Reduced Order Solver for High Rank Lyapunov Equations

Miroslav K Stoyanov

Department of Mathematics
Virginia Tech

Feb 21, 2009

We consider the high rank Lyapunov equation:

$$\mathbf{A}^T\mathbf{P} + \mathbf{PA} + \mathbf{Q} = 0,$$

where **A** is stable sparse matrix and **Q** is sparse and symmetric.

- The solution **P** is unique and symmetric positive definite.
- **P** is generally dense.
- $\mathbf{P} = \int_0^\infty \mathbf{e}^{\mathbf{A}^T t}\mathbf{Q}\mathbf{e}^{\mathbf{A}t}\mathbf{dt}$.

We do not look for **P**, but for **Pb**, where **b** is some given vector.

LQR Control Problem for PDEs, Riccati Equations

We need to solve series of Lyapunov equations of the form:

$$\mathbf{A}^T\mathbf{P} + \mathbf{P}\mathbf{A} + \mathbf{Q} = 0,$$

and

$$\mathbf{A}^T\mathbf{P} + \mathbf{P}\mathbf{A} + \mathbf{C}\mathbf{C}^T = 0.$$

We do not need **P** explicitly, but rather **Pb**.

Existing Methods

If $\mathbf{Q} = \mathbf{C}\mathbf{C}^T$, for some low rank $\mathbf{C}$, then

$$\mathbf{P} = \int_0^\infty \mathbf{e}^{\mathbf{A}^T t} \mathbf{C}\mathbf{C}^T \mathbf{e}^{\mathbf{A}t} dt.$$

- Convert the system to Discrete time by shifting $\mathbf{A}$ (Banks and Ito (1991), Benner (2007), Singler (2008)) or use model reduction technique to approximate $\mathbf{e}^{\mathbf{A}^T t}\mathbf{C}$.
- If $\mathbf{Q}$ is sparse, but with high rank, then we cannot factorize. Low rank/order approximation of $\mathbf{P}$ may not be possible.

Existing Methods

If $\mathbf{Q} = \mathbf{C}\mathbf{C}^T$, for some low rank $\mathbf{C}$, then

$$\mathbf{P} = \int_0^\infty \mathbf{e}^{\mathbf{A}^T t} \mathbf{C}\mathbf{C}^T \mathbf{e}^{\mathbf{A}t} dt.$$

- Convert the system to Discrete time by shifting $\mathbf{A}$ (Banks and Ito (1991), Benner (2007), Singler (2008)) or use model reduction technique to approximate $\mathbf{e}^{\mathbf{A}^T t}\mathbf{C}$.
- If $\mathbf{Q}$ is sparse, but with high rank, then we cannot factorize. Low rank/order approximation of $\mathbf{P}$ may not be possible.

Proper Orthogonal Decomposition (POD)

Goal is given an ODE system

$$\dot{x}(t) = F(x(t)), \qquad x(t) \in \mathbf{R^n},$$

produce another system

$$\dot{x}_r(t) = F_r(x_r(t)), \qquad x_r(t) \in \mathbf{R^r},$$

with $r << n$, so that $x_r(t)$ approximates $x(t)$.

Given the solution $x(t)$ over some time interval $[0, T]$ we look for

$$\max_{v \in \mathbf{R^n}} \int_0^T \langle v, x(t) \rangle^2 \, dt,$$

subject to

$$\|v\| = 1.$$

This results in an eigenvalue problem, the dominant eigenvalues span the space that contains the bulk of the energy of the system. If we arrange the dominant eigenvalues in the columns of the matrix $V \in \mathbf{R^{n \times r}}$, then:

$$\dot{x}_r(t) = V^T F(V x_r(t)).$$

$V x_r(t)$ approximates $x(t)$.

Test Problem

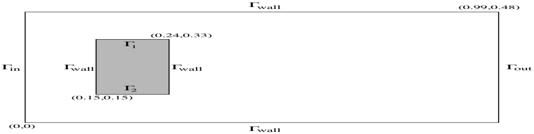

Figure: Domain

$$w_t + \left(\frac{1}{2}c_1 w^2\right)_\xi + \left(\frac{1}{2}c_2 w^2\right)_\eta = \mu\left(w_{\xi\xi} + w_{\eta\eta}\right)$$

Linearized Burgers equation, solving the Lyapunov equation:

$$A^T P + PA + I = 0.$$

$$A^T P + PA + I = 0$$

Apply POD to the Linearized Burgers equation, obtain a reduced system and solve the reduced Lyapunov equation:

$$V^T A^T VP + PV^T AV + I = 0.$$

Even if the reduced system captures the dynamics of the original one (i.e. take enough of the dominant vector), the error in the computed Lyapunov solution does not drop below 50%. In general "reduce then control" does not work, we need another approach.

Introduce the POD approximation later in the process.

$$A^T P + PA + Q = 0$$

- $P = \int_0^\infty e^{A^T s} Q e^{As}\, ds$
- $Pb = \int_0^\infty e^{A^T s} Q e^{As} b\, ds$
- $\dot{x}(t) = Ax(t)$, with $x(0) = b$
- $\lambda(t) = \int_t^\infty e^{A^T s} Q x(s)\, ds$
- $\dot{\lambda}(t) = -A^T \lambda(t) - Qx(t)$
- $\lambda(0) = Pb$

Can use $x(t)$ and $\lambda(t)$ to compute the solution.

Introduce the POD approximation later in the process.

$$A^T P + PA + Q = 0$$

- $P = \int_0^\infty e^{A^T s} Q e^{As} \, ds$
- $Pb = \int_0^\infty e^{A^T s} Q e^{As} b \, ds$
- $\dot{x}(t) = Ax(t)$, with $x(0) = b$
- $\lambda(t) = \int_t^\infty e^{A^T s} Q x(s) \, ds$
- $\dot{\lambda}(t) = -A^T \lambda(t) - Qx(t)$
- $\lambda(0) = Pb$

Can use $x(t)$ and $\lambda(t)$ to compute the solution.

Introduce the POD approximation later in the process.

$$A^T P + PA + Q = 0$$

- $P = \int_0^\infty e^{A^T s} Q e^{As} \, ds$
- $Pb = \int_0^\infty e^{A^T s} Q e^{As} b \, ds$
- $\dot{x}(t) = Ax(t)$, with $x(0) = b$
- $\lambda(t) = \int_t^\infty e^{A^T s} Q x(s) \, ds$
- $\dot{\lambda}(t) = -A^T \lambda(t) - Qx(t)$
- $\lambda(0) = Pb$

Can use $x(t)$ and $\lambda(t)$ to compute the solution.

Introduce the POD approximation later in the process.

$$A^T P + PA + Q = 0$$

- $P = \int_0^\infty e^{A^T s} Q e^{As}\, ds$
- $Pb = \int_0^\infty e^{A^T s} Q e^{As} b\, ds$
- $\dot{x}(t) = Ax(t)$, with $x(0) = b$
- $\lambda(t) = \int_t^\infty e^{A^T s} Q x(s)\, ds$
- $\dot{\lambda}(t) = -A^T \lambda(t) - Qx(t)$
- $\lambda(0) = Pb$

Can use $x(t)$ and $\lambda(t)$ to compute the solution.

Introduce the POD approximation later in the process.

$$A^T P + PA + Q = 0$$

- $P = \int_0^\infty e^{A^T s} Q e^{As} \, ds$
- $Pb = \int_0^\infty e^{A^T s} Q e^{As} b \, ds$
- $\dot{x}(t) = Ax(t)$, with $x(0) = b$
- $\lambda(t) = \int_t^\infty e^{A^T s} Q x(s) \, ds$
- $\dot{\lambda}(t) = -A^T \lambda(t) - Qx(t)$
- $\lambda(0) = Pb$

Can use $x(t)$ and $\lambda(t)$ to compute the solution.

Introduce the POD approximation later in the process.

$$A^T P + PA + Q = 0$$

- $P = \int_0^\infty e^{A^T s} Q e^{As} \, ds$
- $Pb = \int_0^\infty e^{A^T s} Q e^{As} b \, ds$
- $\dot{x}(t) = Ax(t)$, with $x(0) = b$
- $\lambda(t) = \int_t^\infty e^{A^T s} Q x(s) \, ds$
- $\dot{\lambda}(t) = -A^T \lambda(t) - Qx(t)$
- $\lambda(0) = Pb$

Can use $x(t)$ and $\lambda(t)$ to compute the solution.

Introduce the POD approximation later in the process.

$$A^T P + PA + Q = 0$$

- $P = \int_0^\infty e^{A^T s} Q e^{As} \, ds$
- $Pb = \int_0^\infty e^{A^T s} Q e^{As} b \, ds$
- $\dot{x}(t) = Ax(t)$, with $x(0) = b$
- $\lambda(t) = \int_t^\infty e^{A^T s} Q x(s) \, ds$
- $\dot{\lambda}(t) = -A^T \lambda(t) - Qx(t)$
- $\lambda(0) = Pb$

Can use $x(t)$ and $\lambda(t)$ to compute the solution.

One Approach

- Integrate $x(t)$ until some small or moderate T.
- T is large enough so that the remainder of the system can be approximated in dimensions $r << n$.
- Obtain a reduced order system for $x(t)$

$$\dot{x}_r(t) = A_r x_r(t),$$

  with $\int_T^\infty \|x(t) - Vx_r(t)\| dt$ small.
- Save the history for $x(t)$ over $[0, T]$ and $x_r(t)$ over $[T, T_\infty]$.
- Integrate $\lambda(t)$ backwards to zero.

This method saves storage and reduces integration cost, however, we still need to integrate $\lambda(t)$ over a very large time interval.

One Approach

- Integrate $x(t)$ until some small or moderate T.
- T is large enough so that the remainder of the system can be approximated in dimensions $r << n$.
- Obtain a reduced order system for $x(t)$

$$\dot{x}_r(t) = A_r x_r(t),$$

with $\int_T^\infty \|x(t) - Vx_r(t)\| dt$ small.

- Save the history for $x(t)$ over $[0, T]$ and $x_r(t)$ over $[T, T_\infty]$.
- Integrate $\lambda(t)$ backwards to zero.

This method saves storage and reduces integration cost, however, we still need to integrate $\lambda(t)$ over a very large time interval.

One Approach

- Integrate $x(t)$ until some small or moderate T.
- T is large enough so that the remainder of the system can be approximated in dimensions $r << n$.
- Obtain a reduced order system for $x(t)$

$$\dot{x}_r(t) = A_r x_r(t),$$

  with $\int_T^\infty \|x(t) - Vx_r(t)\| dt$ small.
- Save the history for $x(t)$ over $[0, T]$ and $x_r(t)$ over $[T, T_\infty]$.
- Integrate $\lambda(t)$ backwards to zero.

This method saves storage and reduces integration cost, however, we still need to integrate $\lambda(t)$ over a very large time interval.

One Approach

- Integrate $x(t)$ until some small or moderate T.
- T is large enough so that the remainder of the system can be approximated in dimensions $r << n$.
- Obtain a reduced order system for $x(t)$

$$\dot{x}_r(t) = A_r x_r(t),$$

with $\int_T^\infty \|x(t) - Vx_r(t)\| dt$ small.
- Save the history for $x(t)$ over $[0, T]$ and $x_r(t)$ over $[T, T_\infty]$.
- Integrate $\lambda(t)$ backwards to zero.

This method saves storage and reduces integration cost, however, we still need to integrate $\boldsymbol{\lambda}(t)$ over a very large time interval.

Reduced Order Solver for High Rank Lyapunov Equations

- Integrate $x(t)$ until T and save the history.
- Form the reduced order system $\dot{x}_r(t) = A_r x_r(t)$.
- Solve the mixed high and low order Sylvester equation

$$A^T S + S A_r + Q V = 0.$$

- Integrate $\dot{\lambda}(t) = -A^T \lambda(t) - Qx(t)$ over $[0, T]$ using

$$\lambda(T) = S x_r(T)$$

- $\lambda(0) \approx Pb$.

Table: Relative error E_{rel} for $\mu = 1/300$ examples

Mesh size $N = 768$.				
r	5	10	15	20
$T = 2$	1.7533e-05	2.7141e-09	4.5307e-13	3.1216e-13
$T = 4$	1.9402e-08	2.5989e-13	3.3649e-13	3.3688e-13
$T = 6$	1.9677e-11	3.3891e-13	3.3928e-13	3.3930e-13

Mesh size $N = 1413$.			
r	5	10	20
$T = 0$	4.2060e-01	2.2440e-02	7.2987e-06
$T = 2$	1.5882e-05	2.0714e-09	4.6295e-13
$T = 4$	1.2096e-08	4.6095e-13	4.6333e-13
$T = 6$	6.5992e-12	4.6319e-13	4.6317e-13

Conclusions

Efficient method for High-Rank Lyapunov equations.

- Exploit sparsity in the problem.
- Solve for the action of $\mathbf{P}$ onto a vector $\mathbf{b}$.
- Apply model reduction techniques.

Future Work

Theoretical error bound. Relate the model reduction error to the error in $\lambda(0)$.