Homotopy Approaches to the H_2 Reduced Order Model Problem

Dragan Zigic, Layne T. Watson,
Emmanuel G. Collins, Jr., and
Dennis S. Bernstein

TR 91-24

HOMOTOPY APPROACHES TO THE H_2 REDUCED ORDER MODEL PROBLEM

Dragan Žigić and Layne T. Watson
Department of Computer Science
Virginia Polytechnic Institute and State University
Blacksburg, VA 24061

Emmanuel G. Collins, Jr. and Dennis S. Bernstein
Harris Corporation, Government Aerospace Systems Division
Melbourne, FL 32902

703-231-7540, ltw@vtopus.cs.vt.edu

Abstract.

The optimal projection approach to solving the H_2 reduced order model problem produces two coupled, highly nonlinear matrix equations with rank conditions as constraints. The algorithms proposed herein utilize probability-one homotopy theory as the main tool. It is shown that there is a family of systems (the homotopy) that make a continuous transformation from some initial system to the final system. With a carefully chosen initial system all the systems along the homotopy path will be asymptotically stable, controllable and observable. One method, which solves the matrix equations in their original form, requires a decomposition of the projection matrix using the Drazin inverse of a matrix. An effective algorithm for computing the derivative of the projection matrix that involves solving a set of Sylvester equations is given. Another class of methods considers the matrix equations in a modified form, using a decomposition of the pseudogramians based on a contragredient transformation. Several strategies for choosing the homotopy maps and the starting points (initial systems) are discussed and compared, in the context of some reduced order model problems from the literature. Numerical results are included for ten test problems, of sizes 2 through 17.

Keywords: reduced order model problem, optimal projection equations, H_2 control, probability-one homotopy algorithm.

AMS subject classifications: 93B20, 93B25, 93B40, 65H10.

HOMOTOPY APPROACHES TO THE H_2 REDUCED ORDER MODEL PROBLEM

Dragan Žigić and Layne T. Watson
Department of Computer Science
Virginia Polytechnic Institute and State University
Blacksburg, VA 24061

Emmanuel G. Collins, Jr. and Dennis S. Bernstein
Harris Corporation, Government Aerospace Systems Division
Melbourne, FL 32902

Abstract.

The optimal projection approach to solving the H_2 reduced order model problem produces two coupled, highly nonlinear matrix equations with rank conditions as constraints. The algorithms proposed herein utilize probability-one homotopy theory as the main tool. It is shown that there is a family of systems (the homotopy) that make a continuous transformation from some initial system to the final system. With a carefully chosen initial system all the systems along the homotopy path will be asymptotically stable, controllable and observable. One method, which solves the matrix equations in their original form, requires a decomposition of the projection matrix using the Drazin inverse of a matrix. An effective algorithm for computing the derivative of the projection matrix that involves solving a set of Sylvester equations is given. Another class of methods considers the matrix equations in a modified form, using a decomposition of the pseudogramians based on a contragredient transformation. Several strategies for choosing the homotopy maps and the starting points (initial systems) are discussed and compared, in the context of some reduced order model problems from the literature. Numerical results are included for ten test problems, of sizes 2 through 17.

1. Introduction.

In [10] Hyland and Bernstein considered the quadratic (H_2) reduced order model problem, which is to find a reduced order model for a given continuous time stationary linear system which minimizes a quadratic model error criterion. The necessary conditions for the optimal reduced order model are given in the form of two modified Lyapunov equations, matrix equations which resemble the (linear) matrix Lyapunov equations, but are highly nonlinear and mutually coupled.

Among many different approaches for finding reduced order models are component cost analysis [18]–[20], balancing [15]–[16], Hankel-norm approximation [12]–[13], aggregation [1], [14], nonminimal partial realization [8] and the optimal reduction method of Wilson [28]. Some other applications of the optimal projection approach include the H_2/H_∞ model reduction problem [7], the fixed order dynamic compensation problem [9] and the reduced order state estimation problem [3]. A homotopy based algorithm for solving the fixed order dynamic compensation problem is given by Richter and Collins [17].

The complete statement of the reduced order model problem is given in Section 2. Section 3 explains the basics of probability-one homotopy theory and homotopy methods used for solving the reduced order model problem. Numerical results obtained by solving a number of model reduction problems are given in Section 4. Section 5 compares various homotopy approaches, and draws some conclusions.

2. Statement of the Problem.

Given the controllable and observable, time invariant, continuous time system

$$\dot{x}(t) = A\,x(t) + B\,u(t), \tag{1}$$
$$y(t) = C\,x(t), \tag{2}$$

where $A \in R^{n\times n}$, $B \in R^{n\times m}$, $C \in R^{l\times n}$, the goal is to find, for given $n_m < n$, a reduced order model

$$\dot{x}_m(t) = A_m\,x_m(t) + B_m\,u(t),$$
$$y_m(t) = C_m\,x_m(t),$$

where $A_m \in R^{n_m\times n_m}$, $B_m \in R^{n_m\times m}$, $C_m \in R^{l\times n_m}$, which minimizes the quadratic model-reduction criterion

$$J(A_m, B_m, C_m) \equiv \lim_{t\to\infty} E\left[(y - y_m)^t\,R\,(y - y_m)\right],$$

where the input $u(t)$ is white noise with positive definite intensity V and R is a positive definite weighting matrix.

It is assumed that A is asymptotically stable and diagonalizable, and a solution (A_m, B_m, C_m) is sought in the set

$$A_+ = \{(A_m, B_m, C_m):\ A_m \text{ is stable, } (A_m, B_m) \text{ is controllable and } (A_m, C_m) \text{ is observable}\}.$$

DEFINITION 1. Given symmetric positive semidefinite matrices $\hat{Q}$, $\hat{P} \in R^{n\times n}$ such that rank $(\hat{Q})$ = rank $(\hat{P})$ = rank $(\hat{Q}\hat{P})$ = n_m, matrices $G, \Gamma \in R^{n_m\times n}$ and positive semisimple $M \in R^{n_m\times n_m}$ are called a (G, M, Γ)-factorization (projective factorization) of $\hat{Q}\hat{P}$ if

$$\hat{Q}\,\hat{P} = G^t\,M\,\Gamma,$$
$$\Gamma\,G^t = I_{n_m}.$$

Positive semisimple means similar to a symmetric positive definite matrix.

The following theorem from [10] gives necessary conditions for the optimal solution to the reduced order model problem.

THEOREM 2. *Suppose $(A_m, B_m, C_m) \in A_+$ solves the optimal model-reduction problem. Then there exist symmetric positive semidefinite matrices $\hat{Q}$, $\hat{P} \in R^{n\times n}$ such that for some (G, M, Γ)-factorization of $\hat{Q}\hat{P}$, A_m, B_m and C_m are given by*

$$A_m = \Gamma\,A\,G^t, \tag{3}$$
$$B_m = \Gamma\,B, \tag{4}$$
$$C_m = C\,G^t, \tag{5}$$

and such that, with $\tau \equiv G^t\Gamma$ the following conditions are satisfied:

$$0 = \tau[A\,\hat{Q} + \hat{Q}\,A^t + B\,V\,B^t], \tag{6}$$
$$0 = [A^t\,\hat{P} + \hat{P}\,A + C^t\,R\,C]\,\tau, \tag{7}$$
$$\text{rank } (\hat{Q}) = \text{rank } (\hat{P}) = \text{rank } (\hat{Q}\,\hat{P}) = n_m. \tag{8}$$

The equations (6)–(7) can be written in an equivalent form

$$A\hat{Q} + \hat{Q}A^t + \tau B V B^t + B V B^t \tau^t - \tau B V B^t \tau^t = 0,$$
$$A^t \hat{P} + \hat{P}A + \tau^t C^t R C + C^t R C \tau - \tau^t C^t R C \tau = 0.$$

The matrices $\hat{Q}$ and $\hat{P}$ are called the *controllability* and *observability pseudogramians*, respectively, since they are analogous to the Gramians G_c and G_o which satisfy the dual Lyapunov equations

$$A G_c + G_c A^t + B V B^t = 0,$$
$$A^t G_o + G_o A + C^t R C = 0.$$

τ is an oblique projection (idempotent) operator since $\tau^2 = \tau$. The projection matrix τ can be expressed as

$$\tau = (\hat{Q}\hat{P})(\hat{Q}\hat{P})^{\sharp},$$

where $(\hat{Q}\hat{P})^{\sharp}$ is the Drazin inverse [4].

The following theorem from [10], which is a special case of a result in [6], gives a sufficient condition for simultaneous reduction of two symmetric positive semidefinite matrices to a diagonal form using a contragredient transformation.

THEOREM 3. [10] *Let symmetric positive semidefinite Q, $P \in R^{n\times n}$ satisfy*

$$\text{rank}\,(Q) = \text{rank}\,(P) = \text{rank}\,(QP) = n_m, \tag{9}$$

where $n_m \leq n$. Then, there exists a nonsingular $W \in R^{n\times n}$ (contragredient transformation) and positive definite diagonal Σ, $\Omega \in R^{n_m \times n_m}$ such that

$$Q = W \begin{pmatrix} \Sigma & 0 \\ 0 & 0 \end{pmatrix} W^t, \quad P = W^{-t} \begin{pmatrix} \Omega & 0 \\ 0 & 0 \end{pmatrix} W^{-1}. \tag{10}$$

REMARK 4. [10] *Let Q and P be as in Theorem 3. Then there exists a nonsingular $U \in R^{n\times n}$ and positive definite diagonal $\Lambda \in R^{n_m \times n_m}$ such that*

$$Q = U \begin{pmatrix} \Lambda & 0 \\ 0 & 0 \end{pmatrix} U^t, \quad P = U^{-t} \begin{pmatrix} \Lambda & 0 \\ 0 & 0 \end{pmatrix} U^{-1}.$$

The following lemma defines a factorization which plays an important role in the optimal projection approach for solving the reduced order model problem.

LEMMA 5. [10] *Let symmetric positive semidefinite $\hat{Q}$, $\hat{P} \in R^{n\times n}$ satisfy the rank conditions (9). Then, there exist G, $\Gamma \in R^{n_m \times n}$ and positive semisimple* (positive semisimple *means similar to a symmetric positive definite matrix) $M \in R^{n_m \times n_m}$ such that*

$$\hat{Q}\hat{P} = G^t M \Gamma, \tag{11}$$
$$\Gamma G^t = I_{n_m}. \tag{12}$$

Proof. Due to Remark 4 there exist nonsingular $W \in R^{n\times n}$ and positive definite diagonal $\Sigma \in R^{n_m \times n_m}$ such that

$$\hat{Q} = W \begin{pmatrix} \Sigma & 0 \\ 0 & 0 \end{pmatrix} W^t, \quad \hat{P} = W^{-t} \begin{pmatrix} \Sigma & 0 \\ 0 & 0 \end{pmatrix} W^{-1}. \tag{13}$$

The equations (13) can be expressed in the equivalent form

$$\hat{Q} = W_1 \Sigma W_1^t, \quad \hat{P} = U_1^t \Sigma U_1, \tag{14}$$

where

$$W = \overbrace{\begin{pmatrix} W_1 & W_2 \end{pmatrix}}^{n_m}, \quad W^{-1} = U = \begin{matrix} n_m \{ \\ \\ \end{matrix} \begin{pmatrix} U_1 \\ U_2 \end{pmatrix}. \tag{15}$$

From (14)–(15) with $G \equiv W_1^t$ and $\Gamma \equiv U_1$ follow (11) and (12). Q. E. D.

Matrices G, M and Γ from Lemma 5 are a (G, M, Γ)-*factorization* of $\hat{Q}\hat{P}$.

3. Homotopy Algorithms.

3.1. Probability-One Homotopy Theory.

Homotopies are a traditional part of topology, and have found significant application recently in control analysis and design [17], [29]. Homotopy methods are globally convergent, which distinguishes them from most iterative methods, which are only locally convergent. The general idea of homotopy methods is to make a continuous transformation from an initial problem, which can be solved trivially, to the target problem.

Following [24], the theoretical foundation of all probability-one globally convergent homotopy methods is given in the following differential geometry theorem:

DEFINITION 6. Let $U \subset R^m$ and $V \subset R^p$ be open sets, and let $\rho : U \times [0,1) \times V \to R^p$ be a C^2 map. ρ is said to be transversal to zero if the Jacobian matrix $D\rho$ has full rank on $\rho^{-1}(0)$.

THEOREM 7. *If $\rho(a, \lambda, x)$ is transversal to zero, then for almost all $a \in U$ the map*

$$\rho_a(\lambda, x) = \rho(a, \lambda, x)$$

is also transversal to zero; i.e, with probability one the Jacobian matrix $D\rho_a(\lambda, x)$ has full rank on $\rho_a^{-1}(0)$.

The recipe for constructing a globally convergent homotopy algorithm to solve the nonlinear system of equations

$$f(x) = 0,$$

where $f : R^p \to R^p$ is a C^2 map, is as follows: For an open set $U \subset R^m$ construct a C^2 homotopy map $\rho : U \times [0,1) \times R^p \to R^p$ such that

1) $\rho(a, \lambda, x)$ is transversal to zero,
2) $\rho_a(0, x) = \rho(a, 0, x) = 0$ is trivial to solve and has a unique solution x_0,
3) $\rho_a(1, x) = f(x)$,
4) $\rho_a^{-1}(0)$ is bounded.

Then for almost all $a \in U$ there exists a zero curve γ of ρ_a, along which the Jacobian matrix $D\rho_a$ has rank p, emanating from $(0, x_0)$ and reaching a zero $\bar{x}$ of f at $\lambda = 1$. This zero curve γ does not intersect itself, is disjoint from any other zeros of ρ_a, and has finite arc length in every compact subset of $[0, 1) \times R^p$. Furthermore, if $Df(\bar{x})$ is nonsingular, then γ has finite arc length. The general idea of the algorithm is to follow the zero curve γ emanating from $(0, x_0)$ until a zero $\bar{x}$ of $f(x)$ is reached (at $\lambda = 1$).

The zero curve γ is tracked by the normal flow algorithm [26], a predictor-corrector scheme. In the predictor phase, the next point is produced using Hermite cubic interpolation. Starting at the predicted point, the corrector iteration involves computing (implicitly) the Moore-Penrose pseudo-inverse of the Jacobian matrix at each point. The most complex part of the homotopy algorithm is the computation of the tangent vectors to γ, which involves the computation of the kernel of the $p \times (p+1)$ Jacobian matrix $D\rho_a$. The kernel is found by computing a QR factorization of $D\rho_a$, and then using back substitution. This strategy is implemented in the mathematical software package HOMPACK [27], which was used for the curve tracking here.

Two different homotopy maps are used for solving the optimal projection equations. When the initial problem, $g(x; a) = 0$, can be solved, then the homotopy map is [25]

$$\rho_a(\lambda, x) = F(a, \lambda, x) \equiv \lambda\, f(x) + (1 - \lambda) g(x; a), \tag{16}$$

where $f(x) = 0$ is the final problem, and a is a parameter vector used in defining the function g.

When the initial problem is not solved exactly, i.e., $g(x_0; b) \neq 0$, then the map is a Newton homotopy [21]

$$\rho_a(\lambda, x) = F(b, \lambda, x) - (1 - \lambda) F(b, 0, x_0), \tag{17}$$

where $a = (b, x_0)$. For $\lambda = 0$, $\rho_a(0, x_0) = F(b, 0, x_0) - F(b, 0, x_0) = 0$, and for $\lambda = 1$, $\rho_a(1, x) = F(b, 1, x) = f(x) = 0$.

For the homotopies considered here, the theoretical verification of properties 1) and 4) is highly technical and has not been done.

3.2. Homotopy Method Based on the Drazin Inverse.

One approach in designing a homotopy algorithm for solving the optimal projection equations is to use the decomposition of the projection matrix τ based on the Drazin inverse

$$\tau = (\hat{Q}\,\hat{P})\,(\hat{Q}\,\hat{P})^{\sharp},$$

where $(\hat{Q}\,\hat{P})^{\sharp}$ is the Drazin inverse of $\hat{Q}\,\hat{P}$.

$A^{\sharp}$ is computed using the Hermite echelon form as described in [4].

The derivative of the Drazin inverse is computed using the Sylvester equations. For a given A of index 1, let X be the unique matrix (Drazin inverse) that satisfies:

$$A\,X\,A = A, \tag{18}$$

$$X\,A\,X = X, \tag{19}$$

$$A\,X = X\,A. \tag{20}$$

Differentiating (18)–(20) yields:

$$A' X A + A X' A + A X A' = A', \tag{21}$$

$$X' A X + X A' X + X A X' = X', \tag{22}$$

$$A X' - X' A = X A' - A' X. \tag{23}$$

Substituting $A X'$ from (23) into (21), and summing up equations (21)–(23) gives the Sylvester equation

$$\begin{aligned}(A + X A - I)X' + X'(A X - A + A^2) \\ = X A' - A' X - X A' X + A' - X A' A - A X A',\end{aligned} \tag{24}$$

which has a unique solution for generic A.

Solving (24) for X' completes the computation of the derivative of the Drazin inverse of the matrix A.

From (14),

$$\begin{aligned}\text{rank } (\hat{Q}\,\hat{P})^2 &= \text{rank } (W_1\,\Sigma\,W_1^t\,U_1^t\Sigma\,U_1)^2 \\ &= \text{rank } (W_1\,\Sigma^4\,U_1) = \text{rank } (\Sigma^4) = \text{rank } (\Sigma^2) \\ &= \text{rank } (W_1\,\Sigma\,W_1^t\,U_1^t\,\Sigma\,U_1) = \text{rank } (\hat{Q}\,\hat{P}) = n_m < n,\end{aligned}$$

and therefore $\hat{Q}\hat{P}$ has index 1.

Since $\hat{Q}\,\hat{P} = W_1\,\Sigma^2\,U_1$, it follows $(\hat{Q}\,\hat{P})^\sharp = W_1\,\Sigma^{-2}\,U_1$. Using the definition of τ it implies

$$\tau = (\hat{Q}\,\hat{P})\,(\hat{Q}\,\hat{P})^\sharp. \tag{25}$$

From (25) it follows

$$\tau' = (\hat{Q}\,\hat{P})'(\hat{Q}\,\hat{P})^\sharp + (\hat{Q}\,\hat{P})[(\hat{Q}\,\hat{P})^\sharp]',$$

where everything is directly computable except $[(\hat{Q}\,\hat{P})^\sharp]'$. $[(\hat{Q}\,\hat{P})^\sharp]'$ can be computed using the procedure for computation of the derivative of the Drazin inverse described above.

The following is a description of the algorithm. The algorithm is based on the normal flow algorithm for dense Jacobian matrices described in [26], slightly modified here to handle the rank requirements of the solution $(\hat{Q}, \hat{P})$.

The algorithm is using homotopy map (16) or (17), where $F(a, \lambda, x)$ is represented by two equations:

$$A(\lambda)\,\hat{Q} + \hat{Q}\,A^t(\lambda) + \tau\,B\,V\,B^t + B\,V\,B^t\,\tau^t - \tau\,B\,V\,B^t\,\tau^t = 0,$$

$$A^t(\lambda)\,\hat{P} + \hat{P}\,A(\lambda) + \tau^t\,C^t\,R\,C + C^t\,R\,C\,\tau - \tau^t\,C^t\,R\,C\,\tau = 0,$$

where $A(\lambda) = \lambda\,A + (1 - \lambda)\,D$, and the choice of D is discussed in Section 3.6.

The algorithm starts at the point $(\lambda, x) = (0, x_0) = (0, \hat{Q}_0, \hat{P}_0)$ with some $x_0 = (\hat{Q}_0, \hat{P}_0)$. Then it follows the zero curve γ of the homotopy map until a point where $\lambda = 1$ is reached.

Since the equations are symmetric, only the upper right triangles are considered, i.e., q_{ij} and p_{ij} are computed only for $j \geq i$. Therefore, the number of variables is $2\,[n(n+1)/2] = n(n+1)$.

The mathematical software package HOMPACK requires that the user provide routines to evaluate $\rho_a(\lambda, x)$ and the Jacobian matrix $D\rho_a$ at each step. While the former is relatively simple, the latter involves considerable computational effort.

The Jacobian matrix consists of $n(n+1)+1$ derivative vectors, which correspond to the partial derivatives with respect to λ, q_{ij} and p_{ij}. The terms that do not include τ are simple to evaluate analytically. For example,

$$\left[\frac{\partial}{\partial q_{ij}}(A\,Q)\right]_{kl} = \frac{\partial}{\partial q_{ij}} \sum_{m=1}^{n} a_{km} q_{ml} = \delta_{jl}\, a_{ki}. \tag{26}$$

On the other hand, the components of τ' have to be evaluated numerically. Each evaluation involves solving a Sylvester equation (24). Fortunately, since for different q_{ij} and p_{ij} only the right hand side of the equation changes, the whole process can be done efficiently. In order to compute $\partial/\partial q_{11}$ all the computations related to (24) are done completely. For all subsequent partial derivative evaluations only the right hand side of (24) is evaluated and submitted to the procedure that solves a Sylvester equation. The procedure used for that purpose [2] supports this approach very efficiently.

When a final solution to the equations (6)–(8) is obtained, the computation of (A_m, B_m, C_m) is completed by applying the formulas (3)–(5), where G and Γ are obtained as explained in Lemma 5.

In summary, the whole algorithm is:

1) Define D using formula (38) or (39).
2) Choose a starting point x_0 using one of the strategies explained in Section 3.6.
3) Set $\lambda := 0$, $x := x_0$.
4) Compute Drazin inverse $(\hat{Q}\,\hat{P})^{\sharp}$.
5) Compute $\tau = (\hat{Q}\,\hat{P})\,(\hat{Q}\,\hat{P})^{\sharp}$.
6) Evaluate $\rho_a(\lambda, x)$.
7) (Evaluate $D\rho_a(\lambda, x)$.) For λ and each q_{ij}, p_{ij} such that $j \geq i$, do Steps 8–11.
   8) Compute derivatives of terms that do not include τ using analytical formulas similar to (26).
   9) Compute $[(\hat{Q}\,\hat{P})^{\sharp}]'$ using equation (24).
   10) Complete computation of τ' as $\tau' := (\hat{Q}\,\hat{P})'(\hat{Q}\,\hat{P})^{\sharp} + (\hat{Q}\,\hat{P})[(\hat{Q}\,\hat{P})^{\sharp}]'$.
   11) Sum values obtained in steps 8 and 10 to the final value of the derivative vector.
12) Take a step along the curve and obtain $x_1 = (\hat{Q}_1, \hat{P}_1)$.
13) Compute a contragredient transformation (13) as if x_1 satisfied the rank conditions.
14) Use formulas (14) to compute $\bar{x}_1 = (\hat{Q}, \hat{P})$.
15) If $\lambda < 1$, then set $x := \bar{x}_1$, and go to Step 4.
16) Interpolate to get the solution $\bar{x}_1$ at $\lambda = 1$. Obtain G and Γ as explained in Lemma 5.
17) Compute the reduced order model using (3)–(5).

3.3. First Method Based on Decomposition of Pseudogramians.

Homotopy algorithms for solving optimal projection equations can be designed using decompositions of the pseudogramians based on contragredient transformations.

The equations (6)–(7) can be considered in another, equivalent form. If (6) is multiplied by U_1 from the left, and (7) is multiplied by W_1 from the right, using the contragredient transformation

$$\hat{Q} = W_1 \Sigma W_1^t, \quad \hat{P} = U_1^t \Sigma U_1, \tag{27}$$

the following two equations are obtained:

$$U_1 A W_1 \Sigma W_1^t + \Sigma W_1^t A^t + U_1 B V B^t = 0, \tag{28}$$

$$A^t U_1^t \Sigma + U_1^t \Sigma U_1 A W_1 + C^t R C W_1 = 0. \tag{29}$$

The third equation

$$U_1 W_1 - I = 0 \tag{30}$$

determines the relationship between W_1 and U_1.

The matrix equations (28)–(30) contain $2\,n\,n_m + n_m^2$ scalar equations. On the other hand, the only natural unknowns in (28)–(30), W_1, U_1 and diagonal Σ, contain $2\,n\,n_m + n_m$ variables. Hence, some additional techniques are necessary in order to make an exact match between the number of equations and the number of unknowns.

One approach is to consider Σ to be symmetric and all elements of Σ as unknowns. This is appropriate, since the equations (28)–(30) with a full symmetric Σ can be transformed into equations of the same form with a diagonal Σ by computing

$$\Sigma = T \bar{\Sigma} T^t, \quad \bar{W}_1 = W_1 T, \quad \bar{U}_1 = T^t U_1,$$

where $\bar{\Sigma}$ is diagonal and T is orthogonal.

The following is a description of the algorithm for the method determined by the equations (28)–(30). The algorithm is based on the normal flow algorithm for dense Jacobian matrices described in [26]. Depending on the relative size of $F(a, 0, x_0)$ the algorithm may be modified. If $F(a, 0, x_0)$ is relatively large, computational experience shows that it is desirable (but not theoretically necessary) to enforce the symmetry of Σ along the homotopy path. This is done by observing that a symmetrized Σ corresponds to *some* homotopy map that *could* have been chosen initially. In effect, x_0 is changed in the homotopy map at each step along the homotopy zero curve γ. Obviously, in that case the homotopy map (17) must be used.

The algorithm is using the homotopy map (16) or (17), where $F(a, \lambda, x)$ is represented by three equations:

$$U_1 A(\lambda) W_1 \Sigma W_1^t + \Sigma W_1^t A^t(\lambda) + U_1 B V B^t = 0, \tag{31}$$

$$A^t(\lambda) U_1^t \Sigma + U_1^t \Sigma U_1 A(\lambda) W_1 + C^t R C W_1 = 0, \tag{32}$$

$$U_1 W_1 - I = 0. \tag{33}$$

In summary, the whole algorithm is:

1) Define D using formula (38) or (39).
2) Choose a starting point (Q_0, P_0) using one of strategies explained in Section 3.6. Compute $(W_1)_0$, $(U_1)_0$ and Σ_0 using a contragredient transformation.

3) Set $\lambda := 0$, $x := x_0 = ((W_1)_0, (U_1)_0, \Sigma_0)$.
4) Evaluate $\rho_a(\lambda, x)$ given by (16) or (17), and (31)–(33).
5) Evaluate $D\rho_a(\lambda, x)$.
6) Take a step along the curve and obtain $x_1 = (W_1, U_1, \Sigma)$, $\bar{\lambda}$.
7) Compute $\bar{x}_1 = (W_1, U_1, \bar{\Sigma}) = (W_1, U_1, (\Sigma + \Sigma^t)/2)$.
8) Change the homotopy to

$$F(a, \lambda, x) - (1 - \lambda)v = 0,$$

where $v = F(a, \bar{\lambda}, \bar{x}_1)/(1 - \bar{\lambda})$.
9) If $\bar{\lambda} < 1$, then set $x := \bar{x}_1$, $\lambda := \bar{\lambda}$, and go to Step 4.
10) If $\bar{\lambda} \geq 1$, compute the solution $\bar{x}_1$ at $\bar{\lambda} = 1$. Compute the reduced order model by diagonalizing $\Sigma = T\,\bar{\Sigma}\,T^t$.

Note: if $F(a, 0, x_0)$ is small, Steps 7 and 8 can be omitted without a serious loss of efficiency.

3.4. Second Method Based on Decomposition of Pseudogramians.

Another approach in transforming (6)–(7) is to consider the decomposition

$$\hat{Q} = W_1\,\Sigma\,W_1^t, \quad \hat{P} = U_1^t\,\Omega\,U_1, \tag{34}$$

which leads to the equations

$$U_1\,A\,W_1\Sigma\,W_1^t + \Sigma\,W_1^t\,A^t + U_1\,B\,V\,B^t = 0, \tag{35}$$

$$A^t\,U_1^t\,\Omega + U_1^t\,\Omega\,U_1\,A\,W_1 + C^t\,R\,C\,W_1 = 0, \tag{36}$$

$$U_1\,W_1 - I = 0, \tag{37}$$

which also have $2\,n\,n_m + n_m^2$ scalar equations. In this case the number of unknowns in W_1, U_1 and symmetric Σ and Ω is $2\,n\,n_m + n_m^2 + n_m$. An additional n_m equations can be obtained, for example, by requiring

$$\sigma_{ii} - \omega_{ii} = 0 \quad \text{for } i = 1, \ldots, n_m.$$

3.5. Third Method Based on Decomposition of Pseudogramians.

Another way to design a method using equations (35)–(37) is to reduce the number of unknowns. The number of unknowns can be reduced to $2\,n\,n_m + n_m^2$ if the diagonal elements of Ω are actually the diagonal elements of Σ.

3.6. Choosing an Initial System and the Starting Point.

While with homotopy algorithms in general an initial problem can be chosen practically at random, this problem has some special limitations. The reason is that Theorem 2 provides necessary conditions on a solution only under certain assumptions. In other words, every intermediate problem solution satisfies these equations only if the system is asymptotically stable, controllable and observable. While the absence of these features does not automatically mean that the intermediate problem solution will not satisfy the equations, it is clearly better to define a homotopy path in such a way that each problem along it corresponds to an asymptotically stable, controllable and observable system. Existence of a solution to the reduced order problem follows from [22]. Theorem 8, proved in [29], defines a class of initial systems such that these conditions are satisfied.

THEOREM 8. *For the given system* (1)–(2), *let* $A = X \Lambda X^{-1}$, *with* Λ *diagonal. Define* $D = X \Omega X^{-1}$ *for any diagonal matrix* $\Omega = diag\ (\omega_1, \ldots, \omega_n)$, *such that all* ω_i, *for* $i = 1, 2, \ldots, n$, *are in the open left half plane. Then for almost all such* D *any convex combination* $(A(\alpha), B, C)$ *of the systems* (D, B, C) *and* (A, B, C) *will be asymptotically stable, controllable and observable.*

While the random construction of the matrix D given in Theorem 8 is theoretically plausible, in practice it may not be wise. The reason is that the matrix X is complex in general, which for many choices of Ω leads to a complex matrix D, which is undesirable. Hence, it is better to directly construct a matrix D such that Ω satisfies the conditions given in Theorem 8.

One simple choice for D is

$$D \equiv -c_1 I + \text{diag}\ \{\epsilon_1, \ldots, \epsilon_n\}, \tag{38}$$

where $c_1 > 0$ and ϵ_i are small random numbers that correspond to the parameter a in the theory. In this case Ω is a small perturbation of $-c_1 I$.

Also, the matrix D can be defined as

$$D \equiv -c_1 I + c_2 A, \tag{39}$$

for $c_1, c_2 > 0$. In this case $\Omega = -c_1 I + c_2 \Lambda$.

The starting point $x_0 = (Q_0, P_0)$ of the homotopy algorithm can be chosen using a number of different strategies.

One strategy is to choose Q_0 and P_0 that are positive semidefinite and satisfy the rank conditions, but are otherwise random. This approach may lead to relatively large values of $F(a, 0, x_0) = g(x_0; a)$.

The second strategy, which can be applied for any choice of the matrix D described above, generally leads to relatively small but nonzero $g(x_0)$. Since the matrix D is asymptotically stable, the Lyapunov equation

$$D Q + Q D^t + B V B^t = 0 \tag{40}$$

has a unique solution Q. Let $Q = T \Sigma T^t$, where T is orthogonal and

$$\Sigma = \text{diag}\ \{\sigma_1, \ldots, \sigma_n\}.$$

Next, define

$$\Sigma_1 \equiv \text{diag}\ \{\sigma_1, \ldots, \sigma_{n_m}, 0, \ldots, 0\}, \qquad Q_0 \equiv T \Sigma_1 T^t.$$

If Q_0 is substituted for Q in (40), the equation will not be satisfied, but in general, if σ_i are sufficiently small, it will not be very different from zero. A similar procedure can be applied to compute P_0 that will 'almost' satisfy the equation

$$D^t P + P D + C^t R C = 0.$$

The point $x_0 = (Q_0, P_0)$ chosen in this way may lead to small values of $g(x_0)$. Also, this x_0 can be used as the initial guess for a quasi-Newton algorithm which may find a solution to the initial problem

$$\tau[D Q + Q D^t + B V B^t] = 0, \tag{41}$$

$$[D^t P + P D + C^t R C]\tau = 0. \tag{42}$$

The third strategy can be utilized for $n_m \geq \max\{m, l\}$ and a choice of D as in (38) or (39) with small c_2. If $D = -c\,I$ it follows from (41)–(42) that

$$Q = \frac{B\,V\,B^t}{2\,c}, \quad P = \frac{C^t\,R\,C}{2\,c}$$

are solutions with rank $Q = m$ and rank $P = l$. Let $B_1 \in R^{n \times n_m}$ be a full column rank matrix obtained by augmenting (if necessary) B, $C_1 \in R^{n_m \times n}$ a full row rank matrix obtained by augmenting (if necessary) C, and $V_1, R_1 \in R^{n_m \times n_m}$ full rank matrices obtained by augmenting (if necessary), respectively, V and R. If the numbers used for augmentation are relatively small, then

$$Q_0 \equiv \frac{B_1\,V_1\,B_1^t}{2\,c}, \quad P_0 \equiv \frac{C_1^t\,R_1\,C_1}{2\,c}$$

will be close to solutions of (41)–(42). Furthermore, since $x_0 = (Q_0, P_0)$ satisfies the first two rank conditions (8), and with appropriate augmentation of B_1 and C_1 the third rank condition can be satisfied, this point can be used as the initial guess for a quasi-Newton algorithm that can solve (41)–(42), (8), giving an initial point for the homotopy algorithm. The quasi-Newton algorithm is using equivalent forms of these equations (28)–(30), where the rank conditions do not appear explicitly, but are incorporated into the equations.

The fourth strategy can be applied for the same n_m and D as for the third strategy. Let

$$Q = \frac{B\,V\,B^t}{2\,c}, \quad P = \frac{C^t\,R\,C}{2\,c}$$

and $p = \min\{m, l\}$. If $m = l$ set $\tilde{Q}_0 \equiv Q$, $\tilde{P}_0 \equiv P$ and proceed, otherwise a small correction is necessary. Suppose $m > l$. Let $Q = T\,\Sigma\,T^t$, where T is orthogonal and

$$\Sigma = \text{diag}\,\{\sigma_1, \ldots, \sigma_m, 0, \ldots, 0\}.$$

Define

$$\Sigma_1 \equiv \text{diag}\,\{\sigma_1, \ldots, \sigma_l, 0, \ldots, 0\},$$

$\tilde{Q}_0 \equiv T\,\Sigma_1\,T^t$, $\tilde{P}_0 \equiv P$. $\tilde{Q}_0$ and $\tilde{P}_0$ thus obtained will satisfy the first two rank conditions (8) for $n_m = p = l$, but may not satisfy the third.

The next step is to apply a contragredient transformation, if necessary in a modified form, to obtain decompositions

$$\tilde{Q}_0 = W \begin{pmatrix} \Sigma & 0 \\ 0 & 0 \end{pmatrix} W^t, \quad \tilde{P}_0 = U^t \begin{pmatrix} \Sigma & 0 \\ 0 & 0 \end{pmatrix} U,$$

with $W, U \in R^{n \times n}$, $\Sigma \in R^{p \times p}$, where $U\,W = I$ and diagonal Σ is positive definite. Details of this computation are given in [29].

Let

$$\Sigma = \text{diag}\,\{\sigma_1, \ldots, \sigma_p, 0, \ldots, 0\}.$$

Define

$$\Sigma_1 \equiv \text{diag}\,\{\sigma_1, \ldots, \sigma_p, \sigma_{p+1}, \ldots, \sigma_{n_m}, 0, \ldots, 0\},$$

where $\sigma_{p+1}, \ldots, \sigma_{n_m}$ are small positive numbers. Let $Q_0 \equiv W \Sigma_1 W^t$, $P_0 \equiv U^t \Sigma_1 U$. Then, $x_0 = (Q_0, P_0)$ will lead to small values of $g(x_0)$. Also, x_0 can be used as the initial guess for a quasi-Newton algorithm for solving (41)–(42), (8).

The fifth strategy does not appear theoretically elegant, but it has given some very good results. The following is the motivation for this strategy. If the starting point x_0 of the homotopy and the final point $\bar{x}$ are very far apart in norm, it is obvious that the homotopy curve for the problem will be very long, hence it will require a large number of steps to track. If x_0 and $\bar{x}$ are close there is no guarantee that the curve will be short, but intuitivelly it seems that should lead to a curve of smaller length. Hence, the goal is to 'guess' an initial point that will be close to the final point.

Points along the homotopy curve are generally (but not always) closer to the final point as λ grows. Hence, if a point along the curve is taken as x_0 for a new program run starting back at $\lambda = 0$, that starting point should be closer to $\bar{x}$. If that point is chosen for a value of λ closer to 1, it can be expected that x_0 will be closer to $\bar{x}$.

4. Numerical Results.

This section contains results and observations obtained on a number of examples. Most of the examples are taken from journal papers. Subsection 4.1 contains models of different orders for 10 systems. Subsection 4.2 contains some program measures obtained from runs on a DECstation 3100. Subsection 4.3 compares results obtained using four proposed methods. Subsection 4.4 compares results obtained using different strategies for choosing the starting point. Subsection 4.5 gives an example of an efficient use of Strategy 5 for choosing the starting point. Subsection 4.6 compares models obtained using the optimal projection approach and models obtained using other approaches. Subsection 4.7 illustrates the importance of the choice of the initial system. Subsection 4.8 contains a summary of the section and some recommendations for future work.

The following ten examples of systems (A, B, C) were considered.

SYSTEM 1 [11]. The system is given by

$$A = \begin{pmatrix} -0.05 & -0.99 \\ -0.99 & -5000.0 \end{pmatrix}, \quad B = \begin{pmatrix} 1 \\ 100 \end{pmatrix}, \quad C = \begin{pmatrix} 1 & 100 \end{pmatrix}.$$

SYSTEM 2 [20]. The system is given by

$$A = \begin{pmatrix} -1 & 0 \\ 0 & -10 \end{pmatrix}, \quad B = \begin{pmatrix} 1 & 1 \\ 70 & 1 \end{pmatrix}, \quad C = \begin{pmatrix} 1 & -0.2 \end{pmatrix}.$$

SYSTEM 3 [11]. The system is given by

$$A = \begin{pmatrix} -0.25 & -0.4 \\ -0.4 & -0.72 \end{pmatrix}, \quad B = \begin{pmatrix} 1 \\ 1.2 \end{pmatrix}, \quad C = \begin{pmatrix} 1 & 1.2 \end{pmatrix}.$$

SYSTEM 4 [23]. The system is given by

$$A = \begin{pmatrix} -1 & 3 & 0 \\ -1 & -1 & 1 \\ 4 & -5 & -4 \end{pmatrix}, \quad B = \begin{pmatrix} -2 \\ 2 \\ 4 \end{pmatrix}, \quad C = \begin{pmatrix} -2 & 2 & 4 \end{pmatrix}.$$

SYSTEM 5 [20]. The system is given by

$$A = \begin{pmatrix} -10 & 1 & 0 \\ -5 & 0 & 1 \\ -1 & 0 & 0 \end{pmatrix}, \quad B = \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix}, \quad C = (1 \quad 0 \quad 0).$$

SYSTEM 6. The system for this example is given by

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 \\ -2 & -0.02 & 1 & 0.01 \\ 0 & 0 & 0 & 1 \\ 0.1 & 0.001 & -0.1 & -0.001 \end{pmatrix}, \quad B = \begin{pmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix}, \quad C = (0 \quad 1 \quad 0 \quad 0).$$

SYSTEM 7 [28]. The system is given by

$$A = \begin{pmatrix} 0 & 0 & 0 & -150 \\ 1 & 0 & 0 & -245 \\ 0 & 1 & 0 & -1113 \\ 0 & 0 & 1 & -19 \end{pmatrix}, \quad B = \begin{pmatrix} 4 \\ 1 \\ 0 \\ 0 \end{pmatrix}, \quad C = (0 \quad 0 \quad 0 \quad 1).$$

SYSTEM 8 [15]. The system is given by

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -50 & -79 & -33 & -5 \end{pmatrix}, \quad B = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}, \quad C = (50 \quad 15 \quad 1 \quad 0).$$

SYSTEM 9 [8]. The system is given by

$$A = \begin{pmatrix} -6.2036 & 15.054 & -9.8726 & -376.58 & 251.32 & -162.24 & 66.827 \\ 0.53 & -2.0176 & 1.4363 & 0 & 0 & 0 & 0 \\ 16.846 & 25.079 & -43.555 & 0 & 0 & 0 & 0 \\ 377.4 & -89.449 & -162.83 & 57.998 & -65.514 & 68.579 & 157.57 \\ 0 & 0 & 0 & 107.25 & -118.05 & 0 & 0 \\ 0.36992 & -0.14445 & -0.26303 & -0.64719 & 0.49947 & -0.21133 & 0 \\ 0 & 0 & 0 & 0 & 0 & 376.99 & 0 \end{pmatrix},$$

$$B = \begin{pmatrix} 89.353 & 0 \\ 376.99 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0.21133 \\ 0 & 0 \end{pmatrix}, \quad C = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

SYSTEM 10. This is a state space model of the transfer function between a torque activator and an approximately collocated torsional rate sensor for the ACES structure [5], located at NASA Marshall Space Flight Center, Huntsville, AL. The system in this example is of size $n = 17$, $m = 1$, $l = 1$. The nonzero elements of A are

$A(1,1) = A(2,2) = -0.031978272,$ $\qquad A(1,2) = -A(2,1) = -78.54,$

$A(1,17) = 0.0097138566,$
$A(2,17) = -0.0060463517,$
$A(3,3) = A(4,4) = -5.152212,$
$A(3,4) = -A(4,3) = -51.457677,$
$A(3,17) = -0.021760771,$
$A(4,17) = -0.0054538246.$
$A(5,5) = A(6,6) = -0.1351159,$
$A(5,6) = -A(6,5) = -15.417859,$
$A(5,17) = -0.02179972,$
$A(6,17) = -0.015063913,$
$A(7,7) = A(8,8) = -0.42811443,$
$A(7,8) = -A(8,7) = -14.698408,$
$A(7,17) = -0.01042631,$
$A(8,17) = -0.0088479697,$
$A(9,9) = A(10,10) = -0.064896745,$
$A(9,10) = -A(10,9) = -12.077045,$
$A(9,17) = -0.030531575,$
$A(10,17) = -0.030260987,$
$A(11,11) = A(12,12) = -0.048520356,$
$A(11,12) = -A(12,11) = -8.9654448,$
$A(11,17) = -0.016843335,$
$A(12,17) = -0.011449591,$
$A(13,13) = A(14,14) = -0.036781718,$
$A(13,14) = -A(14,13) = -4.9057426,$
$A(13,17) = -0.1248007,$
$A(14,17) = -0.0005136047,$
$A(15,15) = A(16,16) = -0.025112482,$
$A(15,16) = -A(16,15) = -3.8432892,$
$A(15,17) = -0.035415526,$
$A(16,17) = -0.028115589,$
$A(17,17) = -92.399784.$

The matrices B and C are

$$B = \begin{pmatrix} 1.8631111 \\ -1.1413786 \\ -1.2105758 \\ 0.31424169 \\ 0.013307797 \\ -0.211128913 \\ 0.19552894 \\ -0.037391511 \\ -0.01049736 \\ -0.011486242 \\ -0.029376402 \\ 0.0082391613 \\ -0.012609562 \\ -0.0022040505 \\ -0.030853234 \\ 0.0011671662 \\ 0 \end{pmatrix}, \quad C^t = \begin{pmatrix} -0.0097138566 \\ 0.0060463517 \\ 0.021760771 \\ -0.0054538246 \\ 0.02179972 \\ 0.015063913 \\ -0.01042631 \\ -0.0088479697 \\ 0.030531575 \\ 0.030260987 \\ 0.016843335 \\ 0.011449591 \\ 0.1248007 \\ -0.0005136047 \\ 0.035415526 \\ 0.028115589 \\ 184.79957 \end{pmatrix}.$$

4.1. Solved Examples.

This section gives models for the systems. The results are obtained using the method of Section 3.3, using the homotopy (17). For all examples $V = R = I$.

System 1: With $D = -10000\, I$, for the starting point

$$x_0 = (W_1, U_1, \Sigma)_0 = \begin{pmatrix} 0.0099995 \\ 0.99995 \\ 0.0099995 \\ 0.99995 \\ 0.5 \end{pmatrix}$$

the homotopy algorithm converges to a solution corresponding to the model of order $n_m = 1$ given by

$$A_m = (-4998.078625), \quad B_m = (100.000194), \quad C_m = (100.000194).$$

This model yields the (minimum) cost $J = 96.078058$.

With $D = -100\,I$, for the starting point

$$x_0 = \begin{pmatrix} 1 \\ 0.25 \\ 1.07 \\ -0.27 \\ 1 \end{pmatrix}$$

the solution found corresponds to the model of order $n_m = 1$ given by

$$A_m = (-0.485152), \quad B_m = (-0.0000011427), \quad C_m = (-0.000000073400),$$

which yields the (maximum) cost $J = 10100.00$. This example shows that the homotopy method can obtain different solutions.

Due to the results from [10], the optimal projection equations can have at most $\binom{n}{n_m}$ solutions. Hence, in this case all the solutions were obtained. Obtaining all the solutions to the equations is the only known method that guarantees that one of the solutions will correspond to the global minimum.

System 2: A model of order $n_m = 1$ is

$$A_m = (-11.979443), \quad B_m = (-3.649276 \quad 0.442839), \quad C_m = (3.676048).$$

This model yields the cost $J = 0.598377$.

System 3: A model of order $n_m = 1$ is

$$A_m = (-0.838521), \quad B_m = (1.537575), \quad C_m = (1.537575).$$

This model yields the cost $J = 0.107256$.

System 4: A model of order $n_m = 1$ is

$$A_m = (-10.436530), \quad B_m = (-1.597203), \quad C_m = (1.597203).$$

This model yields the cost $J = 1.688216$.

A model of order $n_m = 2$ is

$$A_m = \begin{pmatrix} -3.600739 & 1.376721 \\ 1.376721 & -0.215037 \end{pmatrix}, \quad B_m = \begin{pmatrix} -1.711012 \\ 0.763496 \end{pmatrix}, \quad C_m^t = \begin{pmatrix} 1.711012 \\ 0.763496 \end{pmatrix}.$$

This model yields the cost $J = 0.0197781$.

System 5: A model of order $n_m = 1$ is

$$A_m = (-0.157898), \quad B_m = (0.423088), \quad C_m = (0.423088).$$

This model yields the cost $J = 0.0107792$.

A model of order $n_m = 2$ is

$$A_m = \begin{pmatrix} -0.448192 & -0.245890 \\ 0.245890 & -0.139652 \end{pmatrix}, \quad B_m = \begin{pmatrix} 0.301639 \\ 0.411520 \end{pmatrix}, \quad C_m^t = \begin{pmatrix} -0.301639 \\ 0.411520 \end{pmatrix}.$$

This model yields the cost $J = 0.000329024$.

System 6: A model of order $n_m = 2$ is

$$A_m = \begin{pmatrix} -0.0205177 & -1.431966 \\ 1.432245 & -0.00000309981 \end{pmatrix},$$

$$B_m = \begin{pmatrix} -0.931218 & 0.477408 \\ 0.00704476 & 0.0107630 \end{pmatrix}, \quad C_m^t = \begin{pmatrix} -1.046463 \\ 0.0128636 \end{pmatrix}.$$

This model yields the cost $J = 256.432$.

A model of order $n_m = 3$ is

$$A_m = \begin{pmatrix} -0.384322 & 0.0151428 & 0.000363187 \\ -0.0344809 & -0.0205145 & -1.431952 \\ 0.000412053 & 1.432223 & -0.00000322418 \end{pmatrix},$$

$$B_m = \begin{pmatrix} -0.0450631 & -0.864146 \\ -0.931197 & 0.477361 \\ 0.00659947 & 0.0113390 \end{pmatrix}, \quad C_m = \begin{pmatrix} -0.865320 & -1.046423 & 0.0131197 \end{pmatrix}.$$

This model yields the cost $J = 255.703$.

System 7: A model of order $n_m = 2$ is

$$A_m = \begin{pmatrix} -3.172419 & -1.170749 \\ 1.170749 & -0.437964 \end{pmatrix}, \quad B_m = \begin{pmatrix} -0.131078 \\ -0.118146 \end{pmatrix}, \quad C_m^t = \begin{pmatrix} 0.131078 \\ -0.118146 \end{pmatrix}.$$

This model yields the cost $J = 4.15847 \cdot 10^{-7}$.

A model of order $n_m = 3$ is

$$A_m = \begin{pmatrix} -12.554152 & -2.839372 & -0.414932 \\ 2.839372 & -3.135361 & -1.168474 \\ -0.414932 & 1.168474 & -0.437810 \end{pmatrix},$$

$$B_m = \begin{pmatrix} 0.0564264 \\ 0.130701 \\ 0.118135 \end{pmatrix}, \quad C_m = \begin{pmatrix} 0.0564264 & -0.130701 & 0.118135 \end{pmatrix}.$$

This model yields the cost $J = 4.58560 \cdot 10^{-10}$.

System 8: A model of order $n_m = 2$ is

$$A_m = \begin{pmatrix} -4.815122 & 1.508547 \\ -1.508547 & -0.532330 \end{pmatrix}, \quad B_m = \begin{pmatrix} 0.930361 \\ -0.779384 \end{pmatrix}, \quad C_m^t = \begin{pmatrix} -0.930361 \\ -0.779384 \end{pmatrix}.$$

This model yields the cost $J = 0.673079$.

A model of order $n_m = 3$ is

$$A_m = \begin{pmatrix} -0.746729 & 4.843984 & 0.419342 \\ -4.843984 & -2.235622 & -1.454066 \\ 0.419342 & -1.454066 & -0.520312 \end{pmatrix},$$

$$B_m = \begin{pmatrix} -0.358287 \\ 0.806973 \\ 0.773469 \end{pmatrix}, \quad C_m = (-0.358287 \quad -0.806973 \quad 0.773469).$$

This model yields the cost $J = 0.00148438$.

System 9: A model of order $n_m = 3$ is

$$A_m = \begin{pmatrix} -0.0261157 & 9.349756 & -0.0528086 \\ -9.352068 & -0.912444 & 0.506220 \\ -0.0541716 & -0.506226 & -0.198770 \end{pmatrix},$$

$$B_m = \begin{pmatrix} -2.414471 & -0.571953 \\ 14.906052 & 0.0416151 \\ -14.944459 & 0.0512237 \end{pmatrix}, \quad C_m = \begin{pmatrix} -0.371712 & 0.00240265 & 0.0122915 \\ 2.453290 & 14.906110 & 14.944542 \end{pmatrix}.$$

This model yields the cost $J = 0.673079$.

A model of order $n_m = 4$ is

$$A_m = \begin{pmatrix} -37.55440 & -0.0546940 & 0.326197 & -0.0709427 \\ 0.0561170 & -0.0261155 & 9.349755 & -0.0528084 \\ 0.324384 & -9.352067 & -0.912440 & 0.506220 \\ 0.0705453 & -0.0541714 & -0.506226 & -0.198769 \end{pmatrix},$$

$$B_m = \begin{pmatrix} -2.666516 & -0.00624702 \\ -2.414464 & -0.571953 \\ 14.906036 & 0.0416151 \\ -14.944458 & 0.0512237 \end{pmatrix}, \quad C_m^t = \begin{pmatrix} 0.280123 & -2.651769 \\ -0.371711 & 2.453283 \\ 0.00240433 & 14.906094 \\ 0.0122916 & 14.944541 \end{pmatrix}.$$

This model yields the cost $J = 3.27495 \cdot 10^{-6}$.

System 10: A model of order $n_m = 6$ is

$$A_m = \begin{pmatrix} -0.23442 & -52.59052 & -0.17250 & 0.06085 & 0.000005 & 0.019936 \\ 52.59052 & -10.53247 & -0.91009 & 0.54970 & 0.00004 & 0.13584 \\ 0.17250 & -0.91009 & -0.18410 & 15.43994 & 0.00001 & 0.046293 \\ 0.06085 & -0.54970 & -15.43994 & -0.03904 & -0.000006 & -0.02371 \\ 0.000005 & -0.00004 & -0.00001 & -0.000006 & -0.000000005 & -78.54024 \\ -0.019936 & 0.13584 & 0.046293 & 0.02371 & 78.54024 & -0.06402 \end{pmatrix},$$

$$B_m = \begin{pmatrix} 0.02447 \\ 0.16899 \\ 0.06006 \\ -0.027808 \\ -0.00004 \\ -0.15819 \end{pmatrix}, \quad C_m^t = \begin{pmatrix} 0.02447 \\ -0.16899 \\ -0.06006 \\ -0.027808 \\ -0.00004 \\ 0.15819 \end{pmatrix}.$$

This model yields the cost $J = 4.19165 \cdot 10^{-5}$.

A model of order $n_m = 8$ is given by

$$A_m = \begin{pmatrix} -70.147 & 21.918 & -2.7406 & 2.9917 & -0.3721 & 0.228 & 0.0246 & 0.083 \\ 54.161 & -32.186 & 4.6829 & 9.2995 & -0.4958 & 0.180 & 0.0289 & 0.093 \\ 3.5118 & -4.6512 & -0.2083 & -51.396 & 0.1211 & -0.013 & -0.0049 & -0.0157 \\ -22.253 & 19.045 & 51.852 & -12.043 & 1.0945 & -0.639 & -0.0741 & -0.243 \\ 1.2271 & -1.1976 & -0.2000 & 1.1602 & -0.1936 & 15.44 & 0.0243 & 0.0807 \\ 0.5249 & -0.5415 & -0.0764 & 0.6934 & -15.450 & -0.014 & -0.0125 & 0.041 \\ -0.0705 & 0.0708 & 0.0106 & -0.0770 & 0.0238 & 0.012 & 0.0181 & -78.574 \\ -0.2393 & 0.2397 & 0.0357 & -0.2610 & 0.0803 & 0.042 & 78.508 & -0.082 \end{pmatrix},$$

$$B_m = \begin{pmatrix} -0.05753 \\ -0.06445 \\ 0.01043 \\ 0.16983 \\ -0.05959 \\ 0.02622 \\ 0.04591 \\ 0.15167 \end{pmatrix}, \quad C_m^t = \begin{pmatrix} -0.16432 \\ 0.16512 \\ 0.02442 \\ -0.18165 \\ 0.05966 \\ 0.02629 \\ -0.04472 \\ -0.15162 \end{pmatrix}.$$

This model yields the cost $J = 3.95223 \cdot 10^{-5}$.

4.2. Program Measures.

This section contains some data about memory requirements and run times. All runs were performed on a DECstation 3100.

The most complex part of the algorithms is the computation of the tangent vectors to the curve γ, which involves the computation of the kernel of the Jacobian matrix $D\rho_a$. For large problems approximately 90% of the processor time required by the programs is devoted to that. For smaller problems, especially for smaller models, a lot of processor time is devoted to function evaluations and Jacobian computations that contain many matrix-matrix multiplications. Similarly, the Jacobian matrix takes most of the memory storage the programs require. Again, that is over 90% of all memory requirements for large problems.

Table 1 gives the size of the Jacobian matrices for different problem sizes. The table also contains a typical number of algorithm steps and Jacobian evaluations for each problem size. For different choices of parameters that determine an initial problem and the starting point those two numbers can vary significantly. The given data are for some good choices of those parameters. The last column contains the processor time needed for one tangent computation.

The largest problem listed in Table 1 required a total of approximately 77 CPU hours to solve, of which approximately 70 hours were devoted to tangent vector computations.

TABLE 1. PROGRAM MEASURES.

n	n_m	Jacobian	Steps	Evaluations	Time (sec)
2	1	5	10	20	0.011
3	1	7	12	30	0.017
3	2	16	15	45	0.045
4	1	9	45	75	0.025
4	2	20	60	120	0.08
4	3	33	60	120	0.21
7	1	15	600	700	0.09
7	2	32	600	700	0.29
7	3	51	700	1000	0.71
7	4	72	800	1100	1.5
17	1	35	400	450	1.3
17	2	72	1500	1700	3.6
17	3	111	1000	1100	7.6
17	8	336	1700	2900	96

4.3. Comparison of Different Homotopy Methods.

In this section the performance of the four methods described in Sections 3.2–3.5 is compared. The four methods are numbered according to the following scheme: Method 1 is the method of Section 3.2, Method 2 is the method of Section 3.3, Method 3 is the method of Section 3.4, and Method 4 is the method of Section 3.5. The methods were tested using the homotopy (17), choices of initial system $D = -10\,I$ or $D = -100\,I$, and using Strategy 2 for choosing the starting point.

Table 2 contains a comparison of the number of tangent computations required to solve different problems using these methods. The data are given for some some good choices of initial problems and starting points. Entries marked '–' mean that a solution has not been obtained within a reasonable number of steps or that the algorithm failed. The table represents the typical situation: Method 1 fails to give a result, Method 2 gives a result, while Methods 3 and 4 usually give results, but take more time than Method 2. This behavior is also seen for the other problems considered.

A possible reason for the better performance of Method 2 may be that it is the most 'natural' of the methods. All the methods are designed using some assumptions or constructions that are not inherent in the problem. It seems that these unnatural assumptions are contained to the least extent in Method 2.

TABLE 2. NUMBER OF TANGENT COMPUTATIONS.

Example	n_m	Method 1	Method 2	Method 3	Method 4
1	1	32	18	14	18
5	1	–	28	48	44
5	2	–	38	53	52
6	2	–	76	102	95
6	3	–	122	401	–
8	2	–	70	198	202
8	3	–	87	344	344

4.4. Comparison of Different Choices of Starting Point.

The choice of the starting point has a significant influence on the efficiency of the algorithm. Results for the four strategies explained in Section 3.6 are discussed here. All choices were tested in the context of Method 2, which has the best performance. Table 3 contains the number of tangent computations required to solve different problems using these strategies. According to Table 3, no strategy is clearly superior.

It is important to notice that, with the exception of the first strategy, the choice of the starting point depends on some program parameters which are used as 'small' numbers to augment matrices or make small perturbations. Hence, these results do not necessarily give the final judgement of these strategies, since there may be some choices of parameters that were not tested and which could improve the performance of the algorithm.

Since it is significantly different from the other strategies, Strategy 5 is discussed in a separate section.

TABLE 3. COMPARISON OF THE STRATEGIES FOR CHOOSING THE STARTING POINT.

Example	n_m	Str. 1	Str. 2	Str. 3	Str. 4
1	1	43	14	18	18
5	1	58	44	28	44
5	2	56	51	38	113
6	2	76	138	175	279
6	3	122	127	131	–
8	2	109	92	70	80
8	3	116	89	87	78

4.5. Choosing the Starting Point using Strategy 5.

If the curve tracking is performed using Strategy 5 for choosing the starting point, the result may be obtained in a smaller number of steps. This is illustrated on the problem of finding a model of order $n_m = 2$ for the system given in Example 9.

If the starting point is chosen by Strategy 2 and the initial system is $D = -100\,I$, then a solution is obtained in 833 steps, with 1141 tangent computations and a path arc length of 762. The trace of $x_7 = (W_1)_{4,1}$ is given in Figure 1.

If the solution is obtained by following the same path for the first 200 steps, then starting again from $\lambda = 0$ for another 200 steps, and the third time letting the algorithm finish, then the solution is obtained in a total of 735 steps, with 810 tangent computations and a path arc length of 737. The three traces of x_7 are given in Figure 2; note that the (G, M, Γ)-factorizations corresponding to Figures 1 and 2 are different (but the reduced order models are the same).

Typically Strategy 5 is more efficient than the others, due to smoother curves and convergence to different solutions, but it can be worse. Although ad hoc, Strategy 5 is clearly worth trying.

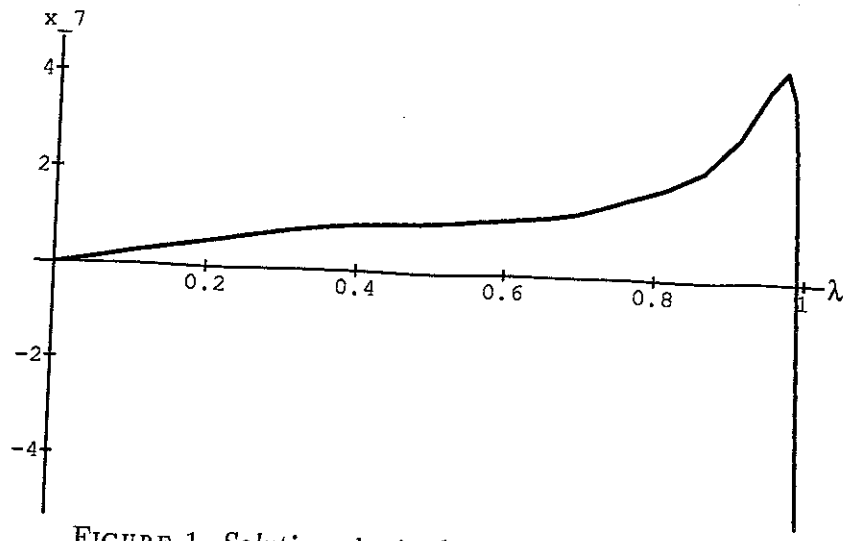

FIGURE 1. *Solution obtained using Strategy 2.*

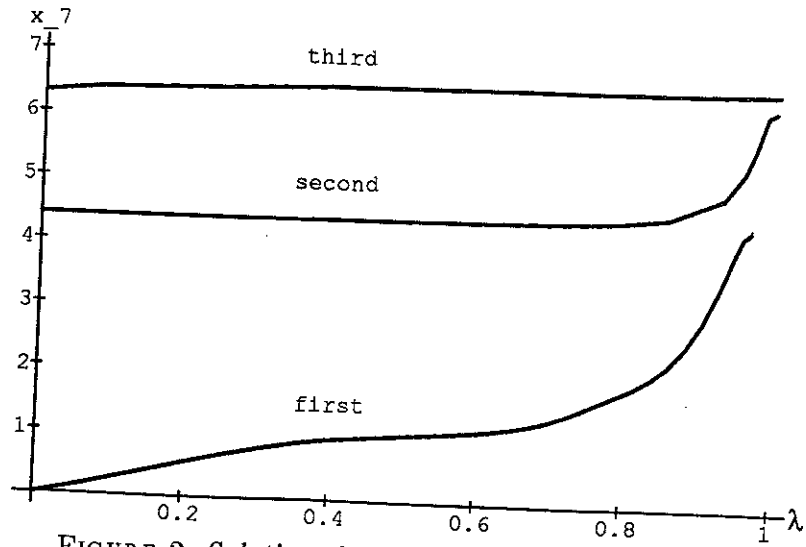

FIGURE 2. *Solution obtained using Strategy 5.*

4.6. Comparison of Optimal Projection Results and Other Methods.

According to Section 2, the optimal projection equations lead to a model which minimizes the cost function. Since the equations may have multiple solutions, there is no guarantee that a solution found will correspond to the minimal model. The only certain way to achieve that is to find all the solutions.

Numerical experiments have shown that using homotopy methods it is relatively simple to obtain multiple solutions to the equations by changing the initial system or the starting point. In all cases where the referenced papers give the models, those models yield higher costs than the best models obtained using optimal projection methods. Table 4 gives a comparison of the

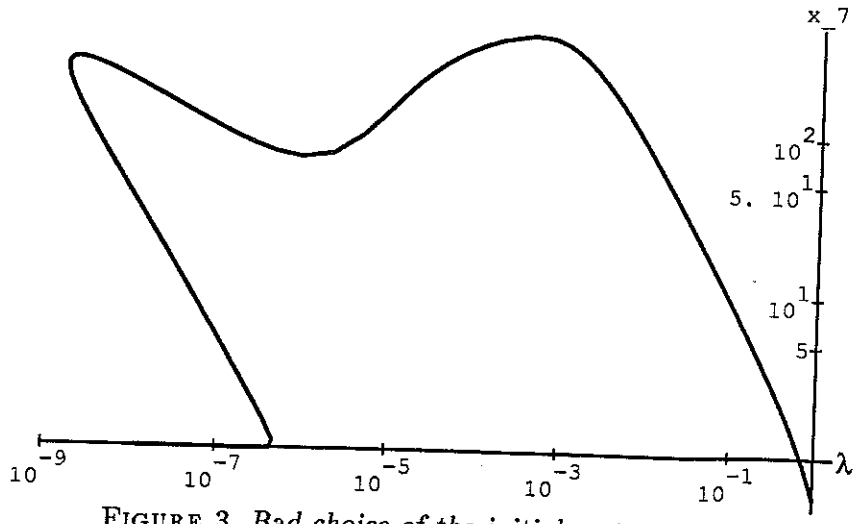

FIGURE 3. *Bad choice of the initial system.*

costs, computed as tr $(\tilde{Q}\,\tilde{R})$, between the optimal projection models and the models given in the referenced papers.

The following methods are compared to the optimal projection method: component cost analysis (Examples 2 and 5), projection formulation (Example 4), balancing (Example 7) and nonminimal partial realization (Example 9).

TABLE 4. COST COMPARISON BETWEEN OPTIMAL PROJECTION AND OTHER METHODS.

Example	n_m	Optimal Projection	Other
2	1	0.598377	0.697174
4	2	0.0197781	0.0469629
5	2	0.000329024	0.0340898
7	2	$4.15847 \cdot 10^{-7}$	$4.17767 \cdot 10^{-7}$
7	3	$4.58560 \cdot 10^{-10}$	$5.39446 \cdot 10^{-10}$
9	3	0.673079	1.003622

4.7. Choice of the initial system.

Although the methods work successfully with most choices (38) and (39) of the initial systems, this choice has a significant impact on the performance of the algorithm. Heuristically, it seems that the best choice is $D \equiv -c_1 I$, with c_1 of the same order of magnitude as the spectral radius of the matrix A.

The following example shows that the performance of the algorithm is strongly affected by the choice of the initial system. The system from Example 5 is considered and the model of order $n_m = 1$ is sought. The data are obtained using Method 2 and Strategy 1 for choosing the starting point.

With the initial system $D = -0.0003\,I$, 3100 steps are necessary to solve the problem. Figure 3 shows the behavior of the variable $x_7 = \Sigma_{11}$.

With the initial system $D = -10\,I$, 21 steps are sufficient. Figure 4 shows Σ_{11} for this case.

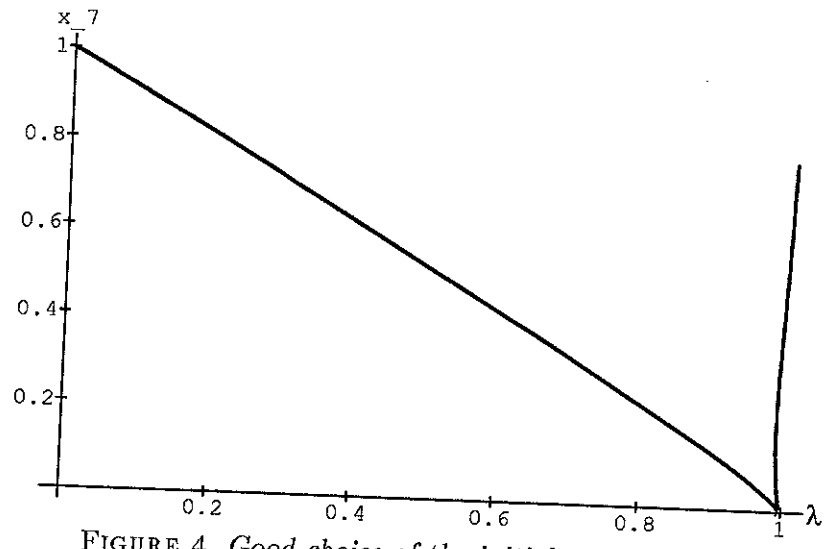

FIGURE 4. *Good choice of the initial system.*

A possible reason for the poor performance in the first case may be that the initial system is close to being asymptotically unstable.

4.8. Summary.

The proposed methods are effective ways for solving the optimal projection equations and obtaining appropriate models. Solved models were given in Section 4.1. Some program measures were given in Section 4.2. A comparison of the four methods defined in Sections 3.2–3.5 was given in Section 4.3. Comparisons of different strategies for choosing the starting point were given in Sections 4.4 and 4.5. Results obtained using these methods and other methods from the referenced papers were compared in Section 4.6. The example in Section 4.7 showed that the algorithm performance can be significantly different with a different choice of the initial system.

The results have shown that the best method is Method 2. A reason may be that in this method the smallest number of variables are forced to assume values that are not inherent (physically meaningful) to the problem, and the way it is done (full Σ, symmetrizing) is more natural than in the other methods. There is a large variety of submethods within four major methods. An example was given in Section 4.4, where it was shown that the same method with different strategies for determining the starting point gave significantly different results.

From this work it is not clear what the optimal strategy is for choosing the parameters c_1 and c_2 of (38) and (39). For different systems, different initial systems lead to efficient solutions. A general rule may be that the eigenvalues of the initial system should be of the same order of magnitude as the eigenvalues of the final system, but this does not always lead to the most efficient results. Further research should clarify this issue.

Another area for further research is the issue of finding a solution to the initial problem using a quasi-Newton method. Although it seems theoretically simple and clear that it should improve the performance, numerical experiments have shown that the initial system obtained in that way may not lead to a better algorithm. Actually, there are examples where initial systems obtained using the quasi-Newton method improved and examples where they corrupted the performance of the algorithm.

The issue of solving the initial system is related to the choice of the homotopy map. Similarly, although it seems obvious that the map (17) should perform better, in some cases that was not true. Numerical data on this issue are limited since solutions to the initial problem have been obtained for only a few small examples.

There are still many open questions and ways to improve the algorithms. It seems that the most important issues are finding better ways to solve the initial problem and improving the performance of the homotopy map (17).

5. Conclusion.

This paper has considered the use of probability-one homotopy methods to solve the optimal projection equations for the model reduction problem. Four different approaches were given for solving the equations. The first approach is based on solving the optimal projection equations in their original form. The three other approaches stem from a decomposition of the pseudogramians based on a contragredient transformation and proved to be more numerically robust than the first approach. The "best" algorithm was shown to be effective in finding the optimal reduced order models for a number of examples.

The number of variables associated with the first approach is of order n^2 (n is the dimension of the original model), while the number of variables for the latter approaches is of order $n\,n_m$ (n_m is the dimension of the reduced order model). Future research will involve the development of homotopy algorithms with fewer variables. It appears that by using a more rudimentary form of the optimal projection equations, it is possible to reduce the number of variables to be of order $n(m+l)$ (m and l are, respectively, the number of inputs and outputs). Future research will also consider the H_2/H_∞ reduced order model problem [7].

REFERENCES

[1] M. Aoki, "Control of large-scale dynamic systems by aggregation," *IEEE Trans. Autom. Contr.*, vol. AC-13, pp. 246–253, 1968.

[2] R. H. Bartels and G. W. Stewart, "Solution of the matrix equation $A\,X + X\,B = C$," *Comm. of the ACM*, vol. 15, pp. 820–826, 1972.

[3] D. S. Bernstein and D. C. Hyland, "The optimal projection equations for reduced-order state estimation," *IEEE Trans. Autom. Contr.*, vol. AC-30, pp. 583–585, 1985.

[4] S. L. Campbell and C. D. Meyer, Jr., *Generalized Inverses of Linear Transformations*, London, England: Pitman, 1979, ch. 7, pp. 120–127.

[5] E. G. Collins, Jr., D. J. Phillips and D. C. Hyland, "Robust decentralized control laws for the ACES structure," *Contr. Sys. Mag.*, April 1991, pp. 62–70.

[6] K. Glover, "All optimal Hankel-norm approximations of linear multivariable systems and their L^∞-error bounds," *Int. J. Contr.*, vol. 39, pp. 1115–1193, 1984.

[7] W. M. Haddad and D. S. Bernstein, "Combined L_2/H_∞ model reduction," *Int. J. Contr.*, vol. 49, pp. 1523–1535, 1989.

[8] J. Hickin and N. K. Sinha, "Model reduction for linear multivariable systems," *IEEE Trans. Autom. Contr.*, vol. AC-25, pp. 1121–1127, 1980.

[9] D. C. Hyland and D. S. Bernstein, "The optimal projection equations for fixed-order dynamic compensation," *IEEE Trans. Autom. Contr.*, vol. AC-29, pp. 1034–1037, 1984.

[10] D. C. HYLAND AND D. S. BERNSTEIN, "The optimal projection equations for model reduction and the relationships among the methods of Wilson, Skelton, and Moore," *IEEE Trans. Autom. Contr.*, vol. AC-30, pp. 1201–1211, 1985.

[11] P. T. KABAMBA, "Balanced gains and their significance for L^2 model reduction," *IEEE Trans. Autom. Contr.*, vol. AC-30, pp. 690–693, 1985.

[12] S. KUNG AND D. W. LIN, "Optimal Hankel-norm model reductions: Multivariable systems," *IEEE Trans. Autom. Contr.*, vol. AC-26, pp. 832–852, 1981.

[13] S. KUNG AND D. W. LIN, "A state-space formulation for optimal Hankel-norm approximations," *IEEE Trans. Autom. Contr.*, vol. AC-26, pp. 942–946, 1981.

[14] C. P. KWONG, "Optimal chained aggregation for reduced-order modelling," *Int. J. Contr.*, vol. 35, pp. 964–982, 1982.

[15] B. C. MOORE, "Principal component analysis in linear systems: Controllability, observability, and model reduction," *IEEE Trans. Autom. Contr.*, vol. AC-26, pp. 17–32, 1981.

[16] L. PERNEBO AND L. M. SILVERMAN, "Model reduction via balanced state space representations," *IEEE Trans. Autom. Contr.*, vol. AC-27, pp. 382–387, 1982.

[17] S. RICHTER AND E. G. COLLINS, JR., "A homotopy algorithm for reduced order compensator design using the optimal projection equations," in *Proc. IEEE Conf. Dec. Contr.*,, Tampa, FL, pp. 506–511, Dec. 1989.

[18] R. E. SKELTON, "Cost decomposition of linear systems with application to model reduction," *Int. J. Contr.*, vol. 32, pp. 1031–1055, 1980.

[19] R. E. SKELTON AND P. C. HUGHES, "Modal cost analysis for linear matrix-second-order systems," *J. Dynamic Systems, Measurement and Control*, vol. 102, pp. 151–158, 1980.

[20] R. E. SKELTON AND A. YOUSUFF, "Component cost analysis of large scale systems," *Int. J. Contr.*, vol. 37, pp. 285–304, 1983.

[21] S. SMALE, "A convergent process of price adjustment and global Newton methods," *J. Math. Econom.*, vol. 3, pp. 107–120, 1976.

[22] J. T. SPANOS, M. H. MILMAN AND D. L. MINGORI, "Optimal model reduction and frequency-weighted extension," *AIAA paper number AIAA–90–3345–CP*, pp. 271–284, 1990.

[23] C. DE VILLEMAGNE AND R. E. SKELTON, "Model reductions using a projection formulation," *Int. J. Contr.*, vol. 46, pp. 2141–2169, 1987.

[24] L. T. WATSON, "Globally convergent homotopy methods: A tutorial," *Appl. Math. Comput.*, vol. 31BK, pp. 369–396, 1989.

[25] L. T. WATSON, "Globally convergent homotopy algorithms for nonlinear systems of equations," *Nonlinear Dynamics*, vol. 1, pp. 143–191, 1990.

[26] L. T. WATSON, "Numerical linear algebra aspects of globally convergent homotopy methods," *SIAM Rev.*, vol. 28, pp. 529–545, 1986.

[27] L. T. WATSON, S. C. BILLUPS AND A. P. MORGAN, "HOMPACK: A suite of codes for globally convergent homotopy algorithms," *ACM Trans. Math. Software*, vol. 13, pp. 281–310, 1987.

[28] D. A. WILSON, "Optimum solution of model-reduction problem," *Proc. IEE*, vol. 117, pp. 1161–1165, 1970.

[29] D. ŽIGIĆ, "Homotopy methods for solving the optimal projection equations for the reduced order model problem," , M.S. thesis, Dept. of Computer Science, Virginia Polytechnic Institute and State University, Blacksburg, VA, June 1991.