Approximation of linear quadratic feedback control for partial differential equations

Kirsten Morris* Carmeliza Navasca†

Abstract

Algebraic Riccati equations (ARE) of large dimension arise when using approximations to design controllers for systems modelled by partial differential equations. We use a modified Newton method to solve the ARE that takes advantage of several special features of these problems. The modified Newton method leads to a right-hand side of rank equal to the number of inputs regardless of the weights. Thus, the resulting Lyapunov equation can be more efficiently solved. The Cholesky-ADI algorithm is used to solve the Lyapunov equation resulting at each step. The algorithm is straightforward to code. Performance is illustrated with a number of standard examples. An example on controlling the deflection of the Euler-Bernoulli beam indicates that for weakly damped problems a low rank solution to the ARE may not exist. Further analysis supports this point.

*Department of Applied Mathematics, University of Waterloo, Waterloo, N2L 3G1 CANADA, kmorris@uwaterloo.ca
†ETIS Lab - CNRS UMR 8051, 6 avenue du Ponceau, 95014 Cergy-Pontoise, FRANCE, navasca@ensea.fr
Introduction

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

$$\int_0^\infty \langle Cz(t), Cz(t) \rangle + \langle u(t), Ru(t) \rangle dt$$

(1)

is minimized where $R$ is a symmetric positive definite matrix weighting the control, $C$ is

a matrix weighting the state, and $z(t)$ is determined by the system of ordinary differential
equations

$$\dot{z}(t) = Az(t) + Bu(t).$$

It is well-known that the solution to this problem is found by finding the positive semi-definite
solution $P$ to the algebraic Riccati equation [27][e.g.]

$$A^*P + PA - PBR^{-1}B^*P = -C^*C.$$  

(2)

The optimal control is then given by $u(t) = -Kx(t)$ where

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

(3)

If the system is modeled 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; see

[10, 14, 22, 23].

In practice, the control is calculated through approximation. The matrices $A$ and $B$ arise

in a finite dimensional approximation of the infinite dimensional system. Letting $n$ indicate

the order of the approximation and $m$ the number of control inputs; $A$ is an $n \times n$ matrix

and $B$ is a $n \times m$ matrix. 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, see for instance, [4, 15, 18, 22, 23]. 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.

In many cases, particularly in partial differential equation models with more than one

space dimension, approximations lead to Riccati equations of very large order that must be

solved. Standard direct methods are based on eigenvector calculation. The main methods

are Potter’s method [33] and the Schur vector-based technique [21]. These algorithms require

computational effort proportional to $n^3$ where $n$ is the order of the system; thus they are

impractical for large order systems. A survey of currently used methods to solve large

AREs can be found in [7]. Two common iterative methods are Chandrasekhar [3, 8] and

Newton-Kleinman iterations [17, 20, 34]. In this paper, we use a modification of the Newton-

Kleinman method first proposed by Banks and Ito [3] as a refinement for a partial solution

to the Chandrasekhar equation. In [3] the Chandrasekhar equations are partially solved to

obtain a stabilizing initial feedback $K_o$ as a stabilizing initial guess to the modified Newton-

Kleinman method. The modified Newton-Kleinman is subsequently used and compared to

a Schur method.

Solution of the Lyapunov equation is a key step in implementing either modified or

standard Newton-Kleinman. For most problems, the matrices involved are sparse. Also,
in general, the number of controls, \( m \), is considerably less than the number of states, \( n \). We implement a method, Cholesky-ADI (CF-ADI) [24, 31], that takes advantage of these features. The use of this Lyapunov solver with a standard Newton-Kleinman method is described in [6, 7, 29, 32].

In the next section both standard and modified Newton-Kleinman are described. Then in Section 2 we compare both methods. We show that the modified Newton-Kleinman method is unconditionally convergent. We show that a modified Newton-Kleinman method is considerably faster than standard Newton-Kleinman when both are coupled with this Lyapunov solver. We then solve a number of standard control examples, including one with several space variables. Our results indicate that the modified Newton-Kleinman method achieves considerable savings in computation time over standard Newton-Kleinman.

Numerical results on a beam problem indicate that convergence is decreased with decreasing damping. In Section 3, we analyze the behavior of low rank approximations for damped systems. Numerical results and theoretical analysis indicate that in some cases a low rank approximation to the Riccati equation solution may not exist for a lightly damped system. Finally, in the last section, we make some concluding remarks.

1 Description of Algorithm

One approach to solving large Riccati equations is the Newton-Kleinman method [20]. The Riccati equation (2) can be rewritten as

\[
(A - BK)^*P + P(A - BK) = -Q - K^*RK
\]

where \( Q = C^*C \) and \( K = -R^{-1}B^*P \). We say a matrix \( A_o \) is Hurwitz if \( \sigma(A_o) \subseteq \mathbb{C}_- \). If \( A - BK \) is Hurwitz, then there exists \( P \geq 0 \) satisfying the Lyapunov equation above.

**Theorem 1.1** [20] Consider a stabilizable pair \((A,B)\) with a feedback \( K_0 \) so that \( A - BK_0 \) is Hurwitz. Then the sequence of solutions \( P_i \) to the Lyapunov equation

\[
S_i^* P_i + P_i S_i = -Q - K_i^*R K_i
\]

where \( S_i = A - BK_i \) and \( K_{i+1} = -R^{-1}B^*P_i \) has

\[
\lim_{i \to \infty} P_i = P
\]

with a quadratic convergence rate.

The key assumption in the above theorem is that there exists an initial estimate \( K_0 \) such that \( A - BK_0 \) is Hurwitz. This condition is not restrictive for Riccati equations arising in control of infinite-dimensional systems. Many of these systems are stable even when uncontrolled and so the initial iterate \( K_0 \) may be chosen as zero. If the original system is not stable, the initial stabilizing iterate can be chosen using a number of approaches. For example, 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 may be used as an ansatz for a higher order approximation. This technique was used successfully in [17, 29, 34].

A modification to the Newton-Kleinman method was proposed by Banks and Ito [3].
Theorem 1.2 Consider equation (4) and choose $K_0$ and $K_1$ so that $A - BK_0$ and $A - BK_1$ are Hurwitz. Then the following Lyapunov equation and feedback update,

$$(A - BK_i)X_i + X_i(A - BK_i) = -[K_i - K_{i-1}]^* R [K_i - K_{i-1}], \quad (6)$$

$$K_{i+1} = K_i - B^* X_i.$$  

has feedback gain $K_i$ where

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

with quadratic convergence.

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

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

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

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

The solution of a Lyapunov equation is a key step in both standard and modified Newton-Kleinman iterations. For $A_o \in \mathbb{R}^{n \times n}$ is Hurwitz and $D \in \mathbb{R}^{n \times r}$ consider the Lyapunov equation

$$XA_o + A_o^* X = -DD^* \quad (9)$$

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 [5] are only appropriate for low model order and do not take advantage of sparsity in the matrices. The most basic iterative method to solve Lyapunov equation is Smith’s method [36]. 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, the solution $X$ is found by using successive substitutions: $X = \lim_{i \to \infty} X_i$ where

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

with $X_0 = 0$. The convergence of the iterations can be improved by careful choice of the parameter $p$; see [35, pg. 297]. This method of successive substitution is unconditionally convergent, but the convergence rate is linear convergence.

The ADI method [25, 38] improves Smith’s method by using a different parameter $p_i$ at each step. Two alternating linear systems,

$$(A_o^* + p_i I)X_{i-\frac{1}{2}} = -DD^* - X_{i-1}(A_o - p_i I) \quad (11)$$

$$(A_o^* + p_i I)X_i^* = -DD^* - X_{i-\frac{1}{2}}^*(A_o - p_i I) \quad (12)$$
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 \([12, 25, 37]\). For symmetric \( A \) the optimal parameters can be easily calculated. We estimate the complex ADI parameters as in \([12]\) if the spectrum of \( A \) is complex. As the spectrum of \( A \) flattens to the real axis the ADI parameters are closer to optimal.

If \( A_0 \) 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 \( X \) \([24, 31]\). Substituting \( Z_iZ_i^* \) for \( X_i \) and setting \( X_0 = 0 \), we obtain the following iterates for \( Z_i \):

\[
Z_1 = \sqrt{-2p_1(A_o^* + p_1I)^{-1}D} \\
Z_i = [\sqrt{-2p_i(A_o^* + p_iI)^{-1}D}, (A_o^* + p_iI)^{-1}(A_o^* - p_iI)Z_{i-1}].
\]

(13)

Note that \( Z_1 \in \mathbb{R}^{n \times r} \), \( Z_2 \in \mathbb{R}^{n \times 2r} \), and \( Z_i \in \mathbb{R}^{n \times ir} \). At each iteration, an approximation to the solution of increasing rank is obtained. The algorithm is stopped when the Cholesky iterations converge within some tolerance. In \([24]\) 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. The Cholesky factor \( Z \) is used to calculate the feedback \( K \) - the full solution \( X \) is never constructed. Also, the complexity of CF-ADI is also considerably less than that of standard ADI as shown in Table 1. For both standard and modified Newton-Kleinman iterations, \( A_o = A - BK_i \). The factor \( D \) differs significantly. Let \( n_Q \) indicate the rank of \( Q = C^*C \). In the case of standard Newton-Kleinman,

\[
D = [C^* K_i^* R^{1/2}],
\]

while for modified Newton-Kleinman,

\[
D = (K_i - K_{i-1})^* R^{1/2}.
\]

In the case of standard Newton-Kleinman, the rank \( r \) of the right-hand-side \( r = m + n_Q \) while for modified Newton-Kleinman, \( r = m \). Since \( n_Q \) is close to \( n \) in many cases, while \( m \) is generally much smaller than \( n \), this means that the rank of the right-hand-side is typically much smaller for modified Newton-Kleinman than for the standard algorithm. The reduction in complexity of CF-ADI method over ADI is marked for the modified Newton-Kleinman method.

\section{Comparison of Standard and Modified Newton-Kleinman}

In this section we compare the standard and modified Newton-Kleinman methods on a number of standard examples. The computations were done in MatLab on a personal computer
Table 1: Complexity of CF-ADI and ADI [24]

<table>
<thead>
<tr>
<th></th>
<th>CF-ADI</th>
<th>ADI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sparse A</td>
<td>$\mathcal{O}(Jr n)$</td>
<td>$\mathcal{O}(Jr n^2)$</td>
</tr>
<tr>
<td>Full A</td>
<td>$\mathcal{O}(Jr n^2)$</td>
<td>$\mathcal{O}(Jr n^3)$</td>
</tr>
</tbody>
</table>

Table 2: Cholesky-ADI Method

Given $A_o$ and $D$
Choose ADI parameters $\{p_1, \ldots, p_J\}$ with $\Re(p_i) < 0$
Define $z_1 = \sqrt{-2p_1(A_o^* + p_1I)^{-1}}D$
and $Z_1 = [z_1]$
For $i = 2, \ldots, J$
Define $W_i = (\sqrt{-2p_{i+1}})[I - (p_{i+1} - p_i)(A_o^* + p_{i+1}I)^{-1}]$
(1) $z_i = W_i z_{i-1}$
(2) If $\|z\| > tol$
$Z_i = [Z_{i-1}, z_i]$
Else, stop.

with two 2.5 GHz AMD Opteron processors with 1024KB cache memory and 2G RAM. The relative error of the feedback $K$ is measured and used as a stopping criterion, with a tolerance of $10^{-9}$ and the stopping criterion for the Lyapunov solver is set to $10^{-9}$ for the heat equation and $10^{-12}$ for the beam problem. This criterion was used because it is cheap to calculate and also for most applications, an accurate calculation of $K$, not $P$ is required. The residuals in the ARE were also calculated as a check on the algorithm and are shown in the tables. The total Lyapunov iterations includes the iterations from the required initializations.

One-dimensional Heat Equation

Consider the linear quadratic regulator problem of minimizing a cost functional (1) subject to [3, 9]

$$\frac{\partial z(t, x)}{\partial t} = \frac{\partial^2 z(t, x)}{\partial x^2}, \quad x \in (0, 1),$$

$$z(0, x) = \psi(x),$$

(14)

with boundary conditions

$$\frac{\partial z(t, 0)}{\partial x} = u(t)$$
$$\frac{\partial z(t, 1)}{\partial x} = 0.$$  

(15)
Table 3: 1-d Heat Equation with single observation: Standard Newton-Kleinman

<table>
<thead>
<tr>
<th>n</th>
<th>N.K. Itn’s</th>
<th>Lyapunov Itn’s</th>
<th>time (s)</th>
<th>residual</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>12</td>
<td>348</td>
<td>.15</td>
<td>$7.7 \times 10^{-9}$</td>
</tr>
<tr>
<td>50</td>
<td>12</td>
<td>406</td>
<td>.23</td>
<td>$1.1 \times 10^{-8}$</td>
</tr>
<tr>
<td>100</td>
<td>12</td>
<td>464</td>
<td>.65</td>
<td>$7.0 \times 10^{-9}$</td>
</tr>
<tr>
<td>200</td>
<td>12</td>
<td>522</td>
<td>4.55</td>
<td>$8.9 \times 10^{-9}$</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>n</th>
<th>N.K. Itn’s</th>
<th>Lyapunov Itn’s</th>
<th>time (s)</th>
<th>residual</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>10</td>
<td>254</td>
<td>.11</td>
<td>$7.7 \times 10^{-9}$</td>
</tr>
<tr>
<td>50</td>
<td>10</td>
<td>300</td>
<td>.17</td>
<td>$1.1 \times 10^{-8}$</td>
</tr>
<tr>
<td>100</td>
<td>10</td>
<td>306</td>
<td>.43</td>
<td>$7.0 \times 10^{-9}$</td>
</tr>
<tr>
<td>200</td>
<td>9</td>
<td>347</td>
<td>2.49</td>
<td>$8.9 \times 10^{-9}$</td>
</tr>
</tbody>
</table>

We choose $R = 1$ and $C$ is defined by

$$Cz(t) = \int_{0}^{1} z(t, x)dx.$$  \hfill (16)

The solution to the infinite-dimensional Riccati equation is

$$Kz = \int_{0}^{1} k(x)z(x)dx$$

where $k \equiv 1$ [4]. 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 - 6 we compare the number of Newton-Kleinman and Lyapunov iterations as well as the CPU time per order $n$ for $C$ as in (16) and also $C = I$. We use the ansatz $k_0(x) = 100$ in all cases. With the modified algorithm, there are fewer Riccati loops than with 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. As predicted by the complexity analysis, the reduction the in number of iterations and computation time of the modified Newton-Kleinman method over that required in the standard method is greater with $C = I$ than for the case of a single observation.
Table 5: 1-d Heat Equation with C=I: Standard Newton-Kleinman

<table>
<thead>
<tr>
<th>n</th>
<th>N.K. Itn’s</th>
<th>Lyapunov Itn’s</th>
<th>time (s)</th>
<th>residual</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>12</td>
<td>363</td>
<td>.22</td>
<td>$7.7 \times 10^{-9}$</td>
</tr>
<tr>
<td>50</td>
<td>12</td>
<td>418</td>
<td>.83</td>
<td>$1.1 \times 10^{-8}$</td>
</tr>
<tr>
<td>100</td>
<td>12</td>
<td>480</td>
<td>4.10</td>
<td>$7.0 \times 10^{-9}$</td>
</tr>
<tr>
<td>200</td>
<td>12</td>
<td>535</td>
<td>28.35</td>
<td>$8.9 \times 10^{-9}$</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>n</th>
<th>N.K. Itn’s</th>
<th>Lyapunov Itn’s</th>
<th>time (s)</th>
<th>residual</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>10</td>
<td>261</td>
<td>.12</td>
<td>$7.7 \times 10^{-9}$</td>
</tr>
<tr>
<td>50</td>
<td>10</td>
<td>306</td>
<td>.21</td>
<td>$1.1 \times 10^{-8}$</td>
</tr>
<tr>
<td>100</td>
<td>10</td>
<td>314</td>
<td>.68</td>
<td>$7.0 \times 10^{-9}$</td>
</tr>
<tr>
<td>200</td>
<td>9</td>
<td>354</td>
<td>4.02</td>
<td>$8.9 \times 10^{-9}$</td>
</tr>
</tbody>
</table>

Two-Dimensional Heating Problem

Define the rectangle $\Omega = [0, 1] \times [0, 1]$ with boundary $\partial \Omega$. Consider the two-dimensional partial differential equation [9]

$$
\frac{\partial z}{\partial t}(x, y, t) = \frac{\partial^2 z}{\partial x^2} + \frac{\partial^2 z}{\partial y^2} + 20\frac{\partial z}{\partial y} + 100z = f(x, y)u(t), \quad (x, y) \in \Omega
$$

$$(x, y) \in \partial \Omega$$

(17)

where $z$ is a function of $x, y$ and $t$ and

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

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

We solved the Riccati equation on a number of grids, using both standard and modified Newton-Kleinman methods with $C = B^*$ and $R = 1$. The data is shown in Tables 7 and 8. We also investigated the performance of the algorithm with the same numerical approximation of the partial differential equation (17) but with control weight $R = 100$ and cost $C = I$. The results are shown in Tables 9 and 10. Fewer Lyapunov iterations are required for modified Newton-Kleinman than for standard Newton-Kleinman and there is also 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 has large rank.
Table 7: 2-d Heat Equation with single observation: Standard Newton-Kleinman

<table>
<thead>
<tr>
<th>grid</th>
<th>n</th>
<th>N.K. Itn’s</th>
<th>Lyapunov Itn’s</th>
<th>time (s)</th>
<th>residual</th>
</tr>
</thead>
<tbody>
<tr>
<td>12 × 12</td>
<td>144</td>
<td>13</td>
<td>431</td>
<td>1.7</td>
<td>7.9 × 10⁻⁷</td>
</tr>
<tr>
<td>23 × 12</td>
<td>276</td>
<td>14</td>
<td>507</td>
<td>5.84</td>
<td>6.2 × 10⁻⁷</td>
</tr>
<tr>
<td>23 × 23</td>
<td>529</td>
<td>14</td>
<td>557</td>
<td>29.18</td>
<td>9.5 × 10⁻⁸</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>grid</th>
<th>n</th>
<th>N.K. Itn’s</th>
<th>Lyapunov Itn’s</th>
<th>time (s)</th>
<th>residual</th>
</tr>
</thead>
<tbody>
<tr>
<td>12 × 12</td>
<td>144</td>
<td>12</td>
<td>378</td>
<td>1.3</td>
<td>7.9 × 10⁻⁷</td>
</tr>
<tr>
<td>23 × 12</td>
<td>276</td>
<td>13</td>
<td>447</td>
<td>4.17</td>
<td>6.2 × 10⁻⁷</td>
</tr>
<tr>
<td>23 × 23</td>
<td>529</td>
<td>13</td>
<td>524</td>
<td>15.15</td>
<td>9.5 × 10⁻⁸</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>grid</th>
<th>n</th>
<th>N.K. Itn’s</th>
<th>Lyapunov Itn’s</th>
<th>time (s)</th>
<th>residual</th>
</tr>
</thead>
<tbody>
<tr>
<td>12 × 12</td>
<td>144</td>
<td>6</td>
<td>96</td>
<td>3.3</td>
<td>8.5 × 10⁻⁸</td>
</tr>
<tr>
<td>23 × 12</td>
<td>276</td>
<td>6</td>
<td>124</td>
<td>25.63</td>
<td>1.0 × 10⁻⁴</td>
</tr>
<tr>
<td>23 × 23</td>
<td>529</td>
<td>7</td>
<td>166</td>
<td>237.21</td>
<td>4.0 × 10⁻⁴</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>grid</th>
<th>n</th>
<th>N.K. Itn’s</th>
<th>Lyapunov Itn’s</th>
<th>time (s)</th>
<th>residual</th>
</tr>
</thead>
<tbody>
<tr>
<td>12 × 12</td>
<td>144</td>
<td>5</td>
<td>68</td>
<td>.59</td>
<td>8.4 × 10⁻⁵</td>
</tr>
<tr>
<td>23 × 12</td>
<td>276</td>
<td>5</td>
<td>88</td>
<td>3.5</td>
<td>1.1 × 10⁻⁴</td>
</tr>
<tr>
<td>23 × 23</td>
<td>529</td>
<td>6</td>
<td>103</td>
<td>27.4</td>
<td>4.0 × 10⁻⁴</td>
</tr>
</tbody>
</table>
Euler-Bernoulli Beam

Consider an 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 torque \(u(t)\) at the clamped end \((r = 0)\). The partial differential equation model with Kelvin-Voigt and viscous damping is

\[
\rho w_{tt}(r, t) + C_v w_t(r, t) + \frac{\partial^2}{\partial r^2} [C_d I_b w_{rr}(x, t) + EI_b w_{rr}(r, t)] = \frac{\rho r}{I_h} u(t),
\]

with boundary conditions

\[
\begin{align*}
    w(0, t) &= 0, \\
    w_r(1, t) &= 0, \\
    EI w_{rr}(1, t) + C_d I_b w_{rr}(1, t) &= 0, \\
    \frac{\partial}{\partial r} [EI(1) w_{rr}(r, t) + C_d I_b w_{rr}(r, t)]_{r=1} &= 0.
\end{align*}
\]

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

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

\[
A = \begin{bmatrix}
0 & I \\
-\frac{EI d^4}{\rho} & -\frac{C_d I d^4}{\rho} & -\frac{C_v}{\rho}
\end{bmatrix}, \quad B = \begin{bmatrix}
0 \\
r \\
\frac{I_h}{l_h}
\end{bmatrix},
\]

with domain

\[
\text{dom} (A) = \{(\phi, \psi) \in X : \psi \in H \text{ and } M = EI \frac{d^2}{dr^2} \phi + C_d I \frac{d^2}{dr^2} \psi \in H^2(0, 1) \text{ with } M(L) = 0 \}.
\]

Set \(R = 1\) and define \(C\) by the tip position

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

We also solved the control problem with \(C = I\), and \(R = 1\).

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 \(H^{2N} \times H^{2N}\) [19, e.g.] of dimension \(n = 4N\). This approximation method
Table 11: Beam physical parameters

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>E</td>
<td>$2.68 \times 10^{10}$ N/m²</td>
</tr>
<tr>
<td>$I_b$</td>
<td>$1.64 \times 10^{-9}$ m⁴</td>
</tr>
<tr>
<td>$\rho$</td>
<td>$1.02087$ kg/m</td>
</tr>
<tr>
<td>$C_v$</td>
<td>$2$ Ns/m²</td>
</tr>
<tr>
<td>$C_d$</td>
<td>$5 \times 10^8$ Ns/m²</td>
</tr>
<tr>
<td>L</td>
<td>$1$ m</td>
</tr>
<tr>
<td>$I_h$</td>
<td>$121.9748$ kg m²</td>
</tr>
<tr>
<td>d</td>
<td>$.041$ kg⁻¹</td>
</tr>
</tbody>
</table>

Table 12: Beam with C=Ctip: Standard Newton-Kleinman

<table>
<thead>
<tr>
<th>$K_0$</th>
<th>n</th>
<th>N.K. Itn’s</th>
<th>Lyapunov Itn’s</th>
<th>time (s)</th>
<th>residual</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>48</td>
<td>1</td>
<td>558</td>
<td>.3</td>
<td>$1.3 \times 10^{-11}$</td>
</tr>
<tr>
<td>0</td>
<td>72</td>
<td>1</td>
<td>620</td>
<td>.7</td>
<td>$1.3 \times 10^{-11}$</td>
</tr>
<tr>
<td>0</td>
<td>96</td>
<td>1</td>
<td>664</td>
<td>1.21</td>
<td>$1.5 \times 10^{-11}$</td>
</tr>
<tr>
<td>0</td>
<td>120</td>
<td>1</td>
<td>698</td>
<td>1.99</td>
<td>$3.4 \times 10^{-11}$</td>
</tr>
<tr>
<td>$B'$</td>
<td>48</td>
<td>1</td>
<td>558</td>
<td>.37</td>
<td>$1.3 \times 10^{-11}$</td>
</tr>
<tr>
<td>$B'$</td>
<td>72</td>
<td>1</td>
<td>620</td>
<td>.72</td>
<td>$1.3 \times 10^{-11}$</td>
</tr>
<tr>
<td>$B'$</td>
<td>96</td>
<td>1</td>
<td>664</td>
<td>1.2</td>
<td>$1.4 \times 10^{-11}$</td>
</tr>
<tr>
<td>$B'$</td>
<td>120</td>
<td>1</td>
<td>698</td>
<td>2.0</td>
<td>$3.5 \times 10^{-11}$</td>
</tr>
</tbody>
</table>

yields a sequence of solutions to the algebraic Riccati equation that converges strongly to
the solution to the infinite-dimensiononal Riccati equation corresponding to the original partial
differential equation description [4, 26].

The control problem was solved with several different choices of ansatz $K_0$: 0 and $B^*$. The
results are shown in Tables 12-15. In all cases, modified Newton-Kleinman converged with
a significant reduction in iterations and computation time over standard Newton-Kleinman.
As for the other examples, the reduction in the computation time is greater for full state
observation than for a single observation.

3 Low rank approximations to Lyapunov function

For the parabolic equations in Examples 1 and 2 the number of Lyapunov iterations required
in each Newton-Kleinman iteration was quite low- less than 50 in all cases. The convergence
was much slower for the beam equation. Although for all n, only 1 or 2 Newton-Kleinman
iterations were required, the number of Lyapunov iterates increases as n increases. Tables
16,17 show the effect of varying $C_d$ on the number of iterations required for convergence.
Larger values of $C_d$ led to a decreasing number of iterations. Small values of $C_d$ led to a
large number of required iterations in each solution of a Lyapunov equation.
Table 13: Beam with C=Ctip: Modified Newton-Kleinman

<table>
<thead>
<tr>
<th>$K_0$</th>
<th>n</th>
<th>N.K. Itn’s</th>
<th>Lyapunov Itn’s</th>
<th>time (s)</th>
<th>residual</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>48</td>
<td>1</td>
<td>280</td>
<td>.34</td>
<td>$1.3 \times 10^{-11}$</td>
</tr>
<tr>
<td>0</td>
<td>72</td>
<td>1</td>
<td>311</td>
<td>.47</td>
<td>$1.3 \times 10^{-11}$</td>
</tr>
<tr>
<td>0</td>
<td>96</td>
<td>1</td>
<td>333</td>
<td>.74</td>
<td>$1.5 \times 10^{-11}$</td>
</tr>
<tr>
<td>0</td>
<td>120</td>
<td>1</td>
<td>350</td>
<td>1.14</td>
<td>$3.4 \times 10^{-11}$</td>
</tr>
<tr>
<td>B'</td>
<td>48</td>
<td>2</td>
<td>547</td>
<td>.46</td>
<td>$1.3 \times 10^{-11}$</td>
</tr>
<tr>
<td>B'</td>
<td>72</td>
<td>2</td>
<td>312</td>
<td>.60</td>
<td>$1.3 \times 10^{-11}$</td>
</tr>
<tr>
<td>B'</td>
<td>96</td>
<td>1</td>
<td>333</td>
<td>.75</td>
<td>$1.4 \times 10^{-11}$</td>
</tr>
<tr>
<td>B'</td>
<td>120</td>
<td>1</td>
<td>350</td>
<td>1.17</td>
<td>$3.5 \times 10^{-11}$</td>
</tr>
</tbody>
</table>

Table 14: Beam with C=I: Standard Newton-Kleinman

<table>
<thead>
<tr>
<th>$K_0$</th>
<th>n</th>
<th>N.K. Itn’s</th>
<th>Lyapunov Itn’s</th>
<th>time (s)</th>
<th>residual</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>48</td>
<td>1</td>
<td>566</td>
<td>3.49</td>
<td>$3.7 \times 10^{-10}$</td>
</tr>
<tr>
<td>0</td>
<td>72</td>
<td>1</td>
<td>628</td>
<td>9.15</td>
<td>$2.7 \times 10^{-9}$</td>
</tr>
<tr>
<td>0</td>
<td>96</td>
<td>1</td>
<td>672</td>
<td>18.41</td>
<td>$3.0 \times 10^{-9}$</td>
</tr>
<tr>
<td>0</td>
<td>120</td>
<td>1</td>
<td>706</td>
<td>32.48</td>
<td>$6.9 \times 10^{-9}$</td>
</tr>
<tr>
<td>B'</td>
<td>48</td>
<td>1</td>
<td>566</td>
<td>3.47</td>
<td>$3.6 \times 10^{-10}$</td>
</tr>
<tr>
<td>B'</td>
<td>72</td>
<td>1</td>
<td>628</td>
<td>9.15</td>
<td>$2.7 \times 10^{-9}$</td>
</tr>
<tr>
<td>B'</td>
<td>96</td>
<td>1</td>
<td>672</td>
<td>18.40</td>
<td>$2.4 \times 10^{-9}$</td>
</tr>
<tr>
<td>B'</td>
<td>120</td>
<td>1</td>
<td>706</td>
<td>32.41</td>
<td>$7.4 \times 10^{-9}$</td>
</tr>
</tbody>
</table>

Table 15: Beam with C=I: Modified Newton-Kleinman

<table>
<thead>
<tr>
<th>$K_0$</th>
<th>n</th>
<th>N.K. Itn’s</th>
<th>Lyapunov Itn’s</th>
<th>time (s)</th>
<th>residual</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>48</td>
<td>1</td>
<td>284</td>
<td>1.87</td>
<td>$3.6 \times 10^{-10}$</td>
</tr>
<tr>
<td>0</td>
<td>72</td>
<td>1</td>
<td>315</td>
<td>4.82</td>
<td>$2.7 \times 10^{-9}$</td>
</tr>
<tr>
<td>0</td>
<td>96</td>
<td>1</td>
<td>337</td>
<td>9.59</td>
<td>$2.8 \times 10^{-9}$</td>
</tr>
<tr>
<td>0</td>
<td>120</td>
<td>1</td>
<td>354</td>
<td>16.88</td>
<td>$7.0 \times 10^{-9}$</td>
</tr>
<tr>
<td>B'</td>
<td>48</td>
<td>2</td>
<td>550</td>
<td>2.03</td>
<td>$3.6 \times 10^{-10}$</td>
</tr>
<tr>
<td>B'</td>
<td>72</td>
<td>1</td>
<td>315</td>
<td>4.82</td>
<td>$3.0 \times 10^{-9}$</td>
</tr>
<tr>
<td>B'</td>
<td>96</td>
<td>1</td>
<td>337</td>
<td>9.6</td>
<td>$2.3 \times 10^{-9}$</td>
</tr>
<tr>
<td>B'</td>
<td>120</td>
<td>1</td>
<td>354</td>
<td>16.85</td>
<td>$7.6 \times 10^{-9}$</td>
</tr>
</tbody>
</table>
Recall that the CF-ADI algorithm used here starts with a rank 1 initial estimate of the Cholesky factor and the rank of the solution is increased at each step. The Cholesky-ADI method relies on its efficiency on the existence of a low-rank approximation to the solution $X$ to the Lyapunov equation being solved. This is true of many other iterative algorithms to solve large Lyapunov equations.

**Theorem 3.1** [16, e.g. Thm. 3.5.2]. For any symmetric, positive semi-definite matrix $X$ of rank $n$ let $\lambda_1 \geq \lambda_2 \ldots \geq \lambda_n$ be its eigenvalues. For any rank $k < n$ matrix $X_k$,

$$\frac{\|X - X_k\|_2}{\|X\|_2} \geq \frac{\lambda_{k+1}}{\lambda_1}.$$ 

Thus, the relative size of the largest and smallest eigenvalues determines the existence of a low rank approximation $X_k$ that is close to $X$, regardless of how this approximation is obtained.

A bound on the rank of the solution to a Lyapunov equation where $A$ is symmetric is given in [30]. A tighter bound on the error in a low-rank approximation has been obtained [1] in terms of the eigenvalues and eigenvectors of $A$. This bound is applicable to all diagonalizable matrices $A$. The bound for the case where the right-hand-side $D$ has rank one is as follows.

**Theorem 3.2** [1, Thm 3.1] Let $V$ be the matrix of right eigenvectors of $A$, and denote its condition number by $\kappa(X)$. Denote the eigenvalues of $A$ by $\mu_1, \ldots, \mu_n$. There is a rank $k < n$ approximation to $X$ satisfying

$$\|X - X_k\|_2 \leq (n - k)^2 \delta_{k+1} (\kappa(V)\|D\|_2)^2$$

where

$$\delta_{k+1} = \frac{-1}{2\text{Real}(\mu_{k+1})} \Pi_{j=1}^{k} \left| \frac{\mu_{k+1} - \mu_j}{\mu_{k+1} + \mu_j} \right|^2.$$ 

The eigenvalues $\mu_i$ are ordered so that $\delta_i$ are decreasing.

If the condition number $\kappa(V)$ is not too large, for instance if $A$ is normal, then $\delta_{k+1}/\delta_1$ gives a relative error estimate of the error $\lambda_{k+1}/\lambda_1$ in the approximation $X_k$. This estimate is observed in [1] to estimate the actual decay rate of the eigenvalues of $X$ quite well, even in cases where the decay is very slow.

Consider a parabolic partial differential equation, where the $A$-matrix of the approximation is symmetric and negative definite and so all the eigenvalues $\mu_i$ of $A$ are real. A routine calculation yields that

$$\frac{\delta_{k+1}}{\delta_1} \approx \frac{\mu_1}{\mu_k}$$

and so the rate of growth of the eigenvalues determines the accuracy of the low rank approximation. Since the eigenvalues typically increase in magnitude quite quickly, the Lyapunov function for such problems can be approximated by a matrix of low rank.

The accuracy of the low-rank approximant can be quite different if a matrix $A$ is non-symmetric. The solution $X$ to the Lyapunov equation could be, for instance, the identity matrix [1, 30] in which case the eigenvalues of $X$ do not decay.
It is observed through several numerical examples in [1] that it is not just complex eigenvalues that cause the decay rate to slow down, but the dominance of the imaginary parts of the eigenvalues as compared to the real parts, together with absence of clustering in the spectrum of $A$.

Consider the bound (19) for the beam with $C_d = 0$, and an eigenfunction basis for the approximation. Defining $c_v = C_v/(\rho A)$, $c^2 = EI/(\rho A)$ and indicating the roots of $1 + \cos(a)cosh(a)$ by $a_k$, the eigenvalues of $A$ are [28]

$$\mu_k = -c_v/2 \pm i\omega_k$$

where

$$\omega_k = \sqrt{ca_k^2 - c_v^2/4}.$$ 

A simple calculation yields that for all $k$,

$$\frac{\delta_k}{\delta_1} = \prod_{j=1}^{k-1} \frac{1}{1 + \frac{4c_v^2}{(w_k-w_j)^2}} \approx 1.$$ 

This indicates that the eigenvalues of the Lyapunov solution do not decay. This effect was observed in the beam equation. The ratio $\lambda_{k+1}/\lambda_1$ is plotted for several values of $C_d$ in Figures 1 and 2. For larger values of $C_d$ the solution $X$ is closer to a low rank matrix than it is for smaller values of $C_d$.

These observations are supported by results in the theory of control of partial differential equations. If $B$ and $C$ are bounded finite-rank operators and $A$ generates an exponentially stable semigroup then the solution to the ARE is nuclear:

$$\sum_{i=1}^{\infty} \lambda_i(P) < \infty.$$ 

[11, 13]. Uniform approximation by low order matrices is possible. This does not provide an estimate about the rate of decay of the eigenvalues. Figure 1 shows eigenvalue ratios for a beam example in which the underlying semigroup is stable, and $C$ and $B$ have rank 1. In all cases the eigenvalues decay, although more slowly for a lightly damped system.

The results quoted above only apply if $B$ and $C$ are compact operators. If the original semigroup is analytic, for instance in the case of Examples 1, 2 and the beam equation with $C_d > 0$, then the approximations converge uniformly in operator norm under fairly standard assumptions; see [22]. However, if the semigroup is not analytic, the solution $P$ to the ARE may not be compact. For example, consider the class of systems where $A^* = -A$ (as in the undamped wave and beam equations). If $C = B^*$ then $P = I$ is a solution to the ARE [10]. Clearly, if $P$ is not compact, it cannot be uniformly approximated by a finite rank operator. Convergence of the approximating solutions to the ARE for an non-analytic semigroup, in general requires that $C$ must be compact; see [23]. In Figure 2 the observation $C$ is not compact and with $C_d = 0$ the underlying semigroup is not analytic; note that no decay in the eigenvalues occurs in this case. For problems where the underlying semigroup is not analytic the state weighting matrix $C$ must be compact. For lightly damped systems the decay of the eigenvalues in the Riccati solution may be very slow, even if the underlying
Figure 1: Eigenvalue ratio for solution to beam ARE for varying $C_d$ ($N = 96, C = C_{tip}$)

The analysis and results shown here indicate that a modified Newton-Kleinman method is considerably faster than standard Newton-Kleinman. The combination of the modified Newton-Kleinman method with the Cholesky-ADI method takes advantage of the fact that the number of controls is generally considerably less than the number of states. The reduction in number of iterations and computation time over standard Newton-Kleinman increases with the rank of the state-weighting matrix.

For the beam problem, increasing numbers of iterations were required as the damping was decreased. The convergence could probably be improved with better ADI parameter selection. However, as the damping is decreased, the ability to approximate the Riccati solution (or the solution to the Lyapunov equation) by a solution of low rank, is lost. Thus, one expects a greater number of iterations in the Lyapunov loop to be required as $C_d$ is decreased.

4 Conclusions
Figure 2: Eigenvalue ratio for solution to beam ARE for varying $C_d$ ($N = 96, C = I$)

Table 16: Beam: Effect of Changing $C_d$ (Standard Newton-Kleinman)

<table>
<thead>
<tr>
<th>$C_v$</th>
<th>$C_d$</th>
<th>$\alpha$</th>
<th>Newton-Kleinman Itn's</th>
<th>Lyapunov Itn's</th>
<th>CPU time</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>$1 \times 10^4$</td>
<td>1.5699</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td></td>
<td>$3 \times 10^5$</td>
<td>1.5661</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td></td>
<td>$4 \times 10^6$</td>
<td>1.5654</td>
<td>3</td>
<td>1620;1620;1620</td>
<td>63.14</td>
</tr>
<tr>
<td></td>
<td>$1 \times 10^7$</td>
<td>1.5370</td>
<td>3</td>
<td>1316;1316;1316</td>
<td>42.91</td>
</tr>
<tr>
<td></td>
<td>$1 \times 10^8$</td>
<td>1.4852</td>
<td>3</td>
<td>744;744;744</td>
<td>18.01</td>
</tr>
<tr>
<td></td>
<td>$5 \times 10^8$</td>
<td>1.3102</td>
<td>3</td>
<td>301;301;301</td>
<td>5.32</td>
</tr>
</tbody>
</table>
Table 17: Beam: Effect of Changing $C_d$ (Modified Newton-Kleinman)

<table>
<thead>
<tr>
<th>$C_V$</th>
<th>$C_d$</th>
<th>$\alpha$</th>
<th>Newton-Kleinman It’s</th>
<th>Lyapunov It’s</th>
<th>CPU time</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>$1 \times 10^4$</td>
<td>1.5699</td>
<td>2</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td>3</td>
<td>$3 \times 10^5$</td>
<td>1.5661</td>
<td>2</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td>4</td>
<td>$4 \times 10^5$</td>
<td>1.5654</td>
<td>2</td>
<td>1620:1</td>
<td>24.83</td>
</tr>
<tr>
<td>1</td>
<td>$1 \times 10^6$</td>
<td>1.5370</td>
<td>2</td>
<td>1316:1</td>
<td>16.79</td>
</tr>
<tr>
<td>1</td>
<td>$1 \times 10^8$</td>
<td>1.4852</td>
<td>2</td>
<td>744:1</td>
<td>7.49</td>
</tr>
<tr>
<td>5</td>
<td>$5 \times 10^8$</td>
<td>1.3102</td>
<td>2</td>
<td>301:1</td>
<td>2.32</td>
</tr>
</tbody>
</table>

Figure 3: Beam: Spectrum of A for $C_d = 1 \times 10^4$, $4 \times 10^5$, $1 \times 10^8$. 
$\alpha = 1.5699, 1.5654, 1.4852$.
Analysis indicates that the fundamental reason for the slow convergence with small $C_d$ is the “hyperbolic-like” behaviour of the problem. This may have consequences for control of systems such as plates and coupled acoustic-structures where the spectra have large imaginary parts relative to the real parts. Special methods may be needed to calculate controllers for these systems.

Acknowledgements This research was supported by the National Science and Engineering Research Council of Canada.

References


