NASA Technical Paper 1231

Applied Routh Approximation

Walter C. Merrill

APRIL 1978

NASA

NASA Technical Paper 1231

Applied Routh Approximation

Walter C. Merrill
Lewis Research Center
Cleveland, Ohio

National Aeronautics
and Space Administration

**Scientific and Technical
Information Office**

1978

APPLIED ROUTH APPROXIMATION

by Walter C. Merrill

Lewis Research Center

SUMMARY

Models of physical processes are often too complex to be handled directly. Approximations of these models that are of reduced complexity but that maintain the essential model characteristics are often used for analysis rather than the original model. One way to generate these approximations is the Routh approximation technique. In this report, this technique is programmed and applied in the frequency domain to a 16th-order, state-variable model of the F100 engine and to a 43rd-order, transfer-function model of a launch vehicle boost-pump pressure regulator. The accuracies of reduced-order approximants obtained for both models are compared with the original model. The results indicate that the Routh method is an excellent technique for linear system reduction in the frequency domain. Therefore, the frequency-domain formulation was extended to the time domain in order to handle the state-variable model directly. The time-domain formulation is derived herein and a new characterization that specifies all possible Routh similarity transformations is given. This characterization is computed by using eigenvector-eigenvalue techniques and is therefore quite accurate. The application of the time-domain Routh technique to the state-variable engine model is described, and reduced-order approximation results are given. Additionally, an optimization procedure that can improve the approximation accuracy by taking advantage of the transformation characterization is applied to the engine example. Also, the limitations of applying the Routh technique to the fixed-dimension, multiple-input - multiple-output reduction problems are discussed.

INTRODUCTION

In control system design the system to be controlled is often represented by a complex mathematical model. Practically, this complex model may be difficult to use for design purposes. Additionally, the resultant control design may be too complex to implement. To eliminate these problems, model reduction methods are often used before designing the control system in order to reduce the complexity of the original model while maintaining its important characteristics. Alternatively, model reduction methods could be applied to the reduction of complex control designs. In either case the model reduction process would be the same.

One philosophy adopted in reduction methods in dominant-mode approximation (ref. 1). In this report, a dominant-mode-approximation technique, the Routh approximation technique (refs. 2 and 3), is applied in both the frequency and time domains to simplify a state-variable turbofan engine model, and in the frequency domain to simplify a transfer-function model of a launch vehicle boost-pump pressure regulator. Hutton and Rabins (ref. 4) have shown the frequency-domain formulation of the Routh approximation technique to be an excellent reduction method for many mechanical systems. This formulation is applicable to turbofan engine and boost-pump regulator models as well and is therefore a strong motivation for the work of this report.

The objectives considered included the application of the Routh approximation method in the frequency domain to the F100 turbofan engine. This application demonstrated that adequate, reduced-complexity models can be determined by the Routh technique. Next, the applicability of the frequency-domain approach to high-order systems was studied by using the boost-pump pressure regulator model. Finally, the frequency-domain Routh formulation was extended to the time domain. This extension allows the usual engine time-domain formulation to be handled directly. The time-domain Routh approximation formulation also allows additional flexibility in the choice of the approximate system. By using the additional flexibility of the time-domain formulation, optimized approximations can be readily generated. The time-domain formulation was applied to the engine model, and some results are given. Optimization results and the application of the Routh method to deriving fixed-dimension realizations of the F100 engine are also discussed.

ROUTH APPROXIMATION

Many physical processes can be represented as multiple-input - multiple-output (MIMO) dynamic systems by using linear, constant-coefficient, differential equations to model small variations at various operating-point conditions. A transfer-function representation of these equations can be used as input information in a classical or multivariable-frequency-response control system design. The complexity of these transfer functions can cloud the design process with unnecessary detail or difficult computational problems. Additionally, control systems may be made more complex by unnecessary model complexity. Thus, there is a strong motivation to reduce the complexity of system and control models.

Frequency Domain

The Routh approximation method (refs. 2 and 3) is a dominant-mode reduction technique. In the frequency domain the method incorporates parameters obtained from a Routh stability table analysis in the reduction process. These parameters, called alpha and beta parameters, are obtained from the coefficients of the MIMO transfer function

$$W(s) = \frac{B_1 s^{n-1} + \ldots + B_n}{a_0 s^n + a_1 s^{n-1} + \ldots + a_n} \tag{1}$$

where the a_i are scalars and the B_i are $l \times m$ matrices. (All symbols are defined in appendix A.) Algorithmically, the n alpha parameters are computed from the denominator of equation (1) as (ref. 4)

$$\left.\begin{aligned} a_j^0 &= a_{n-j} \\ a_j^1 &= a_{n-j-1} \end{aligned}\right\} \quad j = 0, 2, 4, \ldots, \leq n \tag{2}$$

and

$$\left.\begin{aligned} \alpha_i &= \frac{a_0^{i-1}}{a_0^i} \qquad & i = 1, 2, \ldots, n-1 \\ a_j^{i+1} &= a_{j+2}^{i-1} - \alpha_i a_{j+2}^i \qquad & j = 0, 2, 4, \ldots, \leq n-i-1 \end{aligned}\right\} \tag{3}$$

where

$$a_j^i = 0 \qquad \text{for} \qquad i + j > n$$

The coefficients a_j^i can be arranged as entries in a Routh table (ref. 2). This tabular construction suggests the name Routh approximation.

The beta parameters are computed from both the numerator and the denominator of the given transfer function. Consider the single-input - single-output (SISO) transfer function from input q to output p implicitly given in equation (1), $W_{pq}(s)$. The beta parameters for this SISO transfer function are

$$\left.\begin{aligned} b_j^1 &= b_{n-j} \\ b_j^2 &= b_{n-j-1} \end{aligned}\right\} \quad j = 0, 2, 4, \ldots, \leq n \tag{4}$$

and

$$\left.\begin{array}{r} \beta_i = \dfrac{b_0^i}{a_0^i} \\[2ex] b_i^{i+2} = b_{j+2}^i - \beta_i \, a_{j+2}^i \end{array}\right\} \quad \begin{array}{l} i = 1, 2, \ldots, n \\ j = 0, 2, 4, \ldots, \leq n - i - 2 \end{array} \tag{5}$$

where

$$b_j^i = 0 \qquad \text{for} \qquad i + j > n$$

and b_r is the $(qp)^{th}$ element of the matrix B_r, where $r = 1$ to n. In this way, beta parameters for the $l \times m$ possible SISO transfer functions of equation (1) can be found. Again, considering the SISO transfer function $W_{pq}(s)$, the computation of the k^{th} approximant from the first k alpha and beta parameters is given recursively as

$$\left.\begin{array}{r} a_0^k = 1 \\ a_0^{i-1} = \alpha_i a_0^i \\ b_0^i = \beta_i a_0^i \\ a_j^{i-1} = \alpha_i a_j^i + a_{j-2}^{i+1} \\ b_j^i = \beta_i a_j^i + b_{j-1}^{i+2} \end{array}\right\} \quad \begin{array}{l} i = k, k - 1, \ldots, 1 \\ j = 2, 4, 6, \ldots, \leq k - i \end{array} \tag{6}$$

and

$$\left.\begin{array}{r} c_{k-j} = a_j^0 \\ c_{k-j-1} = a_j^1 \end{array}\right\} \; j = 0, 2, 4, \ldots, \leq k \tag{7}$$

$$\left.\begin{array}{r} d_{k-j} = b_j^1 \\ d_{k-j-1} = b_j^2 \end{array}\right\} \; j = 0, 2, 4, \ldots, \leq k \tag{8}$$

where

$$a_j^i = b_j^i = 0 \qquad \text{for} \qquad i + j > k$$

Note that, for different output-input pairs, different approximant orders k can be chosen. The k^{th}-order approximant for the $(pq)^{th}$ output-input pair is written as

$$\left(W_{pq}\right)_k(s) = \frac{d_1 s^{k-1} + d_2 s^{k-2} + \ldots + d_k}{c_0 s^k + c_1 s^{k-1} + \ldots + c_k} \tag{9}$$

Properties. - The Routh approximation method exhibits several very useful properties. Each is important in the reduction problem and is described briefly here.

Stability: If the original transfer function is asymptotically stable, the Routh approximant, of any order, will be asymptotically stable.

Pole-zero locations: The poles and zeros of the approximants approach the poles and zeros of the original function as the order of the approximation is increased.

Impulse response energy: If $h(t)$ is the impulse response of $W_{pq}(s)$, and if the impulse response energy is defined as

$$E = \int_0^\infty h^2(t)\, dt \tag{10}$$

then

$$E = \sum_{i=1}^{n} \frac{\beta_i^2}{2\alpha_i} \tag{11}$$

Also, if $h_k(t)$ is the impulse response of the k^{th}-order approximant of $W_{pq}(s)$, then

$$E_{k+1} = E_k + \frac{\beta_{k+1}^2}{2\alpha_{k+1}} \tag{12}$$

Since $W_{pq}(s)$ is assumed to be asymptotically stable, $\alpha_i > 0$ for $i=1, \ldots, n$ and

$$0 < E_1 < E_2 < E_3 < \ldots < E_n = E \tag{13}$$

The ratio E_k/E gives an indication of the approximation accuracy in terms of the percentage of total energy accounted for by the k^{th} approximation.

Derivatives. - The k^{th} Routh approximant satisfies the following derivative condition:

$$\left.\frac{d^i}{ds^i}[W(s)]\right|_{s=0} = \left.\frac{d^i}{ds^i}[W_k(s)]\right|_{s=0} \qquad i = 0, 1, \ldots, k-1 \tag{14}$$

Initial and final values. - The initial and final values of the step response trajectories for the original system and its k^{th}-order Routh approximant are the same. This is true even though the approximant is a strictly proper transfer function for all p and q.

Time Domain

The Routh approximation method was extended to the time domain in order that Routh approximants could be directly determined from a state-space model formulation. The alternative would be to determine the $p \times q$ transfer functions for the MIMO system and then reduce each function individually. For a typical engine model with $p = 16$ and $q = 5$, this alternative would involve reducing 80 transfer functions at each operating point. Computationally, this is undesirable. Hutton (ref. 3) outlines a method for directly determining the alpha and beta parameters from the state-space formulation. However, computationally, this method failed for the 16th-order engine example of this report. Thus, an alternative computational procedure was devised. It incorporates eigenvalue-eigenvector solutions that are well known and for which excellent computational solutions exist (ref. 5).

Given the state-space representation Σ_1,

$$\left.\begin{aligned} \dot{\hat{x}} &= \hat{A}\hat{x} + \hat{B}u \\ y &= \hat{C}\hat{x} \end{aligned}\right\} \tag{15}$$

where $x \in R^n$, $u \in R^m$, and $y \in R^l$, define a reciprocal system, Σ_2, such that

$$\left.\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx \end{aligned}\right\} \tag{16}$$

where

$$\left.\begin{aligned} A &= \hat{A}^{-1} \\ B &= -A\hat{B} \\ C &= \hat{C} \end{aligned}\right\} \tag{17}$$

The system Σ_1 is assumed to be stable and have distinct eigenvalues. For the jet engine system studied herein, this is a realistic assumption. Note that the reciprocal transformation of Σ_2 is Σ_1. This transformation will always exist since A is assumed to be stable. The transformation is required to preserve the dominant or low-frequency information of Σ_1 during the reduction process (ref. 3). Now consider a nonunique similarity transformation of Σ_2, called the Routh transformation, such that

$$AT_R = T_R R \tag{18}$$

where R is the Routh stability matrix. That is,

$$R = \Lambda_R^{-1} \Gamma \tag{19}$$

$$\Gamma = \begin{bmatrix} -1 & -1 & & & & & \\ 1 & 0 & -1 & & & 0 & \\ & 1 & 0 & -1 & & & \\ & & & \cdot & & & \\ & & & & \cdot & & \\ & & & & & \cdot & \\ & & 0 & & & & \cdot \\ & & & & & 0 & -1 \\ & & & & & 1 & 0 \end{bmatrix} \tag{20}$$

and

$$\Lambda_R = \operatorname{diag}[\alpha_i] \qquad i = 1, \ldots, n \tag{21}$$

Also, consider the modal transformation of Σ_2, where

$$AT_m D_m = T_m D_m \Lambda \tag{22}$$

and

$$\Lambda = \operatorname{diag}[\lambda_i] \qquad i = 1, \ldots, n \tag{23}$$

Here the λ_i are eigenvalues of A and the i^{th} column of T_m is the corresponding eigenvector. Note that T_m will be unique with respect to a scaling convention and a specific order of the λ_i's. The scaling convention is represented by D_m, an arbitrary, full rank, diagonal matrix with diagonal elements d_i that may be complex.

Since R is similar to A,

$$RT_zD_z = T_zD_z\Lambda \tag{24}$$

Again T_z is a modal matrix for R and Λ, and it is unique with respect to a scaling convention represented by D_z. From equations (18), (22), and (24) and the definition

$$D = D_mD_z^{-1} \tag{25}$$

all possible Routh transformations can be characterized as

$$T_R = T_mDT_z^{-1} \tag{26}$$

Recall that a fixed ordering of eigenvalues is given. Permutations of this ordering can be represented by a permutation matrix E_p, where

$$T_{R,p} = E_pT_R \tag{27}$$

and E_p is the identity matrix with appropriate columns interchanged. Although this permutation does not affect the input-output transfer relationship of the original system, it will affect the input-output characteristics of the approximate system. Undoubtedly, one of the $n!$ possible state permutations will yield a better approximation in some sense than another, but this problem is not considered herein.

From equation (18) a Routh canonical system Σ_3 can be written as

$$\left.\begin{aligned} \dot{x}_R &= Rx_R + Gu \\ y &= Hx_R \end{aligned}\right\} \tag{28}$$

where

$$x = T_Rx_R \tag{29}$$

The Routh approximation procedure starts by assuming that

$$\left(x_R\right)_2 = 0 \tag{30}$$

where

$$x_R = \begin{bmatrix} \left(x_R\right)_1 \\ \hdashline \left(x_R\right)_2 \end{bmatrix} \tag{31}$$

and $\left(x_R\right)_1 \in R^k$, and $\left(x_R\right)_2 \in R^{n-k}$. The system Σ_2 that incorporates the assumption of equation (30) becomes Σ_2', or

$$\left.\begin{aligned} \dot{x}_1 &= T_{R_{11}} R_{11} T_{R_{11}}^{-1} x_1 + T_{R_{11}} G_1 u \\ y &= H_1 T_{R_{11}}^{-1} x_1 \end{aligned}\right\} \tag{32}$$

where

$$x = \begin{bmatrix} x_1 \\ \hline x_2 \end{bmatrix}$$

and $x_1 \in R^k$ and the subscripts indicate conformable partitioning of the appropriate matrices. The desired approximation then is the reciprocal transformation of Σ_2', called Σ_1'.

Examination of equation (32) shows that the reduced-order system matrix, $T_{R_{11}} R_{11} T_{R_{11}}^{-1}$, is simply a similarity transformation of the truncated Routh stability matrix. Truncating an n^{th}-order Routh matrix R yields another Routh matrix of k^{th} order R_{11}. Since the original system is stable, the alpha parameters of R, α_i (i = 1 or n), are positive. Therefore, R_{11} is also stable. The reduction process implied in equation (32) exhibits the same pole property as the frequency-domain approach.

Hutton (ref. 3) has shown that if the system given by equation (16) is a single-input - multiple-output system and T_R is selected such that

$$T_R^{-1} B = \begin{bmatrix} -1/\alpha_1 \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ 0 \end{bmatrix} \tag{33}$$

the elements of the rows h_i of H become the beta parameters for the output considered

$$h_i = (\beta_{1i}, \beta_{2i}, \ldots, \beta_{ni}) \tag{34}$$

and the properties of the approximation described previously hold. It can be shown that if D is selected as $D = \mathrm{diag}[\theta_i]$, where $\theta = T_m^{-1}B$, equation (33) is satisfied.

COMPUTATIONAL ASPECTS OF ROUTH APPROXIMATION

Computational aspects of both the frequency- and time-domain formulations are discussed in this section.

Frequency Domain

In the frequency domain the Routh approximation problem is solved by calculating the alpha and beta parameters for a given transfer function. Straightforward programming of equations (3), (5), and (6) as detailed in subroutines DROUTH and DTHCNV in appendix B gives these parameters (DROUTH) and the Routh approximants (DTHCNV). Also, the impulse response energy property of equation (11) was programmed in subroutine DIMPLS. Implementing this Routh tabular procedure is straightforward, and computationally the procedure is an efficient and accurate method of reducing SISO transfer functions in the frequency domain.

Time Domain

Computationally, given the state-space formulation of equation (15), time-domain Routh approximation requires (1) the alpha coefficients and (2) the Routh transformation matrix. The alpha coefficients can be found readily from equation (3) if the system-characteristic-equation coefficients can be found. A program that incorporates the method of Danielevski (described in ref. 6) to find the characteristic equation of the system matrix $\hat{A}$ was used in this report. The alpha coefficients were found by applying subroutine DROUTH.

A method for computing the Routh transformation matrix of equation (18) has been proposed by Datta (ref. 5). This method was programmed and applied to the 16th-order jet engine model. Numerical results were unacceptable because of large computational errors, and so another technique was sought. Since accurate and efficient eigenvalue-eigenvector techniques are well known (for the case of distinct eigenvalues), the computation of the Routh transformation matrix was reformulated as the solution of two eigenvalue-eigenvector proglems. These two problems are represented by equations (22) and (24) and are solved by using the computational methods of reference 7. Since

A and R are similar, they have the same eigenvalues. The Routh matrix R is also known from equation (19) since the alpha parameters have been found.

The transformation matrices of equations (22) and (24) may have elements in the complex field. Thus, from equations (2) to (6), T_R can in general be complex. However, a real T_R is desired to yield a physically realizable approximation and to facilitate computer computations. The matrix T_R can be constrained to be real by the proper selection of the elements d_i of D. To determine these elements, consider the eigenvector solution of equation (24) by means of real-number computer operations. Given A and its eigenvalues, the modal matrix of A can be written as

$$T_{mm} = T_m V_m \tag{35}$$

where V_m is the block diagonal and T_{mm}, the modified modal matrix of A, is a matrix of real numbers. If the eigenvalues of A were all real, V_m would equal I. However, if λ_i and λ_{i+1} were a complex conjugate pair, the i^{th} block of V_m, $\left(V_m\right)_i$, would be defined as

$$\left(V_m\right)_i = \frac{1}{2}\begin{bmatrix} 1-j & 1+j \\ 1+j & 1-j \end{bmatrix} \tag{36}$$

Likewise, for equation (24),

$$T_{z,m} = T_z V_m \tag{37}$$

Now from equations (26), (33), and (35) and the definition

$$D_B = V_m^{-1} D V_m \tag{38}$$

T_R can be written

$$T_R = T_{mm} D_B T_{z,m}^{-1}$$

Now if T_R is real, D_B, a block diagonal matrix, must be real. This can be assured by selecting the elements of D, d_i, such that

(1) d_i is real if λ_i is real

(2) d_i, d_{i+1} are a complex conjugate pair if λ_i, λ_{i+1} are a complex conjugate pair.

The reciprocal transformation of equation (17) does not represent a significant increase in computations since the modal transformation is the same for A and $\hat{A}$ and the eigenvalues are simply reciprocals.

The matrix inversion of equation (26) can be eliminated by noting that

$$\Gamma = \tilde{T}\, \Gamma^T\, \tilde{T} \tag{40}$$

where

$$\tilde{T} = \mathrm{diag}[(-1)^{i-1}] \tag{41}$$

From equation (19) then

$$\Lambda_R \tilde{T} R = R^T \Lambda_R \tilde{T} \tag{42}$$

Now from equations (24), (26), and (40)

$$T_R = T_m D_m T_z^T\, \tilde{T} \Lambda_R \tag{43}$$

A program that solves for T_R, the Routh transformation in equation (43), was written. This program, called DRTTRC, not only finds T_R, but also the transformed matrices of equation (28), G and H. The FORTRAN listing of DRTTRC is given in appendix B. Additionally, a program that finds the reduced system of equation (32), called DTYPEI, was written and is given in appendix B.

APPLICATION TO ENGINE MODEL

The frequency-domain formulation of the Routh approximation technique was applied to the state-variable model of the F100 engine at the intermediate-power operating point and to the transfer-function model of the boost-pump pressure regulator design. The time-domain formulation was applied only to the engine model.

Frequency-Domain Applications

As an initial experiment the frequency-domain formulation was applied to two 16th-order SISO transfer functions. These transfer functions represent dynamics of the turbofan engine from the input w_f (fuel flow) to the outputs N_c (compressor speed) and T_t (turbine inlet temperature). The coefficients for these transfer functions were calculated and are given in table I. Table II shows the impulse response ratios for Routh approximations of increasing order for the two transfer functions. These ratios were calculated from the alpha and beta coefficients as outlined in equation (12). From these ratios an acceptable approximation order can be estimated.

The approximant order was selected by first choosing a minimum acceptable level of accuracy, as defined by the impulse response energy ratio. The level chosen corresponds to a ratio of 0.81. This level of accuracy was assumed to be adequate for the purpose of this report. Next the order that most nearly corresponded to the selected ratio was chosen as the order of the Routh approximant. For the energy ratios given in table II, the Routh approximant orders, as found by these criteria, are

$$k_1 = 3 \qquad (y_1 = N_c)$$

$$k_2 = 9 \qquad (y_2 = T_t)$$

The Routh approximants for N_c and T_t were calculated by using the approximant orders k_1 and k_2 and the algorithm given in equations (6) to (9). The results are summarized in table III. Comparisons of pole locations and step responses for exact and approximate transfer functions were made to evaluate the adequacy of the approximations. Table IV gives the pole locations, and figure 1 the step response comparisons. The pole comparison shows a good correlation between actual and approximate system poles. Likewise, the step responses of figure 1 indicate an excellent agreement between actual and approximate model representations. Based on these results it was concluded that the Routh approximation technique is a viable approach to reducing the complexity of frequency-domain models of jet engine dynamics. Also, the accuracy level selected in this initial example may be too stringent, based on the step response comparison.

For comparison, the Routh approximation technique was applied to all 80 ($l = 16$, $m = 5$) possible transfer functions of the given engine example. The impulse response energy ciriterion for model order was used to determine the order for each of the approximants. These reduced orders are given in table V. Examination of this table gives dynamically fast (x_3, x_4, e.g.) and slow (x_1, x_2, x_{10}, e.g.) states as well as reduced order. The large difference in reduced order for transfer functions with the same input indicates that modeling of the fast states as constants with respect to the slow states would be acceptable.

The frequency response technique was also applied to a high-order (43rd order) model of the boost-pump pressure regulator, which represents the linearized line dynamics of a liquid-oxygen supply system in a rocket engine (ref. 8). The model input represents a pressure error signal; the output represents an actuator piston-commanded velocity. The transfer function (time scaled by a factor of 100) for this system is given in table VI. This application tests the accuracy and the computational feasibility of the programs DROUTH, DTHCNV, and DIMPLS for high-order problems. Again, straightforward application of the programs was all that was required to obtain Routh approximants to the boost-pump pressure regulation system. Additionally, it was determined that a 43rd-order implementation of this pressure control would be impractical and that a study of lower order implementations would be desirable. Thus, this application is of more than numerical significance.

For the 81-percent impulse response energy criterion, a 26th-order Routh approximant was selected. The poles of the original and approximant system are compared in table VII; the zeros are compared in table VIII. Both show good comparability between original and approximant values. For reference the 26th-order approximant is given in table IX. Figure 2 compares the frequency responses for the original and approximant systems. The comparison is exact for both magnitude and phase for frequencies less than 20 hertz. The approximation is still good for higher frequencies, and the loss of frequency information due to the approximation is clearly shown.

Time-Domain Application

The time-domain formulation of the Routh application technique was applied to the 16th-order, state-variable F100 engine model. This model represents the dynamic response of the F100 engine for small perturbations about the intermediate-power operating point. The normalized system matrices are given in table X and the state, input, and output vector definitions in table XI. Calculating the Routh transformation matrix T_R poses the only significant computational problem in applying the time-domain formulation. Thus, as a test, the alpha parameters calculated from the characteristic equation of $\hat{A}$ (Routh table approach) and from the similarity transformation $T_R^{-1}AT_R$ are compared in table XII. Clearly, calculating T_R with DRTTRC is accomplished with a high degree of accuracy. Therefore, the tools exist to accurately apply the time-domain formulation to high-order systems.

Examining table V shows that a fifth-order approximation for each input would represent an adequate tradeoff between accuracy and complexity. Reduced approximants were calculated by using DRTTRC and DTYPEI for this order. Step response trajectories comparing the original and approximate systems were generated. The T_t result indicates the degradation of accuracy obtained with a lower order model. Typical step responses for one input (fuel flow) and two outputs (compressor speed and turbine inlet temperature) are given in figure 3.

With this time-domain formulation, fifth-order models were calculated for each input that exactly match those that would have been determined with a frequency-domain analysis. However, the approximants were determined directly from the state-variable model without first calculating the frequency transfer function for each input-output pair. All the properties of the frequency-domain approach hold for the time-domain approach when considering the state-variable formulation one input at a time.

Fixed-Dimension Approximant

For a five-input system approximated by fifth-order models, a total reduced system realization may require a state vector dimension of 25. In fact, the realization dimension will probably vary for different numbers of inputs and orders of approximation. Often, however, the allowable dimension of the total reduced approximation is fixed by some engineering or economic constraint. Constructing a fixed-dimension realization of the total system from the Routh approximation can be a very difficult computational problem, especially if the input and output dimensions are large. Thus, the technique as outlined does not directly handle the fixed-dimension problem. However, if the same T_R matrix is used in the time-domain reduction process for each input, the total approximation can be realized by a system of k^{th} order, where k is the number of alpha parameters retained, and the fixed-dimension approximation problem is solved. Selecting the T_R matrix that will give the best k^{th}-order approximation is therefore an interesting problem.

In this regard the somewhat arbitrary selection of D can be used to good advantage. Many different transformations can be found very quickly once the original two eigenvector problems have been solved. Indeed, the selection process can be automated by optimizing some function of error between the original and total approximant systems over the n-parameter space of D. Such an optimization scheme was tried on the example given in this report. Two different error functions were used for comparison. The first was a weighted sum of the differences in system and approximant step response energies. The second was a weighted sum of squares of the differences in system and approximant steady-state values. Significant minimization of each error function was easily achieved in relatively small amounts of computer time by using a conjugate direction optimization scheme. Thus, the general optimization procedure would appear to be a good way to improve the accuracy of approximants for certain systems while maintaining a fixed order of realization. However, the time-domain Routh approximation procedure, when constrained to yield a fixed-dimension realization in the multiple-input case, does not exhibit the final-value property of the single-input case. For the engine example posed, the significant improvement in the approximation gained by function optimization was overshadowed by these final-value errors for the multiple-input case.

Based on this observation the time-domain formulation was modified to ensure that the final-value property would be met by a fixed-dimension, multiple-input Routh approximant. In the original formulation the difficulty with final values in the multiple-input, fixed-dimension problem can be traced to the original assumption of the reduction process (eq. (30)). If the initial assumption were changed to

$$\left(\dot{x}_R\right)_2 = 0 \tag{44}$$

the approximation would force the final-value property. Unfortunately, the assumption of equation (44) when applied to the original system may yield an unstable approximant for a given stable system. This was the case for the engine example and, thus, the modification of the Routh procedure was rejected.

CONCLUSIONS

Models of physical processes are often too complex to handle directly. Approximations of these models that are of reduced complexity but that maintain essential model characteristics are often used for analysis rather than the original model. One way to generate these approximations is the Routh approximation technique. The frequency-domain formulation of the Routh technique was applied to transfer-function models of an F100 engine and a launch vehicle boost-pump pressure regulator. Also, the Routh approximation process was reformulated in the time domain. A new characterization of the nonunique Routh similarity transformation was derived that describes all possible Routh transformations. This characterization casts the computation of the Routh transformation into two eigenvector-eigenvalue problems that are easily solved. The application of the time-domain formulation to a 16th-order state-variable description of a turbofan engine was described, and the results were given. These results indicate that the time-domain Routh approximation technique can be valuable in reducing engine model complexity when dealing with the model on a single-input basis. An optimization procedure was discussed that can significantly improve the approximation in a computationally efficient manner by taking advantage of the new time-domain characterization.

Lewis Research Center,
National Aeronautics and Space Administration,
Cleveland, Ohio, May 11, 1977,
505-05.

APPENDIX A

SYMBOLS

A	reciprocal system matrix
$\hat{A}$	original system matrix
a	coefficient of characteristic equation
B	reciprocal control matrix
$\hat{B}$	original control matrix
B_i	matrix of numerator coefficients
b	element of B_i
C	reciprocal output matrix
$\hat{C}$	original output matrix
c	coefficient of characteristic equation for reduced system
D	diagonal parameter matrix
d	coefficient of numerator polynomial for reduced system
d^i/ds^i	i^{th}-order differentiation operator
E	impulse response energy
E_p	permutation matrix
G	Routh control matrix
H	Routh output matrix
$\mathscr{h}$	impulse response
h_i	number of rows
K	reduced-system order
k	Routh approximant order
l	number of outputs
m	number of inputs
N_c	compressor speed
n	original-system order
R	Routh-system matrix
s	Laplace transform variable

T	similarity transformation matrix
T_t	turbine inlet temperature
t	time, sec
u	control vector
V	block diagonal modification matrix
$W(s)$	matrix transfer function
w_f	engine fuel flow
x	reciprocal state vector
$\hat{x}$	original state vector
y	output vector
α	alpha parameter
β	beta parameters
Γ	gamma matrix
λ	diagonal parameter matrix
Σ	state-space system

Subscripts:

k	k^{th}-order approximant
m	modal
mm	modified modal
p	output number
q	input number
R	Routh transformation
z	Routh transformed system

APPENDIX B

SUBROUTINES USED IN ROUTH APPROXIMATION

```
C       SUBROUTINE ROUTH
C
C       PURPOSE
C           TO CALCULATE THE ALPHA-BETA EXPANSION
C           COEFFICIENTS OF A LINEAR, TIME-INVARIANT
C           TRANSFER FUNCTION(SISO)
C
C       METHOD
C           ROUTH TABLE
C
        SUBROUTINE DROUTH(Z1,Z2,ALPHA,BETA,RTA,RTB,N,NMAX)
        IMPLICIT DOUBLE PRECISION (A-H,O-Z)
        DIMENSION Z1(1),Z2(1),ALPHA(1),BETA(1),RTA(NMAX,1),RTB(NMAX,1)
        KK=0
        DO 5 I=1,N
        DO 5 J=1,N
        RTA(I,J)=0.
        RTB(I,J)=0.
      5 CONTINUE
C
C       INITIALIZE THE ALPHA AND BETA ARRAYS
C
        IF(MOD(N,2).EQ.0) N1=N/2
        IF(MOD(N,2).EQ.1) N1=(N-1)/2
        DO 10 J=1,N1
        DO 10 I=1,2
        NN=N-KK
        RTA(I,J)=Z1(NN)
        RTB(I,J)=Z2(NN)
        KK=KK+1
     10 CONTINUE
        N2=N1+1
        RTA(1,N2)=1.
        RTB(2,N2)=1.
        IF(NN.EQ.1) RTA(2,N2)=0.
        IF(NN.EQ.2) RTA(1,N2)=Z1(1)
        IF(NN.EQ.1) GO TO 20
        RTB(1,N2)=Z2(1)
        RTB(2,N2)=0.
     20 CONTINUE
C
C       CALCULATE THE ALPHA AND BETA ARRAYS
C
        DO 30 I=1,N
        DO 30 J=1,N1
        ALPHA(I)=RTA(I,1)/RTA(I+1,1)
        BETA(I)=RTB(I,1)/RTA(I+1,1)
        RTA(I+2,J)=RTA(I,J+1)-ALPHA(I)*RTA(I+1,J+1)
        RTB(I+2,J)=RTB(I,J+1)-BETA(I)*RTA(I+1,J+1)
     30 CONTINUE
        RETURN
        END
```

```
C      SUBROUTINE IMPULS
C
C      PURPOSE
C          TO COMPUTE THE ENERGY ASSOCIATED WITH EACH
C          ROUTH APPROXIMANT ASSUMING THE SYSTEM IS
C          DRIVEN WITH AN IMPULSE
C
       SUBROUTINE DIMPLS(ALPHA,BETA,RMS,N)
       IMPLICIT DOUBLE PRECISION (A-H,O-Z)
       DIMENSION ALPHA(1),BETA(1),RMS(1)
       SUM=0.0
       DO 10 I=1,N
       SUM=BETA(I)**2/(2.*ALPHA(I))+SUM
       IF(SUM.LT.0.0) RETURN
       RMS(I)=DSQRT(SUM)
    10 CONTINUE
       RETURN
       END


C      SUBROUTINE MRTTRC
C
C      PURPOSE
C          TO FORM THE STATE SPACE ROUTH APPROXIMATE
C          OF THE ORDER NHAT FROM THE ORIGINAL RECIPROCAL
C          SYSTEM AND THE ROUTH TRANSFORMATION MATRIX
C
       SUBROUTINE DRTTRC(ALFA,XI,B,C,T,TZ,TR,D,BHT,CHT,WA,WB,WC,
      1 N,M,MM,NHAT)
       IMPLICIT DOUBLE PRECISION (A-H,O-Z)
       DIMENSION ALFA(N),B(N,M),C(MM,N),BHT(NHAT,M),CHT(MM,NHAT)
      1 ,WA(N,N),WB(N,N),WC(MM,N),T(N,N),TZ(N,N),TR(N,N),D(N),XI(N)
C
C      FORM THE ROUTH TRANSFORMATION MATRIX TR
C
       IJ=0
       DO 1 J=1,N
       IF(IJ.EQ.1) GO TO 6
       IF(XI(J).NE.0.0) GO TO 5
       DO 11 I=1,N
       WB(I,J)=TZ(I,J)
    11 WA(I,J)=T(I,J)*D(J)
       GO TO 1
     5 DO 12 I=1,N
       WB(I,J)=TZ(I,J+1)
       WB(I,J+1)=TZ(I,J)
       WA(I,J)=T(I,J)*D(J)+T(I,J+1)*D(J+1)
    12 WA(I,J+1)=-T(I,J)*D(J+1)+T(I,J+1)*D(J)
       IJ=1
       GO TO 1
     6 IJ=0
     1 CONTINUE
       CALL DTMULT(WA,WB,TR,N,N,N)
       IJK=-1
       DO 2 J=1,N
       IJK=-IJK
       ALFD=IJK*ALFA(J)
```

```
      DO 2 I=1,N
    2 TR(I,J)=TR(I,J)*ALFD
C
C     FORM CHAT
C
      CALL DMULT(C,TR,WC,MM,N,N)
      DO 3 I=1,MM
      DO 3 J=1,NHAT
    3 CHT(I,J)=WC(I,J)
C
C     FIND THE INVERSE OF THE TR MATRIX
C
      IJ=0
      CALL DMTMPY(WA,WB,1.D0,N,N,N)
      DO 7 J=1,N
      IF(IJ.EQ.1) GO TO 72
      IF(XI(J).EQ.0.D0) GO TO 7
      DO 71 I=1,N
      WB(I,J)=WA(I,J+1)
   71 WB(I,J+1)=WA(I,J)
      IJ=1
      GO TO 7
   72 IJ=0
    7 CONTINUE
      CALL DTMULT(WB,TZ,WA,N,N,N)
      IJK=-1
      DO 8 J=1,N
      IJK=-IJK
      ALFD=ALFA(J)*IJK
      DO 8 I=1,N
    8 WA(I,J)=WA(I,J)*ALFD
      CALL DMULT(WA,TZ,WB,N,N,N)
      IDGT=6
      CALL DLINVIF(WB,N,N,WA,IDGT,WC,IER)
      CALL DMULT(TZ,WA,WB,N,N,N)
C
C     FORM BHAT
C
      CALL DMULT(WB,B,WA,N,N,N)
      DO 4 I=1,NHAT
      DO 4 J=1,M
    4 BHT(I,J)=WA(I,J)
      RETURN
      END



C     SUBROUTINE RTHCNV
C
C     PURPOSE
C          TO COMPUTE THE COEFFICIENTS OF THE ROUTH
C          CONVERGENT APPROXIMANTS
C
      SUBROUTINE DTHCNV(ALPHA,BETA,NMMTX,DNMTX,N,NMAX)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DOUBLE PRECISION NMMTX
      DIMENSION ALPHA(1),BETA(1),NMMTX(NMAX,1),DNMTX(NMAX,1)
```

```
      DO 10 I=1,NMAX
      DO 10 J=1,NMAX
      NMMTX(I,J)=0.
      DNMTX(I,J)=0.
   10 CONTINUE
C
C     THE DENOMINATOR CONVERGENTS ARE IN DNMTX
C     THE NUMERATOR CONVERGENTS ARE IN NMMTX
C
      DO 20 I=1,N
      DNMTX(1,I)=1.0
   20 CONTINUE
      DNMTX(2,1)=ALPHA(1)
      DNMTX(2,2)=ALPHA(2)
      DNMTX(3,2)=ALPHA(1)*ALPHA(2)
      NMMTX(2,1)=BETA(1)
      NMMTX(2,2)=BETA(2)
      NMMTX(3,2)=ALPHA(2)*BETA(1)
      DO 30 I=3,N
      DO 30 J=2,N
      DNMTX(J,I)=DNMTX(J-1,I-1)*ALPHA(I)+DNMTX(J,I-2)
      NMMTX(J,I)=NMMTX(J-1,I-1)*ALPHA(I)+NMMTX(J,I-2)
      IF(J.EQ.2) NMMTX(J,I)=NMMTX(J,I)+BETA(I)
   30 CONTINUE
      DO 40 I=2,N
      DO 40 J=1,N
      DNMTX(I-1,J)=DNMTX(I,J)
      NMMTX(I-1,J)=NMMTX(I,J)
   40 CONTINUE
      DNMTX(N,N-1)=0.
      NMMTX(N,N-1)=0.
      PROD=1.
      DO 50 I=1,N
      PROD=PROD*ALPHA(I)
   50 CONTINUE
      DNMTX(N,N)=PROD
      NMMTX(N,N)=BETA(1)*PROD/ALPHA(1)
      RETURN
      END




C     SUBROUTINE DTYPEI
C
C     PURPOSE
C         TO FIND THE KTH ORDER TYPE I DOMINANT MODE
C         APPROXIMATION OF AN NTH ORDER SYSTEM
C
C         THE ORIGINAL SYSTEM
C               ZDOT = T*R*INV(T)*Z + T*G*U
C               Y = H*INV(T)*Z
C
      SUBROUTINE DTYPEI(A,B,C,T,TINV,R11,G,H,DKK1,DKK2,DNN1,N,M,L,K)
      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      DIMENSION A(K,K),B(K,M),C(L,K),T(N,N),R11(K,K),G(N,M),H(L,N)
     1 ,DKK1(K,K),DKK2(K,K),DNN1(N,N),TINV(N,N)
      IF(M.GT.K) WRITE(6,1000)
```

```
1000 FORMAT(1H0,10X,'THE CONTROL DIMENSION IS GREATER THAN THE'
     1  ,1X,'REDUCED ORDER DIMENSION IN DTYPEI')
      IDGT=6
C
C     FIND THE INVERSE OF THE K X K BLOCK OF T, T11
C
      DO 1 I=1,K
      DO 1 J=1,K
    1 DKK1(I,J)=T(I,J)
      CALL DMTMPY(DKK1,DKK2,1.D0,K,K,K)
      CALL DLINV1F(DKK2,K,K,A,IDGT,B,IER)
C
C     FIND THE CONTROL MATRIX P
C
      CALL DMULT(TINV,G,DNN1,N,M,N)
      DO 2 I=1,K
      DO 2 J=1,M
    2 DKK2(I,J)=DNN1(I,J)
      CALL DMULT(DKK1,DKK2,B,K,M,K)
C
C     FIND THE OUTPUT MATRIX C
C
      CALL DMULT(T,A,DNN1,N,K,K)
      CALL DMULT(H,DNN1,C,L,K,N)
C
C     FIND THE SIMILARITY TRANSFORM T11*R11*INV(T11)
C
      CALL DMULT(R11,A,DKK2,K,K,K)
      CALL DMULT(DKK1,DKK2,A,K,K,K)
      RETURN
      END
```

REFERENCES

1. Davison, Edward J.: A Method for Simplifying Linear Dynamic Systems. IEEE Trans. Autom. Control, vol. AC-11, no. 1, Jan. 1966, pp. 93-101.

2. Hutton, Maurice F.; and Friedland, Bernard: Routh Approximations for Reducing Order of Linear, Time-Invariant Systems. IEEE Trans. Autom. Control, vol. AC-20, no. 3, June 1975, pp. 329-338.

3. Hutton, M. F.: Routh Approximation Method for High-Order Linear Systems. Ph. D. Dissertation, Polytechnic Institute of Brooklyn, 1974.

4. Hutton, M.; and Rabins, M. J.: Simplification of High-Order Mechanical Systems Using the Routh Approximation. ASME Paper 75-WA/Aut-10, Nov./Dec. 1975.

5. Datta, B. N.: On the Similarity Between a Matrix and Its Routh Canonical Form. IEEE Trans. Autom. Control, vol. AC-20, no. 2, April 1975, pp. 273-274.

6. Seidel, Robert C.: Computer Programs for Calculation of Matrix Stability and Frequency Response from a State-Space System Description. NASA TM X-71581, 1974.

7. Geyser, Lucille C.; and Lehtinen, Bruce: Digital Program for Solving the Linear Stochastic Optimal Control and Estimation Problem. NASA TN D-7820, 1975.

8. Lehtinen, Bruce; and Lorenzo, Carl F.: Space Shuttle Active POGO Suppressor Control Design Using Linear Quadratic Regulator Techniques. NASA TP-1217, 1978.

TABLE I. - COEFFICIENTS FOR TWO 16th-ORDER ENGINE TRANSFER FUNCTIONS WITH FUEL FLOW INPUT

Coefficient, i	Coefficient of characteristic equation, a_i	Compressor speed, N_c ($p = 1$, $q = 1$)	Turbine inlet temperature, T_t ($p = 2$, $q = 1$)
		Element of matrix of numerator coefficients B_i, b_i	
0	1.0000	----------	----------
1	1.0638×10^{3}	0.1140	5.7270
2	3.7805×10^{6}	4.2916×10^{1}	5.8161×10^{3}
3	6.6915×10^{8}	-4.2414×10^{3}	1.8835×10^{6}
4	7.0216×10^{10}	-2.2082×10^{6}	2.9209×10^{8}
5	4.7772×10^{12}	-8.2163×10^{7}	2.6079×10^{10}
6	2.2150×10^{14}	2.2542×10^{10}	1.4714×10^{12}
7	7.1949×10^{15}	2.8702×10^{12}	5.5192×10^{13}
8	1.6584×10^{17}	1.5789×10^{14}	1.4142×10^{15}
9	2.7151×10^{18}	5.0230×10^{15}	2.5027×10^{16}
10	3.1255×10^{19}	1.0028×10^{17}	3.0507×10^{17}
11	2.4739×10^{20}	1.2850×10^{18}	2.5208×10^{18}
12	1.2973×10^{21}	1.0446×10^{19}	1.3685×10^{19}
13	4.2578×10^{21}	5.1525×10^{19}	4.6358×10^{19}
14	7.9983×10^{21}	1.4079×10^{20}	9.0249×10^{19}
15	7.4304×10^{21}	1.7841×10^{20}	8.7170×10^{19}
16	2.4119×10^{21}	7.4230×10^{19}	2.9263×10^{19}

TABLE II. - IMPULSE RESPONSE ENERGY RATIOS FOR INCREASING APPROXIMANT ORDER

[Fuel flow input.]

Routh approximant order, k	Compressor speed, N_c ($p = 1$, $q = 1$)	Turbine inlet temperature, T_t ($p = 2$, $q = 1$)
	Impulse response energy ratio	
1	0.41	0.007
2	.72	.03
3	.89	.07
4	.94	.14
5	.941	.23
6	.95	.36
7	.96	.51
8	.98	.67
9	.99	.81
10	.997	.91
11	.998	.97
12	.999	.99
13	.9992	.998
14	.9998	.999
15	.9999	.9999
16	1.0000	1.0000

TABLE III. - ROUTH APPROXIMANT COEFFICIENTS FOR TWO TRANSFER FUNCTIONS

Coefficient, i	Compressor speed, N_c (p = 1, q = 1)		Turbine inlet temperature, T_t (p = 2, q = 1)	
	Routh approximant coefficients			
	c_i	d_i	c_i	d_i
0	1.0000	-------	1.0000	-----------
1	2.6132	0.43978	4.7092	3.6223
2	2.5702	.61711	1.0446×10^{3}	9.3469×10^{1}
3	.8343	.25676	1.3858×10^{4}	1.3400×10^{3}
4	------	-------	1.1722×10^{5}	1.1906×10^{4}
5	------	-------	6.3304×10^{5}	6.6700×10^{4}
6	------	-------	2.1019×10^{6}	2.2877×10^{5}
7	------	-------	3.9634×10^{6}	4.4718×10^{5}
8	------	-------	3.6848×10^{6}	4.3229×10^{5}
9	------	-------	1.1961×10^{6}	1.4512×10^{5}

TABLE IV. - COMPARISON OF POLE LOCATIONS FOR EXACT (16th ORDER) AND APPROXIMATE (3rd AND 9th ORDER) TRANSFER FUNCTIONS

Poles of transfer functions		
16th Order	3rd-Order Routh approximant	9th-Order Routh approximant
-0.648	-0.635	-0.648
-1.91	-0.989±0.580j	-1.91
-2.62		-2.62
-6.71±1.31j		-6.50±1.16j
-17.8±4.80j		-7.64±3.29j
-18.2		-6.82±8.73j
-21.6±1.55j		
-38.7		
-47.1		
-50.6		
-59.2		
-175		
-577		

TABLE V. - MINIMUM TRANSFER FUNCTION ORDER FOR 81-PERCENT IMPULSE RESPONSE ENERGY RATIO

Transfer function output	Transfer function input				
	u_1	u_2	u_3	u_4	u_5
x_1	3	3	4	5	4
x_2	3	3	5	4	4
x_3	6	12	13	12	13
x_4	11	12	16	11	12
x_5	4	5	6	4	4
x_6	3	6	8	3	↓
x_7	3	8	7	5	
x_8	4	5	7	7	
x_9	8	8	7	8	8
x_{10}	4	2	2	2	4
x_{11}	9	9	7	9	9
x_{12}	8	8	7	8	8
x_{13}	7	3	5	4	6
x_{14}	7	8	7	7	6
x_{15}	4	7	7	5	4
x_{16}	6	10	9	11	10

TABLE VI. - BOOST-PUMP PRESSURE REGULATOR TRANSFER FUNCTION FOR PRESSURE ERROR MEASUREMENT INPUT P(s) TO ACTUATOR PISTON-COMMANDED VELOCITY OUTPUT Q(s)

$$\frac{Q(s)}{P(s)} = \left[\frac{-261.31\, s^{42} - 9821.8\, s^{41} - 139360\, s^{40} + \ldots + 0.34557\times10^{8}}{s^{43} + 530.64\, s^{42} + 1239.2\, s^{41} + 17914\, s^{40} + \ldots + 0.33972\times10^{8}}\right].$$

Power of s	Denominator coefficient	Numerator coefficient
42	0.53064×10^{2}	0.26131×10^{3}
41	$.12392\times10^{4}$	$-.98218\times10^{4}$
40	$.17914\times10^{5}$	$-.13936\times10^{6}$
39	$.17766\times10^{6}$	$-.13898\times10^{7}$
38	$.14079\times10^{7}$	$-.10524\times10^{8}$
37	$.91649\times10^{7}$	$-.68170\times10^{8}$
36	$.52194\times10^{8}$	$-.37098\times10^{9}$
35	$.25635\times10^{9}$	$-.18063\times10^{10}$
34	$.11467\times10^{10}$	$-.78084\times10^{10}$
33	$.45559\times10^{10}$	$-.30434\times10^{11}$
32	$.16623\times10^{11}$	$-.10919\times10^{12}$
31	$.55692\times10^{11}$	$-.35347\times10^{12}$
30	$.16873\times10^{12}$	$-.10737\times10^{13}$
29	$.48960\times10^{12}$	$-.29612\times10^{13}$
28	$.12415\times10^{13}$	$-.76725\times10^{13}$
27	$.31769\times10^{13}$	$-.18374\times10^{14}$
26	$.67590\times10^{13}$	$-.40591\times10^{14}$
25	$.15437\times10^{14}$	$-.85715\times10^{14}$
24	$.27519\times10^{14}$	$-.16043\times10^{15}$
23	$.56516\times10^{14}$	$-.30264\times10^{15}$
22	$.84066\times10^{14}$	$-.47440\times10^{15}$
21	$.15582\times10^{15}$	$-.80866\times10^{15}$
20	$.19203\times10^{15}$	$-.10439\times10^{16}$
19	$.32126\times10^{15}$	$-.16249\times10^{16}$
18	$.32473\times10^{15}$	$-.16879\times10^{16}$
17	$.48853\times10^{15}$	$-.24235\times10^{16}$
16	$.39945\times10^{15}$	$-.19629\times10^{16}$
15	$.53594\times10^{15}$	$-.26269\times10^{16}$
14	$.34801\times10^{15}$	$-.15896\times10^{16}$
13	$.41055\times10^{15}$	$-.20056\times10^{16}$
12	$.20670\times10^{15}$	$-.85553\times10^{15}$
11	$.20970\times10^{15}$	$-.10314\times10^{16}$
10	$.79450\times10^{14}$	$-.28689\times10^{15}$
9	$.67087\times10^{14}$	$-.33582\times10^{15}$
8	$.18474\times10^{14}$	$-.55134\times10^{14}$
7	$.12418\times10^{14}$	$-.63866\times10^{14}$
6	$.23847\times10^{13}$	$-.54308\times10^{13}$
5	$.11997\times10^{13}$	$-.63912\times10^{13}$
4	$.15079\times10^{12}$	$-.22981\times10^{12}$
3	$.51938\times10^{11}$	$-.28645\times10^{12}$
2	$.39472\times10^{10}$	$-.18961\times10^{10}$
1	$.66059\times10^{9}$	$-.37172\times10^{10}$
0	$.33972\times10^{8}$	$.34557\times10^{8}$

TABLE VII. - POLE COMPARISON BETWEEN APPROXIMANT AND ORIGINAL SYSTEMS

(a) Approximant system

Root	Real part of root	Imaginary part of root
1	-0.57680×10^{-1}	0
2	$-.17846\times10^{-2}$	.14393
3	$-.17846\times10^{-2}$	-.14393
4	$-.29297\times10^{-2}$	.29631
5	$-.29297\times10^{-2}$	-.29631
6	$-.51814\times10^{-1}$	.34438
7	$-.51814\times10^{-1}$	-.34438
8	$-.57957\times10^{-2}$	.52760
9	$-.57957\times10^{-2}$	-.52760
10	$-.43869\times10^{-1}$	.69008
11	$-.43869\times10^{-1}$	-.69008
12	$-.37413\times10^{-1}$	$-.10277\times10^{1}$
13	$-.37413\times10^{-1}$	$-.10277\times10^{1}$
14	$-.35326\times10^{-1}$	$.13448\times10^{1}$
15	$-.35326\times10^{-1}$	$-.13448\times10^{1}$
16	$-.23819\times10^{-1}$	$.14622\times10^{1}$
17	$-.23819\times10^{-1}$	$-.14622\times10^{1}$
18	$-.37308\times10^{-1}$	$.17055\times10^{1}$
19	$-.37308\times10^{-1}$	$-.17055\times10^{1}$
20	$-.38127\times10^{-1}$	$.22899\times10^{1}$
21	$-.38127\times10^{-1}$	$-.22899\times10^{1}$
22	$-.42813\times10^{1}$	0
23	$-.55877\times10^{1}$	$.43007\times10^{1}$
24	$-.55877\times10^{1}$	$-.43007\times10^{1}$
25	$-.44588\times10^{1}$	$.60078\times10^{1}$
26	$-.44588\times10^{1}$	$-.60078\times10^{1}$

(b) Original system

Root	Real part of root	Imaginary part of root	Root	Real part of root	Imaginary part of root
1	-0.57680×10^{-1}	0	23	-0.12750×10^{-1}	-0.17253×10^{1}
2	$-.17846\times10^{-2}$	.14393	24	$-.14813\times10^{-1}$	$.19714\times10^{1}$
3	$-.17846\times10^{-2}$	-.14393	25	$-.14813\times10^{-1}$	$-.19714\times10^{1}$
4	$-.29297\times10^{-2}$	.29631	26	$-.11560\times10^{-1}$	$.22272\times10^{1}$
5	$-.29297\times10^{-2}$	-.29631	27	$-.11560\times10^{-1}$	$-.22272\times10^{1}$
6	$-.51814\times10^{-1}$	.34438	28	$-.87273\times10^{-2}$	$.24548\times10^{1}$
7	$-.51814\times10^{-1}$	-.34438	29	$-.87273\times10^{-2}$	$-.24548\times10^{1}$
8	$-.57957\times10^{-2}$	.52760	30	$-.64893\times10^{-2}$	$.26466\times10^{1}$
9	$-.57957\times10^{-2}$	-.52760	31	$-.64893\times10^{-2}$	$-.26466\times10^{1}$
10	$-.43869\times10^{-1}$	.69008	32	$-.48184\times10^{-2}$	$.27998\times10^{1}$
11	$-.43869\times10^{-1}$	-.69008	33	$-.48184\times10^{-2}$	$-.27998\times10^{1}$
12	$-.37413\times10^{-1}$	$.10277\times10^{1}$	34	$-.36949\times10^{-2}$	$.29114\times10^{1}$
13	$-.37413\times10^{-1}$	$-.10277\times10^{1}$	35	$-.36949\times10^{-2}$	$-.29114\times10^{1}$
14	$-.32909\times10^{-1}$	$.13445\times10^{1}$	36	$-.30218\times10^{-2}$	$.29785\times10^{1}$
15	$-.32909\times10^{-1}$	$-.13445\times10^{1}$	37	$-.30218\times10^{-2}$	$-.29785\times10^{1}$
16	$-.14112\times10^{-1}$	$.14142\times10^{1}$	38	$-.44449\times10^{1}$	$.44417\times10^{1}$
17	$-.14112\times10^{-1}$	$-.14142\times10^{1}$	39	$-.44449\times10^{1}$	$-.44417\times10^{1}$
18	$-.12075\times10^{-1}$	$.15131\times10^{1}$	40	$-.91623\times10^{1}$	$.90855\times10^{1}$
19	$-.12075\times10^{-1}$	$-.15131\times10^{1}$	41	$-.91623\times10^{1}$	$-.90855\times10^{1}$
20	$-.18582\times10^{-1}$	$.16471\times10^{1}$	42	$-.42910\times10^{1}$	0
21	$-.18582\times10^{-1}$	$-.16471\times10^{1}$	43	$-.20926\times10^{2}$	0
22	$-.12750\times10^{-1}$	$.17253\times10^{1}$			

TABLE VIII. - ZERO COMPARISON BETWEEN APPROXIMANT AND ORIGINAL SYSTEMS

(a) Approximant system

Root	Real part of root	Imaginary part of root
1	0.91929×10^{-2}	0
2	$-.35351\times10^{-2}$	.14517
3	$-.35351\times10^{-2}$	-.14517
4	$-.29437\times10^{-2}$	.29626
5	$-.29437\times10^{-2}$	-.29626
6	$.92709\times10^{-2}$	.35815
7	$.92709\times10^{-2}$	-.35815
8	$-.72729\times10^{-2}$	.52853
9	$-.72729\times10^{-2}$	-.52853
10	$.67278\times10^{-2}$	.71196
11	$.67278\times10^{-2}$	-.71196
12	$.26794\times10^{-2}$	$.10536\times10^{1}$
13	$.26794\times10^{-2}$	$-.10536\times10^{1}$
14	$-.57825\times10^{-2}$	$.13715\times10^{1}$
15	$-.57825\times10^{-2}$	$-.13715\times10^{1}$
16	$-.67315\times10^{-1}$	$.14759\times10^{1}$
17	$-.67315\times10^{-1}$	$-.14759\times10^{1}$
18	$-.22702\times10^{-1}$	$.17112\times10^{1}$
19	$-.22702\times10^{-1}$	$-.17112\times10^{1}$
20	$.11499\times10^{-1}$	$.23645\times10^{1}$
21	$.11499\times10^{-1}$	$-.23645\times10^{1}$
22	$-.32228\times10^{1}$	.37158
23	$-.32228\times10^{1}$	-.37158
24	$-.30949\times10^{1}$	$-.40104\times10^{1}$
25	$-.30949\times10^{1}$	$.40104\times10^{1}$

(b) Original system

Root	Real part of root	Imaginary part of root	Root	Real part of root	Imaginary part of root
1	0.91929×10^{-2}	0	22	0.69556×10^{-3}	0.19874×10^{1}
2	$-.29437\times10^{-2}$	.29626	23	$.69556\times10^{-3}$	$-.19874\times10^{1}$
3	$-.29437\times10^{-2}$	-.29626	24	$-.15648\times10^{-2}$	$.22398\times10^{1}$
4	$.92709\times10^{-2}$	.35815	25	$-.15648\times10^{-2}$	$-.22398\times10^{1}$
5	$.92709\times10^{-2}$	-.35815	26	$-.26061\times10^{-2}$	$.24639\times10^{1}$
6	$-.72729\times10^{-2}$	.52853	27	$-.26061\times10^{-2}$	$-.24639\times10^{1}$
7	$-.72729\times10^{-2}$	-.52853	28	$-.29910\times10^{-2}$	$.26526\times10^{1}$
8	$.67278\times10^{-2}$	.71196	29	$-.29910\times10^{-2}$	$-.26526\times10^{1}$
9	$.67278\times10^{-2}$	-.71196	30	$-.30101\times10^{-2}$	$.28034\times10^{1}$
10	$.27081\times10^{-2}$	$.10536\times10^{1}$	31	$-.30101\times10^{-2}$	$-.28034\times10^{1}$
11	$.27081\times10^{-2}$	$-.10536\times10^{1}$	32	$-.29153\times10^{-2}$	$.29131\times10^{1}$
12	$-.66105\times10^{-2}$	$.13790\times10^{1}$	33	$-.29153\times10^{-2}$	$-.29131\times10^{1}$
13	$-.66105\times10^{-2}$	$-.13790\times10^{1}$	34	$-.28165\times10^{-2}$	$.29789\times10^{1}$
14	$-.14819\times10^{-1}$	$.14161\times10^{1}$	35	$-.28165\times10^{-2}$	$-.29789\times10^{1}$
15	$-.14819\times10^{-1}$	$-.14161\times10^{1}$	36	$-.35351\times10^{-2}$	.14517
16	$-.51392\times10^{-1}$	$.15083\times10^{1}$	37	$-.35351\times10^{-2}$	-.14517
17	$-.51392\times10^{-1}$	$-.15083\times10^{1}$	38	$-.31477\times10^{1}$	0
18	$-.23310\times10^{-1}$	$.16573\times10^{1}$	39	$-.39426\times10^{1}$	0
19	$-.23310\times10^{-1}$	$-.16573\times10^{1}$	40	$-.46850\times10^{1}$	$.43838\times10^{1}$
20	$.25798\times10^{-2}$	$.17352\times10^{1}$	41	$-.46850\times10^{1}$	$-.43838\times10^{1}$
21	$.25798\times10^{-2}$	$-.17352\times10^{1}$	42	$-.20927\times10^{2}$	0

TABLE IX. - 26th-ORDER

ROUTH APPROXIMANT

$$\left[\frac{Q}{P} = \frac{\sum_{i=1}^{26} d_i s^{26-i}}{s^{26} + \sum_{i=1}^{26} c_i s^{26-i}}, \quad c_0 = 1.\right]$$

Coefficient, i	Routh approximant coefficients	
	d_i	c_i
1	-0.205702×10^{3}	0.249887×10^{2}
2	$-.262997\times10^{4}$	$.320663\times10^{3}$
3	$-.190677\times10^{5}$	$.248369\times10^{4}$
4	$-.884145\times10^{5}$	$.129942\times10^{5}$
5	$-.315100\times10^{6}$	$.485403\times10^{5}$
6	$-.954589\times10^{6}$	$.151935\times10^{6}$
7	$-.228406\times10^{7}$	$.398134\times10^{6}$
8	$-.481475\times10^{7}$	$.814672\times10^{6}$
9	$-.887488\times10^{7}$	$.168262\times10^{7}$
10	$-.129769\times10^{8}$	$.234334\times10^{7}$
11	$-.199129\times10^{8}$	$.398045\times10^{7}$
12	$-.195556\times10^{8}$	$.383160\times10^{7}$
13	$-.263469\times10^{8}$	$.540632\times10^{7}$
14	$-.164054\times10^{8}$	$.359295\times10^{7}$
15	$-.202690\times10^{8}$	$.416608\times10^{7}$
16	$-.739529\times10^{7}$	$.189273\times10^{7}$
17	$-.870168\times10^{7}$	$.175577\times10^{7}$
18	$-.170706\times10^{7}$	$.539769\times10^{6}$
19	$-.197216\times10^{7}$	$.386031\times10^{6}$
20	$-.186761\times10^{6}$	$.789457\times10^{5}$
21	$-.218839\times10^{6}$	$.411942\times10^{5}$
22	$-.840811\times10^{4}$	$.534361\times10^{4}$
23	$-.103948\times10^{5}$	$.188562\times10^{4}$
24	$-.726830\times10^{2}$	$.144366\times10^{3}$
25	$-.138053\times10^{3}$	$.245338\times10^{2}$
26	$.128341\times10^{1}$	$.126168\times10^{1}$

TABLE X. - ENGINE

[n = 16, 1 = 2,

(a) A

-4.32800	0.218859	0.200384	3.95571	-2.98299	-0.144833	0.769294×10^{-1}	-0.144819
-.344744	-5.64300	3.72188	-1.80121	-1.40020	1.56028	.120495	-.299362
27.8479	208.042	-165.000	-1.18466	115.858	-165.738	-10.7525	20.1663
53.8484	-14.0784	496.865	-578.300	42.6308	-70.2873	-8.77498	-36.0530
2.05891	-4.84795	.507217	3.76362	-10.0500	3.55273	-.161294	-.755865
11.1442	-.212853	-.177448×10^{-1}	-.798490×10^{-1}	2.06863	-19.7900	-.182496	-.582995×10^{-1}
8.97363	-.213997	-.478545×10^{-1}	-.465845×10^{-1}	4.37978	-.818499×10^{-1}	-20.4700	-.768059×10^{-1}
-.657472	7.50337	4.26387	.701969×10^{-1}	-2.85571	15.0693	.303936	-19.9700
-.347415×10^{-1}	-.826121×10^{-1}	.272257×10^{-1}	.844885×10^{-2}	-.137080	.207504	.168996×10^{-1}	3.86258
-.461982×10^{-2}	-.110436×10^{-1}	.360582×10^{-2}	.112675×10^{-2}	-.182567×10^{-1}	.277078×10^{-1}	.225353×10^{-2}	.515019
-4.09217	-29.0787	3.31853	.417099	-17.7117	26.1649	1.91267	18.4196
-.143460	-3.04565	-10.3264	10.2900	-.607676	.883145	.711020×10^{-1}	7.55591
-.570369^{-1}	-1.21806	-4.13034	4.11573	-.242984	.353474	.287463×10^{-1}	3.02213
-.908447	-1.77558	-5.04877	1.29700	3.39206	.691218	.588312×10^{-1}	3.72598
-.163698	.507438	-.587604×10^{-1}	-.624872×10^{-2}	.307427	-.446147	19.6607	.242382
-11.9372	16.8839	-1.02879	8.06630	10.1496	-14.7933	-1.11110	-.656673

(b) B matrix

-0.469024×10^{-1}	-0.124675
.895387×10^{-1}	-.118071
5.92813	10.0878
33.9885	5.82984
2.48005	-6.59813
.373669	.263890
.233509	.262489
-.538716	-.264328
.156463	-.841755×10^{-2}
.208467×10^{-1}	-.112409×10^{-2}
19.9275	-1.63330
.750732	-.352531×10^{-1}
.299580	-.141273×10^{-1}
.391725	.258106×10^{-1}
.253898×10^{-1}	.332296×10^{-1}
1.24292	.868843

[a]For p = 16, C is the 16th-order identity matrix.

MODEL DATA

$q = 2^a]$

matrix

0.252614	0.427346×10^{-1}	-1.19810	1.82058	0.213133	-0.106835	-0.997036×10^{-2}	0.132963×10^{-1}
2.24174	.239841	0.940377	$-.855296\times10^{-2}$	$-.100398\times10^{-1}$	$-.104123\times10^{-1}$	$-.353279\times10^{-1}$	$-.104131\times10^{-1}$
$-.636953\times10^{-1}$	.101911	44.6143	.114667	.101938	.114658	2.24267	.101904
-79.0958	-12.9188	170.896	-4.53242	-4.53157	-4.43444	-3.61533	-4.53105
.128376	-.358717	-2.92106	-.424440	-.420034	1.47850	$.922059\times10^{-1}$	-.422479
$-.659112\times10^{-1}$	$-.608226\times10^{-1}$	$-.253476\times10^{-1}$	$-.595834\times10^{-1}$	$-.621119\times10^{-1}$	$-.595750\times10^{-1}$	$-.380239\times10^{-2}$	$-.621158\times10^{-1}$
$.629645\times10^{-1}$	$-.793110\times10^{-1}$	$-.365148\times10^{-1}$	$-.793274\times10^{-1}$	$-.818338\times10^{-1}$	$-.793363\times10^{-1}$	$.894036\times10^{-1}$	$-.818498\times10^{-1}$
$.386312\times10^{-1}$	$.341217\times10^{-1}$	$.167423\times10^{-1}$	$.341306\times10^{-1}$	$.367042\times10^{-1}$	$.341328\times10^{-1}$	$-.193187\times10^{-1}$	$.341318\times10^{-1}$
-49.9900	$.751111\times10^{-2}$	46.0276	$.375602\times10^{-2}$	$.751138\times10^{-2}$	$.375590\times10^{-2}$	0	$.751331\times10^{-2}$
-5.99940	-.665700	6.13815	$.450678\times10^{-3}$	$.101370\times10^{-2}$	$.450645\times10^{-3}$	0	$.101380\times10^{-2}$
.244677	.222139	-47.6500	.219747	.219719	.218157	-.111896	.219763
-12.7924	-1.41104	59.6425	-50.0100	$.112333\times10^{-1}$	$.112245\times10^{-1}$	$-.187131\times10^{-1}$	$.124796\times10^{-1}$
-5.11614	-.564075	23.8539	-18.0000	-1.99600	$.449175\times10^{-2}$	$-.718453\times10^{-2}$	$.494061\times10^{-2}$
-6.28623	-.688976	29.3366	-3.33640	-.359398	-19.7700	$-.213957\times10^{-1}$	$.117686\times10^{-1}$
$-.375007\times10^{-2}$	$-.249931\times10^{-2}$	$.712218\times10^{-1}$	$-.249937\times10^{-2}$	$-.374785\times10^{-2}$	$-.250001\times10^{-2}$	-20.0000	$-.374824\times10^{-2}$
.913777	$-.291566\times10^{-1}$	-5.14614	-.147456	-.164604	39.2746	10.9155	-50.1600

(c) C^T matrix

1.00000	0
0	
0	
0	
0	
0	
0	
0	
0	
0	
0	1.00000
0	0
0	
0	
0	
0	

TABLE XI. - INPUT, OUTPUT, AND STATE VECTOR DEFINITIONS

[Linear model: $\dot{x} = Ax + Bu$; $y = Cx$, where x are state variables, y are output variables, and u are input variables. Since $C = 1$ (identity matrix), $y = x$.]

State variables	Control variables
x_1 - Fan speed	u_1 - Main combustor fuel flow
x_2 - Compressor speed	u_2 - Nozzle jet area
x_3 - Compressor discharge pressure	u_3 - Inlet guide vane position
x_4 - Interturbine volume pressure	u_4 - High variable stator position
x_5 - Augmentor pressure	u_5 - Compressor bleed
x_6 - Fan inside-diameter discharge temperature	
x_7 - Duct temperature	
x_8 - Compressor discharge temperature	
x_9 - Combustor exit temperature - fast component	
x_{10} - Combustor exit temperature - slow component	
x_{11} - Combustor exit total temperature	
x_{12} - Fan turbine inlet temperature - fast component	
x_{13} - Fan turbine inlet temperature - slow component	
x_{14} - Fan turbine exit temperature	
x_{15} - Duct exit temperature	
x_{16} - Augmentor exit temperature	

TABLE XII. - COMPARISON OF COMPUTATIONAL ACCURACY OF FREQUENCY- AND TIME-DOMAIN FORMULATIONS

Alpha parameters	
From Routh table	From transformation $T_R^{-1} A T_R$
0.32460	0.32460
1.1230	1.1231
2.2886	2.2886
3.9662	3.9662
6.2954	6.2954
9.3914	9.3914
13.377	13.378
18.425	18.425
24.806	24.806
33.033	33.033
44.121	44.121
60.072	60.072
85.236	85.236
131.79	131.79
254.39	254.39
806.00	806.00

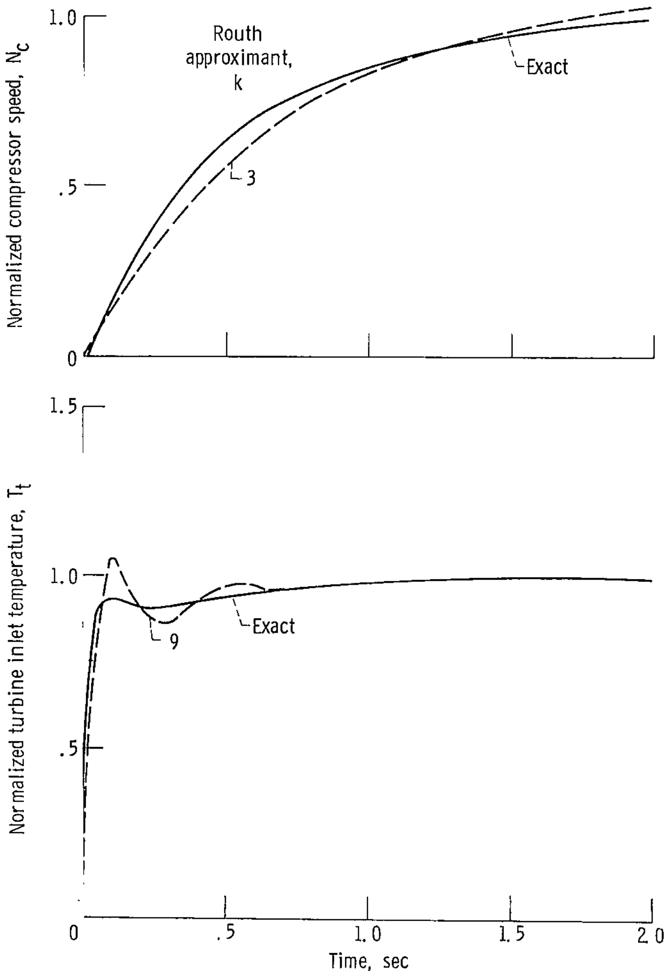

Figure 1. - Step response comparisons for normalized compressor speed and turbine inlet temperature outputs for fuel flow input - frequency-domain approach.

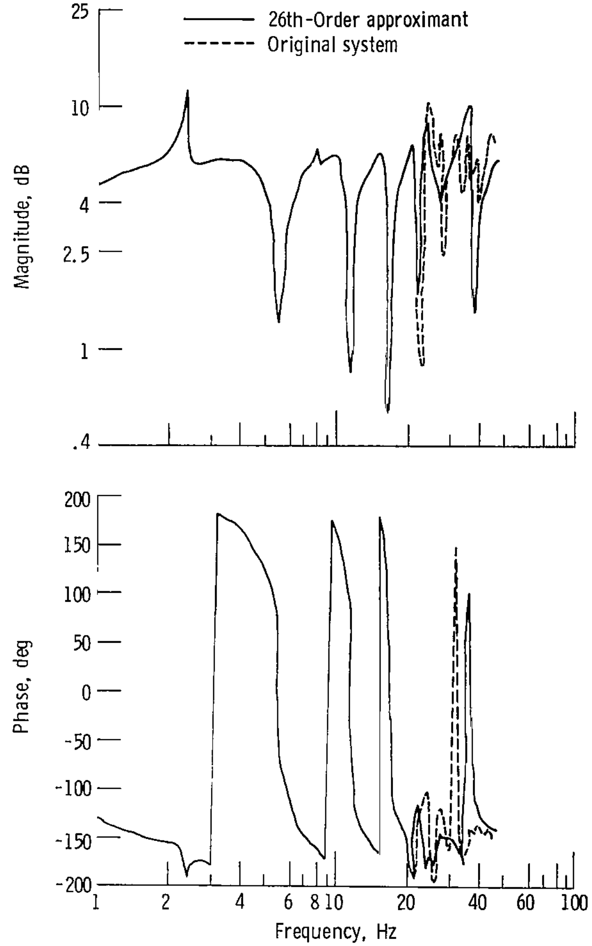

Figure 2. - Frequency response comparison.

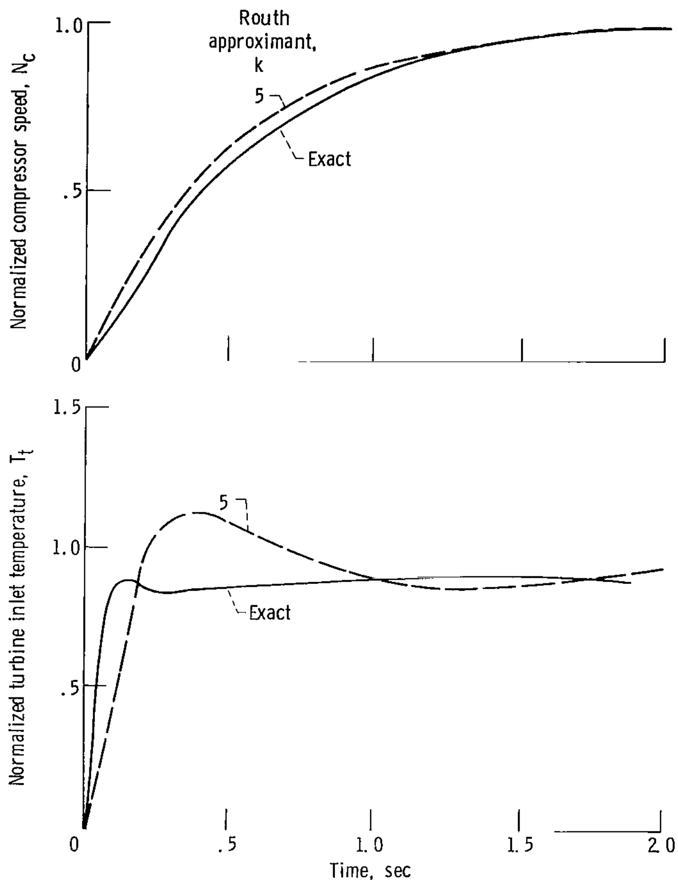

Figure 3. - Step response comparisons for normalized compressor speed and turbine inlet temperature outputs for fuel flow input - time-domain approach.

1. Report No. NASA TP-1231	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle APPLIED ROUTH APPROXIMATION		5. Report Date April 1978
		6. Performing Organization Code
7. Author(s) Walter C. Merrill		8. Performing Organization Report No. E-9114
		10. Work Unit No. 505-05
9. Performing Organization Name and Address National Aeronautics and Space Administration Lewis Research Center Cleveland, Ohio 44135		11. Contract or Grant No.
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, D.C. 20546		13. Type of Report and Period Covered Technical Paper
		14. Sponsoring Agency Code
15. Supplementary Notes		

16. Abstract

The Routh approximation technique for reducing the complexity of system models was applied in the frequency domain to a 16th-order, state-variable model of the F100 engine and to a 43rd-order, transfer-function model of a launch vehicle boost-pump pressure regulator. The results motivate extending the frequency-domain formulation of the Routh method to the time domain in order to handle the state-variable formulation directly. The time-domain formulation is derived and a new characterization that specifies all possible Routh similarity transformations is given. This characterization is computed by solving two eigenvalue-eigenvector problems. The application of the time-domain Routh technique to the state-variable engine model is described, and some results are given. Additional computational problems are discussed, including an optimization procedure that can improve the approximation accuracy by taking advantage of the transformation characterization.

17. Key Words (Suggested by Author(s)) Control theory; State variables; Approximation; Dynamic models; Transfer functions		18. Distribution Statement Unclassified - unlimited STAR Category 31	
19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages 38	22. Price* A03

* For sale by the National Technical Information Service, Springfield, Virginia 22161

NASA-Langley, 1978

National Aeronautics and Space Administration

Washington, D.C. 20546

Official Business
Penalty for Private Use, $300

THIRD-CLASS BULK RATE

Postage and Fees Paid
National Aeronautics and
Space Administration
NASA-451

U.S.MAIL

12 1 1U,D, 042078 S00903DS
DEPT OF THE AIR FORCE
AF WEAPONS LABORATORY
ATTN: TECHNICAL LIBRARY (SUL)
KIRTLAND AFB NM 87117

NASA

POSTMASTER: If Undeliverable (Section 158 Postal Manual) Do Not Return