N\text{-}1)A(N\text{-}2)\cdots A(1)B(0) \mid A(N\text{-}1)A(N\text{-}2)\cdots A(2)B(1) \mid \cdots \mid B(N\text{-}1)] \tag{7}$$

$$C = \begin{bmatrix} C(0) \\ C(1)A(0) \\ \vdots \\ C(N\text{-}1)A(N\text{-}2)\cdots A(0) \end{bmatrix} \tag{8}$$

$$D = \begin{bmatrix} D(0) & & & \\ C(1)B(0) & & & \\ C(2)A(1)B(0) & & & \\ \vdots & & & \\ C(N\text{-}1)A(N\text{-}2)\cdots A(2)B(0) & & & \\ 0 & \cdots & 0 & 0 \\ D(1) & & & \vdots \\ C(2)B(1)) & \ddots & 0 & 0 \\ \vdots & & D(N\text{-}2) & 0 \\ C(N\text{-}1)A(N\text{-}2)\cdots A(2)B(1) & \cdots & C(N\text{-}1)B(N\text{-}2) & D(N\text{-}1) \end{bmatrix} \tag{9}$$

$$\text{where } \hat{y}(Nk) = \begin{bmatrix} y(Nk) \\ y(Nk+1) \\ \vdots \\ y(Nk+N\text{-}1) \end{bmatrix} \text{ and } \hat{u}(Nk) = \begin{bmatrix} u(Nk) \\ u(Nk+1) \\ \vdots \\ u(Nk+N\text{-}1) \end{bmatrix} \tag{10}$$

Equations (6)-(10) transforms the linear periodically time varying system, (2)-(3), with p inputs, q outputs and a sampling period of one *STP* to a linear time invariant system, (4)-(5), with Np inputs, Nq outputs and a sampling period of one *BTP*. We will refer to (4)-(5) as the *equivalent single rate system* (*ESRS*) of (2)-(3). It is important to keep in mind that the inputs and outputs of an *ESRS* are comprised of samples of the inputs and outputs of a time varying system. A consequence of this is the relationship:

$$\|\hat{y}\|_2 = \|y\|_2 \tag{11}$$

We can calculate the transfer function of (4)-(5) using the following definitions for the Z Transform. Let

$$Z(x(k)) = \sum_{i=0}^{\infty} x(i)z^{-i} \tag{12}$$

$$Z_N(x(k)) = \sum_{i=0}^{\infty} x(iN)z^{-iN} \tag{13}$$

then

$$y(z^N,l) = Z_N(y(k+l)) \tag{14}$$

$$\hat{y}(z^N) = \begin{bmatrix} Z_N\{y(k)\} \\ Z_N\{y(k+1)\} \\ \vdots \\ Z_N\{y(k+N-1)\} \end{bmatrix} \tag{15}$$

with a similar definitions for $\hat{u}(z^N)$ and $u(z^N,l)$.

The transfer function for the *ESRS*, (4)-(5), is

$$\hat{y}(z^N) = G_N(z^N)\hat{u}(z^N) \tag{16}$$

$$\text{where } G_N(z^N) = C(Iz^N - A)^{-1}B + D \tag{17}$$

The transfer function is written as $G_N(z^N)$ to emphasize that the sampling period of the *ESRS* is one *BTP*, or N times the sampling period of the time varying system (2)-(3).

So far, the *ESRS* has only been applied to periodically time varying systems. We could, however, calculate the *ESRS* of a *time invariant* system - in this case N can be any integer.

The transfer function for the *ESRS* of a time invariant system can be calculate using (17). Alternatively, $G_N(z^N)$ can be calculated directly in terms of the transfer function of the time invariant system, $G(z)$. Given

$$y(z) = G(z)u(z) \tag{18}$$

and following [Meyers & Burrus] we can write

$$y(z) = \sum_{l=0}^{N-1} z^{-l} y(z^N,l) \tag{19}$$

$$y(z^N,l) = \frac{1}{N} z^l \sum_{i=0}^{N-1} \phi^{-li} y(z\phi^i) \tag{20}$$

where $\phi = e^{\frac{2\pi}{N}}$

Combine (18)-(20) to obtain

$$y(z^N,l) = \frac{1}{N} z^l \sum_{i=0}^{N-1} \sum_{m=0}^{N-1} z^{-m} \phi^{-i(l-m)} G(z\phi^i) u(z^N,m) \qquad (21)$$

From (15) and (21) the l^{th} row and m^{th} column of $G_N(z^N)$ is given by

$$G_N(z^N)_{l,m} = \frac{1}{N} z^{l-m} \sum_{i=0}^{N-1} \phi^{-i(l-m)} G(z\phi^i) \qquad (22)$$

For a time invariant system, $G_N(z^N)$ is made up of time and frequency shifted versions of $G(z)$. A special case of (22) occurs when $G(z\phi^i) = G(z)$ for $i = 0,1,...,N-1$.

$$\text{If } G(z\phi^i) = G(z) \text{ for } i=0,1,...,N-1 \text{ then } G_N(z^N)_{l,m} = \begin{cases} G(z) & \text{if } l = m \\ 0 & \text{otherwise} \end{cases} \qquad (23)$$

The simplest $G(z)$ satisfying (23) is $G(z)$ = constant. Equation (23) is an important relationship which will be used in Sections II and III.

Equations (6)-(10), (16)-(17) or (21) can be used to compute state space and transfer function descriptions for the *ESRS* of a periodically time varying or time invariant system. The advantage of the *ESRS* is that is it allows us to manipulate time invariant and periodically time varying systems (e.g. multirate) as if they were both time invariant. The state space or transfer functions descriptions can be used to calculate input-output relations for systems in series or in a feedback loop just as in classical control [Khargonekar]. In addition, [Kono] has shown that if the *ESRS* is stable then the time varying system from which it was derived will be stable. So, we need only worry about the stability of the *ESRS*.

II. NYQUIST STABILITY CRITERION

We can determine the stability of the periodically time varying system in Figure 2 by applying standard multiloop nyquist [McFarlane ...] criterion to the *ESRS*, since the periodically time varying system will be stable if its *ESRS* is stable. The *ESRS* return difference is given by

$$I - G_N(z^N)\Delta_N(z^N) \qquad (24)$$

and the nyquist contour is

$$z^N = e^{j\omega} \quad 0 \le w \le 2\pi$$

When the periodically time varying system is SISO, we can determine traditional gain and phase margins from the nyquist plot. Recall that when $\Delta(z\phi^i) = \Delta(z)$, $\Delta_N(z^N)$ is a diagonal transfer function matrix with $\Delta(z)$ on the diagonal. Thus, the *ESRS* for Figure 2 with gain and phase uncertainty can be written as

$$G_N(z^N)_{actual} = G_N(z^N)_{nominal} I_{NxN} k e^{j\theta} \qquad (25)$$

Phase and gain margins from the nyquist plot can be interpreted in the traditional sense even though the *ESRS* is MIMO because the inputs and outputs are correlated in time and a constant gain applies equally over all time. [Thompson] arrived at this same results using Kranc operators.

When the time varying system is MIMO, the standard MIMO nyquist restrictions apply. For MIMO time varying systems, it is best to use a norm based approach such as singular value analysis.

III. Structured and Unstructured Singular Values Analysis

In the previous section we saw that the multiloop nyquist stability criterion can be applied to an *ESRS* to determine the stability of a periodically time varying system. In this section we will see that, with some limitations, both structured and unstructured singular value analysis can be applied to the *ESRS* to determine the robustness properties of a periodically time varying system.

Given stable transfer functions $G(z)$ and $\Delta(z)$ it has been shown that the closed loop system will remain stable through out continuous changes in $\Delta(z)$ if

$$det\,(I - G(z)\Delta(z)) \neq 0 \tag{26}$$

$$\text{or } \underline{\sigma}\,(I - G(z)\Delta(z)) > 0 \tag{27}$$

is satisfied around the nyquist contour, subject to certain restriction on $Gz)$ and $\Delta(z)$ [Maciejousky,...]. By direct application of (26)-(27), a periodically time varying system will be stable if

$$det\,(I - G_N(z^N)\Delta_N(z^N)) \neq 0 \tag{28}$$

$$\text{or } \underline{\sigma}(I - G_N(z^N)\Delta_N(z^N)) > 0 \tag{29}$$

is satisfied around the nyquist contour, because a periodically time varying system will be stable if its *ESRS* is stable. From (28)-(29) it follows that most singular value robustness tests can be applied directly to a *ESRS* to determine the robustness properties of a periodically time varying system. The results, though, must be interpreted in light of the fact that some of the inputs and outputs of the *ESRS* are time correlated. In the following paragraphs we will discuss the important differences between a single rate system and an *ESRS* and how these affect singular value analysis.

1) *There are qN, not q, singular values associated with each point on the nyquist contour for the ESRS of a time varying system with only q inputs and outputs.* The additional singular values come from the time correlated inputs and outputs of the *ESRS*. Remember that the sampling period of the *ESRS* is one BTP, N times slower than the time varying system from which it was derived; but the *ESRS* has N times as many inputs and outputs as the original time varying system. The key point is that *all* of these singular values are important in determining the robustness of a periodically time varying system.

If an *ESRS* is generated from a time invariant system, the singular values of the *ESRS* and the singular values of the original single rate system are related by the following expression.

$$\sigma G_N(e^{jN\omega}) = \left[\sigma G(\phi^0 e^{j\omega}), \sigma G(\phi^1 e^{j\omega}), \cdots \sigma G(\phi^{N-1} e^{j\omega})\right] \tag{30}$$

In (30), singular values associated with frequencies above $1/BTP$ in $G(z)$ are reflected back to lower frequencies in $G_N(z^N)$. It follows directly from (30) that

$$\|G_N(z^N)\|_\infty = \|G(z)\|_\infty \tag{31}$$

where $\|G(z)\|_\infty = \sup_\omega \bar{\sigma}[G(e^{j\omega})]$

2) *The ESRS imposes a structure on any uncertainty* Δ_N. Any Δ_N in (28)-(29) must obey (17) or (22); this automatically imposes a structure on Δ_N.. The problem is, we are often interested in a time invariant plant uncertainty, Δ, that destabilizes the system and not in Δ_N. $\bar{\sigma}(\Delta_N)$ found using unstructured singular value analysis is often overly conservative because it accounts for not only the fictitious perturbations normally associated unstructured singular values but also for time varying and non-causal perturbations. It is important to remember that $\bar{\sigma}(\Delta_N)$ found using *unstructured* singular values can be extremely conservative and may not reflex $\bar{\sigma}(\Delta)$.

Structured singular values provides a mechanism for finding Δ. For an *ESRS* with a time invariant uncertainty, $\Delta(z)$, the definition of the structured singular value, μ, can be written as

$$\mu(G_N(z^N)) = \begin{cases} 0 \text{ if } det(I - G_N(z^N)\Delta_N(z^N)) \neq 0 \text{ for any } \Delta \in \Delta_{BD} \\ \left\{ \min_{\Delta \in \Delta_{BD}} (\bar{\sigma}(\Delta(z)): det\,(I - G_N(z)\Delta_N(z)) = 0 \right\}^{-1} \text{ otherwise} \end{cases} \tag{32}$$

Δ_{BD} is the form of the permissible block diagonal perturbations Δ; and the structure of Δ_N must satisfy equation (22).

Unfortunately for a general $\Delta(z)$, the structure of $\Delta_N(z)$ is often very complex and finding a good estimate of size of Δ is difficult. However, when Δ is a constant, as is the case for many problems,

$$\Delta_N = diag(\Delta, \Delta, \ldots \Delta) \text{ with } N \text{ blocks} \tag{33}$$

and structured singular value analysis can be used to determine Δ.

When Δ is a time varying, but not a function of "z", Δ_N becomes

$$\Delta_N = diag(\Delta(1), \Delta(2), \ldots \Delta(N)) \tag{34}$$

and has no repeated blocks. Equation (34) must be interpreted with care - (34) implies that the value of $\Delta(k)$ is constant over the sampling interval, *STP*, and changes instantaneously to $\Delta(k+1)$ at the next sampling instant. This may not be a good model of time varying uncertainty.

3) *When* Δ *is a constant then each* Δ *block of* Δ_N *can be scaled independently.* Using the block diagonal scaling property of μ [Maciejowski], and (33) or (34) it is straightforward to see that

$$\text{If } \Delta \text{ is a } p \text{ by } p \text{ matrix then } \mu(DG_N(z^N)D^{-1}) = \mu(G_N(z^N)) \tag{35}$$
$$\text{where } D = (d_1 I_p,\ d_2 I_p, \ldots d_N I_p) \text{ and } I \text{ is a } p \text{ by } p \text{ identity matrix}$$

In addition if Δ is block diagonal then each of sub-block of Δ can be scaled in a similar manner.

An interesting result of (35) is that the upper bound for $\mu(G_N(z^N))$ given by

$$\mu(G_N(z^N)) \le \inf_D(DG_N(z^N)D^{-1}) \qquad (36)$$

is the same whether Δ is periodically time varying or time invariant, subject to the interpretation of a time varying Δ mentioned in item 2. The upper bound for the time invariant case found using (36) is of course more conservative.

4) *Singular value plots of ESRS transfer function matrices should not be interpreted in the frequency domain.* The *ESRS* has time correlated inputs and outputs. The response from one input to one output represents only part of the total signal between the input and output of the periodically time varying system.

A meaningful quantity for an *ESRS* is its infinity norm. From (11) and [Francis] we can write that

$$\sup \frac{\|y\|_2}{\|u\|_2} = \sup \frac{\|y\|_2}{\|u\|_2} = \|G_N(z^N)\|_\infty \text{ for } \|u\|_2 < \infty \qquad (37)$$

Thus, $\|G_N(z^N)\|_\infty$ can be interpreted as the maximum gain of the system for all u with a bounded two norm, just as in the single rate case. For the single rate case the maximum gain occurs when u is sinusoidal - this is not necessarily true for the periodically time varying system.

Singular value analysis of an *ESRS*, both structured and unstructured, can be used to determine the robustness of periodically time varying system. As we have discussed, there are limitations to this analysis because the inputs and outputs of an ESRS are time correlated.

IV. Two Link Arm Example

The results of the previous sections will be illustrated by calculating the gain margins for a planar two link robot arm (TLA) using structured singular values. Two different cases are considered: 1) the TLA with a 2^{nd} order multirate compensator and 2) the TLA with a 2^{nd} order single rate compensator.

The TLA is shown in Figure 3 and is described further in [Berg and Yang]. The two compensators were designed to minimize a cost function quadratic in the states and controls using the optimization method described in [Mason & Berg].

The 2^{nd} order multirate compensator uses the compensator structure of (38), where a_{ij}, b_{ij}, c_{ij}, and d_{ij} are the parameters which were optimized.

$$\bar{A} = \begin{bmatrix} a_{11} & 0 \\ 0 & a_{22} \end{bmatrix} \quad \bar{B} = \begin{bmatrix} 1 & b_{12} \\ b_{21} & 1 \end{bmatrix} \quad \bar{C} = \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix} \quad \bar{D} = \begin{bmatrix} d_{11} & d_{12} \\ d_{21} & d_{22} \end{bmatrix} \qquad (38)$$

The sampling/update rates for the compensator are listed in Table 1. In addition the compensator state associated with θ and T_1 is updated at the slow rate while the state associated with δ and T_2 is updated at the fast rate. For the multirate compensator $STP = .028125$, $BTP = .225$ and $N = 8$

	Multirate Compensator	Single Rate Compensator
θ	0.225 s	0.05 s
δ	0.028125 s	0.05 s
T_1	0.225 s	0.05 s
T_2	0.028125 s	0.05 s

Table 1. Sampling/Update Periods for the Compensators

The single rate 2^{nd} order compensator is the single rate equivalent of the multirate compensator. It has the same structure as the 2^{nd} order multirate compensator and minimizes the same cost function, but uses a single sampling/update rate. This sampling/update rate was chosen such that the number of computations required to implement either the multirate or single rate compensators during real-time operation is the same. The sampling/update rate for the compensator is shown in Table 1.

A block diagram of the TLA, compensator, and output gain uncertainty is shown in Figure 4. The bock diagram in Figure 4 can be cast into the standard structured uncertainty model shown in Figure 5 where the gains k_1 and k_2 are allowed to vary independently. An upper bound on the structured singular values for the multirate and the single rate cases was calculated using the following [Safonov].

$$\mu(Q) \le \inf_{D} \bar{\sigma}(DQD^{-1}) \le \lambda_P(Q) \qquad (39)$$

where $\lambda_P(Q)$ is the Perron-Frobenius eigenvalue of Q

For the multirate case, the *ESRS* for the plant, compensator and uncertainty was calculated for $N = 8$. They were combined as shown in Figure 5 and an upper bound for μ, as z traversed the nyquist contour, was calculated using (39). A lower bound on the maximum singular value of the gain matrix is given in Table 2.

For the single rate case an upper bound on μ was calculated using two different methods. First, μ was calculated directly for the single rate system, $N = 1$. An exact value for μ can be calculated because there are only two blocks in the uncertainty matrix [Doyle]. Next, an *ESRS* was constructed for the single rate case using $N = 8$. An upper bound for μ was calculated using (39). For the *ESRS* system, the uncertainty matrix has 8, 2 by 2 blocks. These results are summarized in Table 2.

Design	Gain Margin $\bar{\sigma}\begin{bmatrix} k_1 & 0 \\ 0 & k_2 \end{bmatrix} = \frac{1}{\mu}$
Multirate 2^{nd} Order	0.535
Single Rate 2^{nd} Order	0.513
Single Rate 2^{nd} Order using *ESRS*	0.389

Table 2. Gain Margins for TLA with Multirate and Single Rate Compensator

The two μ estimates for the single rate case illustrate the disadvantage of using (39) to calculate the upper bound of μ for an *ESRS* systems. As in the multirate case, the *ESRS* single rate case accounts for periodically time varying uncertainties, resulting in a conservative estimate of μ. See item 3, Section III.

For the assumed uncertainty model the multirate compensator was slightly more robust, even given the conservativeness of the estimate for μ. The multirate compensator is able to compensate for larger gain uncertainty because it has higher bandwidth control of the second link than does the single rate compensator.

V. Summary and Conclusions

In this paper we have shown how nyquist criterion and singular value analysis can be applied to multirate and periodically time varying systems using their *ESRS*. For SISO systems, traditional gain and phase margins can be found by direct application of the nyquist criterion to the *ESRS*. For MIMO systems, structured singular values can be used to determine the maximum size of an uncertainty. The results of singular value analysis, though, must be interpreted in light of the fact that some of the inputs and outputs of the *ESRS* are time correlated. We pointed out several important resulting limitations of singular value analysis using an *ESRS*. Finally we demonstrated robustness analysis for a two link arm with a multirate compensator using structured singular value.

APPENDIX

Lemma: $\sigma G_N(e^{jN\omega}) = \left[\sigma G(\phi^0 e^{j\omega}), \sigma G(\phi^1 e^{j\omega}), \cdots \sigma G(\phi^{N-1} e^{j\omega})\right]$

Proof: From the definition of a transfer function we can write

$$\tilde{y}(z) = \tilde{G}(z)\tilde{u}(z) \tag{A.1}$$

$$\text{where } \tilde{y}(z) = \begin{bmatrix} y(\phi^0 z) \\ y(\phi^1 z) \\ \vdots \\ y(\phi^{N-1} z) \end{bmatrix}, \tilde{u}(z) = \begin{bmatrix} u(\phi^0 z) \\ u(\phi^1 z) \\ \vdots \\ u(\phi^{N-1} z) \end{bmatrix} \tag{A.2}$$

$$\tilde{G}(z) = \begin{bmatrix} G(\phi^0 z) & 0 & \cdots & 0 \\ 0 & G(\phi^1 z) & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & 0 & 0 & G(\phi^{N-1} z) \end{bmatrix}$$

From (12), $y(z) = \sum_{l=0}^{N-1} z^{-l} y(z^N, l)$ so that we can write $\tilde{y}(z) = T y(z^N)$

$$\text{where } T = \begin{bmatrix} I & z^{-1}I & \cdots & z^{-(N-1)}I \\ I & (\phi^1 z)^{-1}I & & (\phi^1 z)^{-(N-1)}I \\ \vdots & \vdots & \ddots & \vdots \\ I & (\phi^{N-1} z)^{-1}I & \cdots & (\phi^{N-1} z)^{-(N-1)}I \end{bmatrix} \tag{A.3}$$

T has the property that $TT^* = NI$ if z is evaluated on the unit circle and I is an identity matrix of appropriate dimensions.

Then

$$y(z^N) = T^{-1}\tilde{G}(z)Tu(z^N) \text{ so that } G_N(z^N) = T^{-1}\tilde{G}(z)T \tag{A.4}$$

Now using the fact that $\sigma^2(A)$ equals the eigenvalues of A^*A, and that the eigenvalues of a block diagonal matrix are the eigenvalues of the individual block it follows that

$$\sigma G_N(e^{jN\omega}) = \left[\sigma G(\phi^0 e^{j\omega}), \sigma G(\phi^1 e^{j\omega}), \cdots \sigma G(\phi^{N-1} e^{j\omega})\right] \tag{A.5}$$

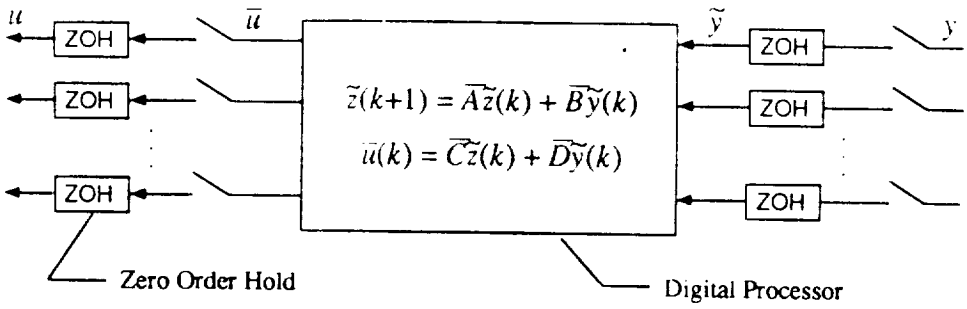


Figure 1. A General Multirate Compensator

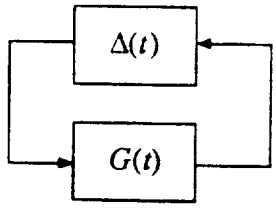
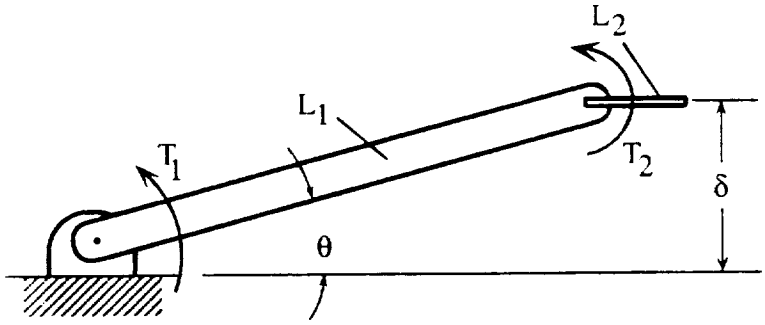


Figure 2. Periodically Time Varying System



Parameters:	Mass	Length
L_1	1.235 kg	0.965 m
L_2	0.163 kg	0.167 m

Inputs: Torque T_1 and T_2

Outputs: θ and δ

Figure 3: Diagram of the Two Link Arm

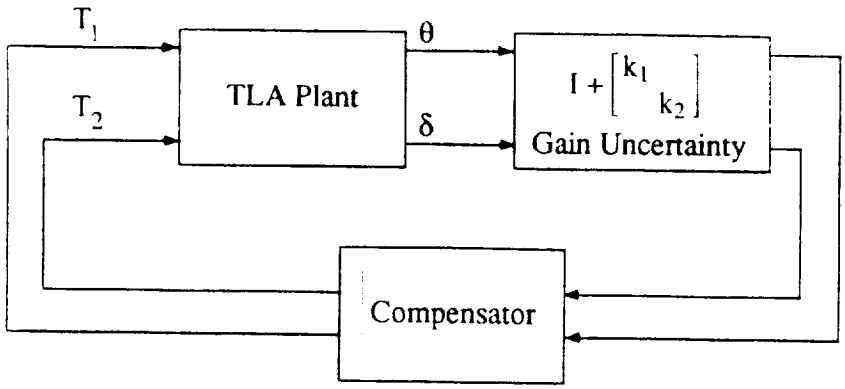
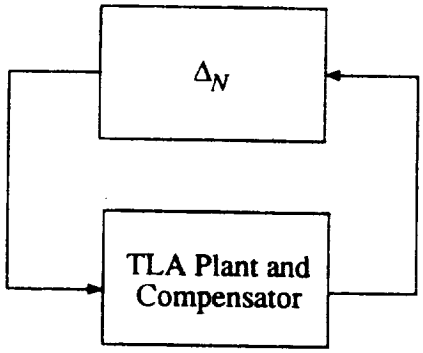


Figure 4. TLA Feedback Loop with Output Gain Uncertainty



$$\Delta_N = \text{diag}\left\{\begin{bmatrix} k_1 & \\ & k_2 \end{bmatrix}_1, \begin{bmatrix} k_1 & \\ & k_2 \end{bmatrix}_2, \cdots \begin{bmatrix} k_1 & \\ & k_2 \end{bmatrix}_N\right\}$$

Figure 5. Structured Uncertainty Model for TLA

NASA
National Aeronautics and Space Administration

Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA CR-187561		

4. Title and Subtitle	5. Report Date
Progress in Multirate Digital Control System Design	June 1991
	6. Performing Organization Code

7. Author(s)	8. Performing Organization Report No.
Martin C. Berg and Gregory S. Mason	
	10. Work Unit No. 505-64-30-01

9. Performing Organization Name and Address	11. Contract or Grant No.
University of Washington Mechanical Engineering Department Seattle, WA 98195	NAG1-1055
	13. Type of Report and Period Covered
12. Sponsoring Agency Name and Address	Contractor Report
NASA Langley Research Center Hampton, VA 23665-5225	14. Sponsoring Agency Code

15. Supplementary Notes

Langley Technical Monitor: Aaron J. Ostroff
Final Report

16. Abstract

A new methodology for multirate sampled-data control system design based on (1) a new generalized control law structure, (2) two new parameter-optimization-based control law synthesis methods, and (3) a new singular-value-based robustness analysis method is described. The control law structure can represent multirate sampled-data control laws of arbitrary structure and dynamic order, with arbitrarily prescribed sampling rates for all sensors and update rates for all processor states and actuators. The two control law synthesis methods employ numerical optimization to determine values for the control law parameters to minimize a quadratic cost function, possibly subject to constraints on those parameters. The robustness analysis method is based on the multivariable Nyquist criterion applied to the loop transfer function for the sampling period equal to the period of repetition of the system's complete sampling/update schedule. The complete methodology is demonstrated by application to the design of a combination yaw damper and modal suppression system for a commercial aircraft.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
Control, Multirate, Sampled-Data, Digital, Synthesis	Unclassified - Unlimited Subject Category 63

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of pages	22. Price
Unclassified	Unclassified	101	A06