Iterative Solution of Algebraic Riccati Equations using a Modified Newton-Kleinman Method MTNS 2004

Kirsten Morris
Department of Applied Mathematics
University of Waterloo
Waterloo, CANADA N2L 3G1
kmorris@uwaterloo.ca

Carmeliza Navasca Dept. of Mathematics
University of California
Los Angeles, CA 90095 USA
navasca@math.ucla.edu

Abstract

Algebraic Riccati equations of large dimension arise when using approximations to design controllers for systems modelled by partial differential equations. For large model order direct solution methods based on eigenvector calculation fail. In this paper we describe an iterative method that takes advantage of several special features of these problems. The algorithm is straightforward to code. Performance is illustrated with a number of standard examples.

Introduction

A classical controller design objective is to find a control $u(t)$ so that the objective function

$$\int_0^\infty \langle x(t), Qx(t)\rangle + \langle u^*(t)Ru(t)\rangle dt \tag{1}$$

is minimized where R is a symmetric positive definite matrix and Q is symmetric positive semi-definite. As is well-known, the solution to this problem is found by solving an algebraic Riccati equation

$$A^*P + PA - PBR^{-1}B^*P = -Q. \tag{2}$$

for a feedback operator

$$K = -R^{-1}B'P. \tag{3}$$

If the system is modelled by partial differential or delay differential equations then the state $x(t)$ lies in an infinite-dimensional space. The theoretical solution to this problem for many infinite-dimensional systems parallels the theory for finite-dimensional systems [9, 10, 16, 17, e.g.]. In practice, the control is calculated through approximation. The matrices A, B, C arise in a finite dimensional approximation of the infinite dimensional system. Let n indicate the order of the approximation, m the number of control inputs and p the number of observations. Thus, A is $n \times n$, B is $n \times m$ and C is $p \times n$. There have been many papers written describing conditions under which approximations lead to approximating controls that converge to the control for the original infinite-dimensional system [3, 10, 13, 16, 17, e.g.]. In this paper we will assume that an approximation has been chosen so that a solution to the Riccati equation (2) exists for sufficiently large n and also that the approximating feedback operators converge. Unfortunately, particularly in partial differential equation models with more than one space dimension, many infinite-dimensional control problems lead to Riccati equations of large order.

For problems where the model order is small, $n < 50$, a direct method based on calculating the eigenvectors of the associated Hamiltonian works well [18]. Due the limitations of the calculation of eigenvectors for large non-symmetric matrices, this method is not suitable for problems where n becomes large. Two common methods are Chrandrasekhar [2, 6] and Newton-Kleinman iterations [12, 15, 25]. Here we use a modification to the Newton-Kleinman method first proposed by by Banks and Ito [2] as a refinement for a partial solution to the Chandraskehar equation.

Solution of the Lyapunov equation is a key step in implementing either modified or standard Newton-Kleinman. The Lyapunov equations arising in the Newton-Kleinman method have several special features: (1) the model order n is generally much larger than the number of controls m or number of observations p and (2) the matrices are often sparse. We use a recently developed method [19, 24] that uses these features, leading to an efficient algorithm.

In the next section we describe the solution method and show convergence. We then illustrate the approach with a number of standard control examples, including one with several space variables. Our results indicate that with the Lyapunov solver used here modified Newton-Kleinman achieves considerable savings in computation time over standard Newton-Kleinman.

1 Description of Algorithm

One approach to solving large Riccati equations is the Newton-Kleinman method [15]. The Riccati equation (2) can be rewritten as

$$(A - BK)^* P + P(A - BK) = -Q - K^* R K. \tag{4}$$

We say a matrix A_o is *Hurwitz* if $\sigma(A_o) \subset \mathbb{C}_-$ If $A - BK$ is Hurwitz, then the above equation is a Lyapunov equation.

Theorem 1.1 *[15] Consider a stabilizable pair (A, B) with a feedback K_0 so that $A - BK_0$ is Hurwitz. Define $S_i = A - BK_i$, and solve the Lyapunov equation*

$$S_i^* P_i + P_i S_i = -Q - K_i^* R K_i \tag{5}$$

for P_i and then update the feedback as $K_{i+1} = -R^{-1}B^*P_i$. Then

$$\lim_{i\to\infty} P_i = P$$

with quadratic convergence.

The key assumption in the above theorem is that an initial estimate, or ansatz, K_0, such that $A - BK_0$ is Hurwitz is available. For an arbitrary large Riccati equation, this condition may be difficult to satisfy. However, this condition is not restrictive for Riccati equations arising in control of infinite-dimensional systems. First, many of these systems are stable even when uncontrolled and so the initial iterate K_0 may be chosen as zero. Second, if the approximation procedure is valid then convergence of the feedback gains is obtained with increasing model order. Thus, a gain obtained from a lower order approximation, perhaps using a direct method, may be used as an ansatz for a higher order approximation. This technique was used successfully in [12, 23, 25].

A modification to this method was proposed by Banks and Ito [2]. In that paper, they partially solve the Chandrasekhar equations and then use the resulting feedback K as a stabilizing initial guess. Instead of the standard Newton-Kleinman iteration (5) above, Banks and Ito rewrote the Riccati equation in the form

$$\begin{aligned} (A - BK_i)^*X_i + X_i(A - BK_i) &= -D_i^*RD_i, \qquad D_i = K_i - K_{i-1}, \\ K_{i+1} &= K_i - B^*X_i. \end{aligned} \tag{6}$$

Corollary 1.1 *Consider equation (6) where K_0, K_1 are chosen so that $A-BK_0$ and $A-BK_1$ are Hurwitz. Then*

$$\lim_{i\to\infty} K_i = K$$

with quadratic convergence.

Proof: Consider two iterations of the Newton-Kleinman method:

$$(A - BK_{i-1})^*P_{i-1} + P_{i-1}(A - BK_{i-1}) = -Q - K_{i-1}^*RK_{i-1},, \tag{7}$$

$$(A - BK_i)^*P_i + P_i(A - BK_i) = -Q - K_i^*RK_i. \tag{8}$$

Subtract (8) from (7) and rearrange, using $K_i = R^{-1}B^*P_{i-1}$ to obtain (6) with $X_i = P_{i-1} - P_i$. Convergence follows from convergence of the Newton-Kleinman method.

Solution of a Lyapunov equation is a key step in each iteration of the Newton-Kleinman method, both standard and modified. Consider the Lyapunov equation

$$XA_o + A_o^*X = -DD^* \tag{9}$$

where $A_o \in \mathbb{R}^{n\times n}$ is Hurwitz and $D \in \mathbb{R}^{n\times r}$. Factor Q into C^*C and let n_Q indicate the rank of C. In the case of standard Newton-Kleinman, $r = m + n_Q$ while for modified Newton-Kleinman, $r = m$. If A_o is Hurwitz, then the Lyapunov equation has a symmetric positive semidefinite solution X. As for the Riccati equation, direct methods such as Bartels-Stewart [4] are only appropriate for low model order and do not take advantage of sparsity in the matrices.

In [2] Smith's method was used to solve the Lyapunov equations. For $p < 0$, define $U = (A_o - pI)(A_o + pI)^{-1}$ and $V = -2p(A_o^* + pI)^{-1}DD^*(A_o + pI)^{-1}$. The Lyapunov equation (9) can be rewritten as

$$X = U^*XU + V.$$

In Smith's method [27], the solution X is found by using successive substitutions: $X = \lim_{i\to\infty} X_i$ where

$$X_i = U^*X_{i-1}U + V \tag{10}$$

with $X_0 = 0$. Convergence of the iterations can be improved by careful choice of the parameter p *e.g.* [26, pg. 297]. This method of successive substitition is unconditionally convergent, but has only linear convergence.

The ADI method [20, 29] improves Smith's method by using a different parameter p_i at each step. Two alternating linear systems,

$$\begin{aligned} (A_o^* + p_iI)X_{i-\frac{1}{2}} &= -DD^* - X_{i-1}(A_o - p_iI) &(11)\\ (A_o^* + p_iI)X_i^* &= -DD^* - X_{i-\frac{1}{2}}^*(A_o - p_iI) &(12)\end{aligned}$$

are solved recursively starting with $X_0 = 0 \in \mathbb{R}^{n\times n}$ and parameters $p_i < 0$. If all parameters $p_i = p$ then equations (11,12) reduce to Smith's method. If the ADI parameters p_i are chosen appropriately, then convergence is obtained in J iterations where $J \ll n$. The parameter selection problem has been studied extensively [8, 20, 28, e.g.].

If A_o is sparse, then sparse arithmetic can be used in calculation of X_i. However, full calculation of the dense iterates X_i is required at each step. Setting $X_0 = 0$, it can be easily shown that X_i is symmetric and positive semidefinite for all i, and so we can write $X = ZZ^*$ where Z is a Cholesky factor of X [19, 24]. (A Cholesky factor does not need to be square or be lower triangular.) Substituting $Z_iZ_i^*$ for X_i and setting $X_0 = 0$, we obtain the following iterates for

$$\begin{aligned} Z_1 &= \sqrt{-2p_1}(A_o^* + p_1I)^{-1}D \\ Z_i &= [\sqrt{-2p_i}(A_o^* + p_iI)^{-1}D,\ \ (A_o^* + p_iI)^{-1}(A_o^* - p_iI)Z_{i-1}] &(13)\end{aligned}$$

Note that $Z_1 \in \mathbb{R}^{n\times r}$, $Z_2 \in \mathbb{R}^{n\times 2r}$, and $Z_i \in \mathbb{R}^{n\times ir}$.

The algorithm is stopped when the Cholesky iterations converge within some tolerance. In [19] these iterates are reformulated in a more efficient form, using the observation that the order in which the ADI parameters are used is irrelevant. This leads to the algorithm shown in Table 2.

This solution method results in considerable savings in computation time and memory. In calculation of the feedback K, the full solution X never needs to be constructed. Also, the complexity of this method (CF-ADI) is also considerably less than that of standard ADI as shown in Table 1. Recall that for standard Newton-Kleinman, r is the sum of the rank of the state weight n_Q and controls m. For modified Newton-Kleinman r is equal to m. The reduction in complexity of CF-ADI method over ADI is marked for the modified Newton-Kleinman method.

Table 1: Complexity of CF-ADI and ADI [19]

	CF-ADI	ADI
Sparse A	$\mathcal{O}(Jrn)$	$\mathcal{O}(Jrn^2)$
Full A	$\mathcal{O}(Jrn^2)$	$\mathcal{O}(Jrn^3)$

Table 2: Cholesky-ADI Method

Given A_o and D
Choose ADI parameters $\{p_1, \ldots, p_J\}$ with $\Re(p_i) < 0$
Define $z_1 = \sqrt{-2p_1}(A_o^* + p_1 I)^{-1} D$
and $Z_1 = [z_1]$
For $i = 2, \ldots, J$
Define $W_i = (\frac{\sqrt{-2p_{i+1}}}{\sqrt{-2p_i}})[I - (p_{i+1} - p_i)(A_o^* + p_{i+1} I)^{-1}]$
(1) $z_i = W_i z_{i-1}$
(2) If $\|z\| > tol$
$Z_i = [Z_{i-1} \;\; z_i]$
Else, stop.

2 Examples

In this section we test the algorithm with a number of standard examples. All computations were done within MATLAB on a computer with two 1.2 GHz AMD processors. The relative error for the Cholesky iterates was set to 10^{-8}. For each example we compare the number of Newton-Kleinman iterations, Lyapunov solution steps and CPU time for both standard Newton-Kleinman and modified Newton-Kleinman. The iteration count and CPU time for modified Newton-Kleinman includes the cost of initially calculating the additional feedback K_1 using one iteration of standard Newton-Kleinman. The cost of calculating K_0 is not included in any counts or CPU times.

One-dimensional Heat Equation

Consider the linear quadratic regulator problem of minimizing a cost functional [2, 7]

$$J(u) = \int_0^\infty (|Cz(t)|^2 + |u(t)|^2) dt$$

subject to

$$\begin{aligned} \frac{\partial z(t,x)}{\partial t} &= \frac{\partial^2 z(t,x)}{\partial x^2}, \; x \in (0,1), \\ z(0,x) &= \psi(x) \end{aligned} \tag{14}$$

Table 3: 1d Heat Equation with single observation: Standard Newton-Kleinman

n	Newton-Kleinman Itn's	Lyapunov Itn's	CPU time
25	11	19,22,23,26,27,29 30,31,31,31,31	0.83
50	11	24,26,28,30,32,34 35,35,35,35,35	1.2
100	11	28,31,32,35,36,38 39,40,40,40,40	3.49
200	11	33,35,37,39,41,43 44,44,44,44,44	23.1

with boundary conditions

$$\begin{aligned} \frac{\partial z(t,0)}{\partial x} &= u(t) \\ \frac{\partial z(t,1)}{\partial x} &= 0. \end{aligned} \tag{15}$$

We choose $R = 1$ and $Q = C^*C$ where

$$Cz(t) = \int_0^1 z(t,x)dx. \tag{16}$$

The solution to the infinite-dimensional Riccati equation is

$$Kz = \int_0^1 k(x)z(x)dx$$

where $k = 1$ [3]. Thus, for this problem we have an exact solution to which we can compare the approximations. The equations (14-16) are discretized using the standard Galerkin approximation with linear spline finite element basis on an uniform partition of $[0, 1]$. Both algorithms converged to the exact solution.

In Tables 3 and 4 we compare the number of Newton-Kleinman and Lyapunov iterations as well as the CPU time per order n. We use the ansatz $k_0(x) = 100$ for all n. With the modified algorithm, there are $1 - 2$ fewer Riccati loops than with the the original Newton-Kleinman iteration. Also, the modified Newton-Kleinman method requires fewer Lyapunov iterations within the last few Newton-Kleinman loops. The computation time with the modified Newton-Kleinman algorithm is significantly less than that of the original algorithm. Tables (5) and (6) are similar, except that $Q = I$. As predicted by the complexity analysis, the reduction in number of iterations and computation time is greater than for the case of a single observation.

Table 4: 1d Heat Equation with single observation: Modified Newton-Kleinman

n	Newton-Kleinman Itn's	Lyapunov Itn's	CPU time
25	10	19,22,23,26,27,29 30,31,29,1	0.66
50	10	24,26,28,30,32,34 35,35,33,1	0.94
100	9	28,31,32,35,36,38 39,40,1	2.32
200	9	33,35,37,39,41,43 44,44,1	14.2

Table 5: 1d Heat Equation with Q=I: Standard Newton-Kleinman

n	Newton-Kleinman It's	Lyapunov It's	cputime
25	13	23,23,23,26,27,30,32,34 35,36,36,36,36	1.48
50	14	27,27,28,30,32,34,36,38 40,41,42,42,42,42	4.89
100	14	32,32,32,35,36,39,41,43 45,46,47,47,47,47	22.27
200	15	36,36,37,39,41,43,45,47 49,51,52,53,53,53,53	178.17

Table 6: 1d Heat Equation with Q=I: Modified Newton-Kleinman

n	Newton-Kleinman It's	Lyapunov It's	cputime
25	12	19,22,23,26,27,30 32,34,35,35,34,1	0.87
50	12	24,26,28,30,32,34 36,38,40,41,41,1	1.35
100	12	28,31,32,35,36,39 41,43,45,46,46,1	4.12
200	11	33,35,37,39,41,43 45,47,49,51,1	24.35

Two-Dimensional Heating Problem

Define the rectangle $\Omega = [0,1] \times [0,1]$ with boundary $\partial\Omega$. Consider the two-dimensional partial differential equation [7]

$$\begin{array}{lcll} \frac{\partial z}{\partial t} & = & \frac{\partial^2 z}{\partial x^2} + \frac{\partial^2 z}{\partial y^2} + 20\frac{\partial z}{\partial y} + 100z = f(x,y)u(t), & (x,y) \in \Omega \\ z(x,y,t) & = & 0, & (x,y) \in \partial\Omega \end{array} \tag{17}$$

where z is a function of x, y and t. Let

$$f(x,y) = \begin{cases} 100, & \text{if .1 < x < .3 and .4 < y < .6} \\ 0, & \text{else} \end{cases}.$$

Central difference approximations are used to discretize (17) on a grid of $N \times M$ points. The resulting approximation has dimension $n = N \times M$: $A \in \mathbb{R}^{n\times n}$ and $B \in \mathbb{R}^{n\times 1}$. The A matrix is sparse with at most 5 non-zero entries in any row. The B matrix is a sparse column vector. We chose $C = B^*$ and $R = 1$.

We solved the Riccati equation on a number of grids, using both standard and modified Newton-Kleinman methods. The data is shown in Tables 7 and 8. We also investigated the performance of the algorithm with the same partial differential equation (17) and numerical approximation, but with control weight $R = 100$, and cost $C = I$ so that $Q = I$. The results are shown in Tables 9 and 10. Modified Newton-Kleinman is clearly much more efficient. Fewer Lyapunov iterations are required for convergence and there is a significant reduction in computation time. The advantage of modified Newton-Kleinman over standard Newton-Kleinman is more pronounced for the case where the state weight is full rank.

We also investigated the use of non-zero initial estimates for K in reducing computation time. We first solve the Riccati equation on a 12×12 grid. Since $\sigma(A) \subset \mathbb{C}_-$, $K_0^{144} = 0$ is a possible ansatz. It required 13 Newton-Kleinman iterations and a total of 419 Lyapunov iterations to obtain a relative error in K of 10^{-11}. Linear interpolation was used to project this solution to a function on a finer grid, 23×12, where $n = 276$. Indicate this projection by $K_{proj}^{12\times 12}$. On the finer grid 23×12 where $n = 276$, we used both zero and $K_{proj}^{12\times 12}$ as initial estimates. As indicated in Table 8, the error of K was 10^{-12} after only 150 Lyapunov and 54 Newton-Kleinman iterations. The same procedure is applied to generate a guess K_0^{529} where the mesh is 23×23 and $n = 529$. Neglecting the computation time to perform the projection, use of a previous solution lead a total computation time over both grids of only 6.3 seconds versus 11.9 seconds for $n = 276$ and a total computation time of only 27.8 versus 79.5 for $n = 529$. Similar improvements in computation time were obtained with standard Newton-Kleinman.

Euler-Bernoulli Beam

Consider a Euler-Bernoulli beam clamped at one end ($r = 0$) and free to vibrate at the other end ($r = 1$). Let $w(r,t)$ denote the deflection of the beam from its rigid body motion at time t and position r. The deflection is controlled by applying a a torque $u(t)$ at the clamped end ($r = 0$). We assume that the hub inertia I_h is much larger than the beam inertia I_b so that

Table 7: 2-d Heat Equation with single observation: Standard Newton-Kleinman

grid	K_0	n	Newton-Kleinman Itn's	Lyapunov Itn's	CPU time
12×12	0	144	14	12,44,41,39,36,34,32 30,28,27,27,27,27,27	4.98
23×12	0	276	15	16,47,45,42,40,38,35,33 31,30,30,30,30,30,30	22.2
23×12	K_{proj}^{12x12}	276	6	29,30,30,30,30,30	10.8
23×23	0	529	16	20,51,48,46,44,41,39,37 35,34,33,33,32,32,32,32	139.
23×23	K_{proj}^{23x12}	529	5	33,32,32,32,32	65.7

Table 8: 2-d Heat Equation with single observation: Modified Newton-Kleinman

grid	K_0	n	Newton-Kleinman Itn's	Lyapunov Itn's	CPU time
12×12	0	144	13	12,44,41,39,36,34,32 30,28,27,27,27,1	2.94
23×12	0	276	13	16,47,45,42,40,38,35 33,31,30,30,30,29	11.9
23×12	K_{proj}^{12x12}	276	4	29,30,30,29	3.36
23×23	0	529	14	20,50,48,46,44,41,39 37,35,34,33,33,33,31	79.5
23×23	K_{proj}^{23x12}	529	4	33,32,32,1	21.5

Table 9: 2-d Heat Equation with $C = I$: Standard Newton-Kleinman

$n \times m$	Initial Guess	N	Newton-Kleinman It's	Lyapunov It's	cputime
12×12	K^{12x12}=0	144	7	14,14,14,14,14,14,14	10.73
23×12	K^{23x12}=0	276	7	18,18,18,18,18,18,18	95.39
23×12	$K_{mg}^{23x12} = K_{proj}^{12x12}$	276	5	18,18,18,18, 18,18	27.7
23×23	K^{23x23}=0	529	7	20,21,21,21,21,21,21	585.94
23×23	$K_{mg}^{23x23} = K_{proj}^{23x12}$	529	5	20,21,21,21,21	461.05

Table 10: 2-d Heat Equation with $C = I$: Modified Newton-Kleinman

$n \times m$	Initial Guess	N	Newton-Kleinman It's	Lyapunov It's	cputime
12×12	K^{12x12}=0	144	6	12,14,14,14,13,1	1.46
23×12	K^{23x12}=0	276	6	16,18,18,18,17,1	10.95
23×12	$K_{mg}^{23x12} = K_{proj}^{12x12}$	276	4	18,18,17,7	10.75
23×23	K^{23x23}=0	529	6	20,20,21,20,20,1	88
23×23	$K_{mg}^{23x23} = K_{proj}^{23x12}$	529	4	21,20,20,18	84.49

$I_h\ddot{\theta} \approx u(t)$. The partial differential equation model with Kelvin-Voigt and viscous damping is

$$w_{tt}(r,t) + C_v w_t(r,t) + \frac{\partial^2}{\partial r^2}\left[C_d I_b w_{rrt}(x,t) + \frac{EI_r}{\rho A} w_{rr}(r,t)\right] = \frac{\rho r}{I_h} u(t), \tag{18}$$

with boundary conditions

$$\begin{aligned} w(0,t) &= 0 \\ w_r(1,t) &= 0. \\ EIw_{rr}(1,t) + C_d I_b w_{rrt}(1,t) &= 0 \\ \frac{\partial}{\partial r}\left[EI(1)w_{rr}(r,t) + C_d I_b w_{rrt}(r,t)\right]_{r=1} &= 0. \end{aligned}$$

The values of the physical parameters in Table 2 are as in [1].

Define H be the closed linear subspace of the Sobolev space $H^2(0,1)$

$$H = \left\{ w \in H^2(0,1) : w(0) = \frac{dw}{dr}(0) = 0 \right\}$$

and define the state-space to be $X = H \times L^2(0,1)$ with state $z(t) = (w(\cdot,t), \frac{\partial}{\partial t} w(\cdot,t))$. A state-space formulation of the above partial differential equation problem is

$$\frac{d}{dt}x(t) = Ax(t) + Bu(t),$$

where

$$A = \begin{bmatrix} 0 & I \\ -\frac{EI}{\rho}\frac{d^4}{dr^4} & -\frac{C_d I}{\rho}\frac{d^4}{dr^4} - \frac{C_v}{\rho} \end{bmatrix}, \qquad B = \begin{bmatrix} 0 \\ \frac{r}{I_h} \end{bmatrix},$$

with domain

$$\begin{aligned} \mathrm{dom}\,(A) = \{(\phi,\psi) \in X : \psi \in H \ \text{ and} \\ M = EI\tfrac{d^2}{dr^2}\phi + C_d I \tfrac{d^2}{dr^2}\psi \in H^2(0,1) \ \text{ with } M(L) = \tfrac{d}{dr}M(L) = 0\}. \end{aligned}$$

E	2.68×10^{10} N/m^2
I_b	1.64×10^{-9} m^4
ρ	1.02087 kg/m
C_v	2 Ns/m
C_d	2.5×10^8 Ns/m
L	1 m
I_h	121.9748 $kg\ m^2$
d	.041 kg^{-1}

Table 11: Table of physical parameters.

Table 12: Beam with measurement of tip position, $K_0 = 0$: Newton-Kleinman

n	Newton-Kleinman It's	Lyapunov It's	cputime
48	2	407,407	5.02
72	2	461,461	8.140
96	2	499,499	12.25
120	2	522,522	18.28

We use $R = 1$ and define C by the tip position:

$$w(1,t) = C[w(x,t) \ \ \dot{w}(x,t)].$$

Then $Q = C^*C$. We also solved the control problem with $Q = I$.

Let $H^{2N} \subset H$ be a sequence of finite-dimensional subspaces spanned by the standard cubic B-splines with a uniform partition of $[0,1]$ into N subintervals. This yields an approximation in $H^{2N} \times H^{2N}$ [14, e.g.] of dimension $n = 4N$. This approximation method yields a sequence of solutions to the algebraic Riccati equation that converge strongly to the solution to the infinite-dimensonal Riccati equation corresponding to the original partial differential equation description [3, 21].

The control problem was solved with several different choices of K_0. The results are shown in Tables 12-19. In all cases, modified Newton-Kleinman converged with a significant reduction in iterations and CPU time over standard Newton-Kleinman. As for the other examples, reduction in CPU time is greater for full state observation than for a single observation.

References

[1] H.T. Banks and D. J. Inman, "On Damping Mechanisms in Beams", *ICASE Report No. 89-64*, NASA, Langley, 1989.

Table 13: Beam with measurement of tip position, $K_0 = 0$: Modified Newton-Kleinman

n	Newton-Kleinman It's	Lyapunov It's	cputime
48	1	1	2.48
72	1	1	3.66
96	1	1	5.33
120	1	1	7.17

Table 14: Beam with measurement of tip position, $K_0 = B^*$: Newton-Kleinman

n	Newton-Kleinman It's	Lyapunov It's	cputime
48	3	407,407,407	6.69
72	3	461,461,461	11.00
96	3	499,499,499	16.98
120	3	522,522,522	24.64

Table 15: Beam with measurement of tip position, $K_0 = B^*$: Modified Newton-Kleinman

n	Newton-Kleinman It's	Lyapunov It's	cputime
48	2	417,1	3.81
72	2	471,1	5.9
96	2	509,1	8.81
120	2	544,1	12.82

Table 16: Beam with full state measurement, $K_0 = 0$: Newton-Kleinman

n	Newton-Kleinman It's	Lyapunov It's	cputime
48	2	437,437	44.23
72	2	491,491	123.24
96	2	527,527	239.62
120	2	557,557	417.48

Table 17: Beam with full state measurement, $K_0 = 0$: Modified Newton-Kleinman

n	Newton-Kleinman It's	Lyapunov It's	cputime
48	1	1	15.42
72	1	1	40.83
96	1	1	79.13
120	1	1	138.25

Table 18: Beam with full state measurement, $K_0 = B'$: Newton-Kleinman

n	Newton-Kleinman It's	Lyapunov It's	cputime
48	3	437,437,437	58.91
72	3	502,502,502	155.67
96	3	527,527,527	314.4
120	3	5,5,5	551.75

Table 19: Beam with full state measurement, $K_0 = B'$: Modified Newton-Kleinman

n	Newton-Kleinman It's	Lyapunov It's	cputime
48	2	417,1	16.79
72	2	471,1	41.76
96	2	509,1	82.16
120	2	544,1	142.32

[2] H. T. Banks and K. Ito, *A Numerical Algorithm for Optimal Feedback Gains in High Dimensional Linear Quadratic Regulator Problems*, SIAM J. Control Optim., vol. 29, no. 3, 499-515, 1991.

[3] H. T. Banks and K. Kunisch, *The linear regulator problem for parabolic systems*, SIAM J. Control Optim., vol. 22, 5, 684-698, 1984.

[4] R.H. Bartels and W. Stewart, "Solution of the matrix equation AX+XB =C" *Comm. of ACM*, Vol. 15, pg. 820-826, 1972.

[5] D. Bau, and L. Trefethen, *Numerical Linear Algebra*, SIAM, Philadephia, 1997.

[6] J.A. Burns and K. P. Hulsing, "Numerical Methods for Approximating Functional Gains in LQR Boundary Control Problems", *Mathematical and Computer Modelling*, vol. 33, no. 1, pg. 89-100, 2001.

[7] Y. Chahlaoui and P. Van Dooren, *A collection of Benchmark examples for model reduction of linear time invariant dynamical systems*, SLICOT Working Note 2002-2, http://www.win.tue.nl/niconet/

[8] N. Ellner and E. L. Wachpress, *Alternating Direction Implicit Iteration for Systems with Complex Spectra*, SIAM J. Numer. Anal., vol. 28, no. 3, 859-870, 1991.

[9] J.S. Gibson, "The Riccati Integral Equations for Optimal Control Problems on Hilbert Spaces", *SIAM J. Control and Optim.*, Vol. 17, pg. 637-565, 1979.

[10] J.S. Gibson, "Linear-quadratic optimal control of hereditary differential systems: infinite dimensional Riccati equations and numerical approximations", *SIAM J. Control and Optim.*, Vol. 21 95-139, 1983.

[11] G.H. Golub and C.F. van Loan, *Matrix Computations*, John Hopkins, 1989.

[12] J.R. Grad and K. A. Morris, "Solving the Linear Quadratic Control Problem for Infinite-Dimensional Systems", *Computers and Mathematics with Applications*, Vol. 32, No. 9, 1996, pg. 99-119.

[13] K. Ito, Strong convergence and convergence rates of approximating solutions for algebraic Riccati equations in Hilbert spaces, *Distributed Parameter Systems*, eds. F. Kappel, K. Kunisch, W. Schappacher, Springer-Verlag, 1987, 151-166.

[14] K. Ito and K. A. Morris, *An approximation theory of solutions to operator Riccati equations for H^∞ control*, SIAM J. Control Optim., 36, 82-99, 1998.

[15] D. Kleinman, *On an Iterative Technique for Riccati Equation Computations*, IEEE Transactions on Automat. Control, 13, 114-115, 1968.

[16] I. Lasiecka and R. Triggiani, *Control Theory for Partial Differential Equations: Continuous and Approximation Theories*, Part 1, Cambridge University Press, 2000.

[17] I. Lasiecka and R. Triggiani, *Control Theory for Partial Differential Equations: Continuous and Approximation Theories*, Part 2, Cambridge University Press, 2000.

[18] A.J. Laub, "A Schur method for solving algebraic Riccati equations", *IEEE Trans. Auto. Control*, Vol. 24, pg. 913-921, 1979.

[19] J. R. Li and J White, *Low rank solution of Lyapunov equations*, *SIAM J. Matrix Anal. Appl.*, Vol. 24, pg. 260-280, 2002.

[20] A. Lu and E. L. Wachpress, *Solution of Lyapunov Equations by Alternating Direction Implicit Iteration*, Computers Math. Applic., 21, 9, 43-58, 1991.

[21] K.A. Morris, "Design of Finite-Dimensional Controllers for Infinite-Dimensional Systems by Approximation", *J. Mathematical Systems, Estimation and Control*, 4, 1-30, 1994.

[22] K.A. Morris, *Introduction to Feedback Control*, Harcourt-Brace, 2001.

[23] Morris, K. and Navasca, C., "Solution of Algebraic Riccati Equations Arising in Control of Partial Differential Equations", in *Control and Boundary Analysis*, ed. J. Cagnola and J-P Zolesio, Marcel Dekker, 2004.

[24] T. Penzl, "A Cyclic Low-Rank Smith Method for Large Sparse Lyapunov Equations", *SIAM J. Sci. Comput.*, Vol. 21, No. 4, pg. 1401-1418, 2000.

[25] I. G. Rosen and C. Wang, "A multilevel technique for the approximate solution of operator Lyapunov and algebraic Riccati equations", *SIAM J. Numer. Anal.* Vol. 32 514–541, 1994.

[26] Russell, D.L., *Mathematics of Finite Dimensional Control Systems: Theory and Design,* Marcel Dekker, New York, 1979.

[27] R.A. Smith, "Matrix Equation XA + BX = C", *SIAM Jour. of Applied Math.*, Vol. 16, pg. 198-201, 1968.

[28] E. L. Wachpress, *The ADI Model Problem*, preprint, 1995.

[29] E. Wachpress, "Iterative Solution of the Lyapunov Matrix Equation", *Appl. Math. Lett.*, Vol. 1, Pg. 87-90, 1988.