Discrete-time Solutions to the Continuous-time Differential Lyapunov Equation With Applications to Kalman Filtering

Patrik Axelsson and Fredrik Gustafsson

Contribution

Kalman filtering based on a *continuous-time state space model* with *discrete-time measurements* requires a solver of a *continuous-time differential Lyapunov equation* (CDLE). This work analyzes **i)** stability, **ii)** computational complexity, and **iii)** numerical properties of three methods to solve the CDLE. A novel *low-complexity analytical solution* is proposed with significant better stability and numerical properties.

Background

- Kalman filtering based on a *continuous-time state space model* with *discrete-time measurements* involves a time update that *integrates the first and second order moments* from one sample time to the next one. The second order moment is a covariance matrix, and it governs a *continuous-time differential Lyapunov equation* (CDLE).
- Practitioners often tend to *discretize the state space model* to fit the discrete-time KF time update. That leads to well known *problems with accuracy and stability*, which can be managed by *oversampling the system*.

Linear Stochastic Differential Equations

For the *linear stochastic differential equation* (SDE)

$$dx(t) = Ax(t)dt + Gd\beta(t), \qquad E\left[d\beta(t)d\beta(\tau)^{\mathsf{T}}\right] = Qdt\delta(t-\tau)$$

the update of the first and second order moments, $\hat{x}(t)$ and $P(t)$ respectively, of the stochastic variable $x(t)$, are

$$\dot{\hat{x}}(t) = A\hat{x}(t), \qquad (*)$$

$$\dot{P}(t) = AP(t) + P(t)A^{\mathsf{T}} + \widetilde{Q}. \qquad (**) \qquad \left(\widetilde{Q} = GQG^{\mathsf{T}}\right)$$

Here, $(*)$ is an ordinary ODE, and $(**)$ a CDLE. Focus is on solving the CDLE. Three methods to solve the CDLE are:

1. **Exact solution:**

$$P(t) = e^{At}P(0)e^{A^{\mathsf{T}}t} + \underbrace{\int_0^t e^{A(t-s)}\widetilde{Q}e^{A^{\mathsf{T}}(t-s)}\,\mathrm{d}s}_{\triangleq Q_d(t)}$$

2. **Matrix fraction decomposition:** Let $P(t) = C(t)D(t)^{-1}$, where $C(t)$ and $D(t)$ are solution to

$$\frac{d}{dt}\begin{pmatrix} C(t) \\ D(t) \end{pmatrix} = \begin{pmatrix} A & \widetilde{Q} \\ 0 & -A^{\mathsf{T}} \end{pmatrix}\begin{pmatrix} C(t) \\ D(t) \end{pmatrix},$$

3. **Vectorization:**

$$\operatorname{vech}\dot{P}(t) = D^{\dagger}\left(I \otimes A + A \otimes I\right) D \operatorname{vech} P(t) + \operatorname{vech}\widetilde{Q}$$

The Matrix Exponential

One key approach for numerical calculation of the matrix exponential is oversampling and Taylor expansion,

$$e^{Ah} = \left(e^{Ah/m}\right)^m \approx \left(I + \left(\frac{Ah}{m}\right) + \dots + \frac{1}{p!}\left(\frac{Ah}{m}\right)^p\right)^m \triangleq e_{p,m}(Ah).$$

The Taylor expansion is a special case of the Padé approximation.

Analysis

Stability Analysis

The CDLE has a unique positive solution if A is Hurwitz, $\widetilde{Q} \succeq 0$, the pair $(A, \sqrt{\widetilde{Q}})$ is observable, and $P(0) \succ 0$. Does a continuous-time system satisfying these properties give a stable discrete-time recursion?

1. **Exact solution:** No stability problems.
2. **Matrix frac. decomp.:** The ODE has eigenvalues in $\pm\lambda_i$, hence the ODE is unstable. However, since $P(t) = C(t)D(t)^{-1}$ it can still give a correct solution.
3. **Vectorization:** No stability problems if the matrix exponential is solved exactly. Euler sampling, i.e., $e_{1,m}(A_P h)$, is stable if the sample time h satisfies

$$h < \min\left\{-\frac{2m\mathfrak{Re}\{\lambda_i + \lambda_j\}}{|\lambda_i + \lambda_j|^2}, 1 \le i \le j \le n_x\right\},$$

where λ_i, $i = 1, \dots, n$, are the eigenvalues to A.

Computational Complexity

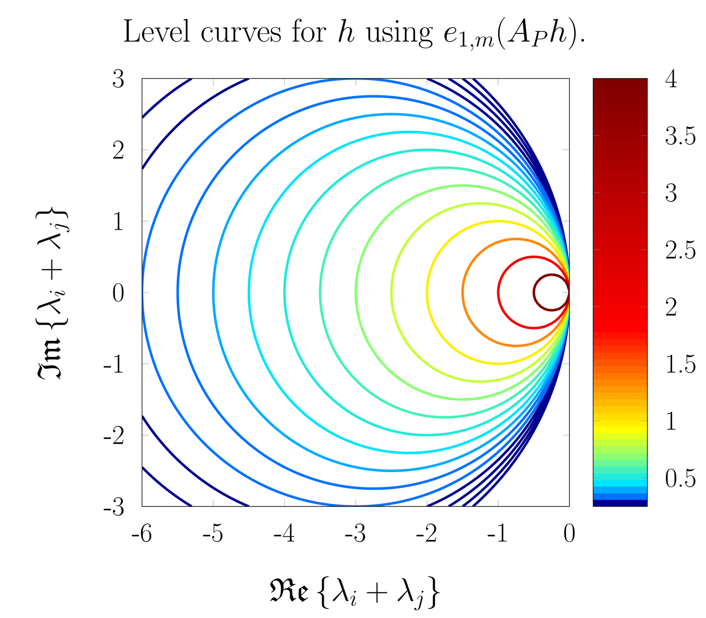
1. **Exact solution:** $(8(\log_2(m) + p) + 6)n_x^3$
2. **Matrix frac. decomp.:** $(8(\log_2(m) + p) + 12)n_x^3$
3. **Vectorization:** $\mathcal{O}(n_x^6)$
   - Rewritten solution: $(\log_2(m) + p + 43)\, n_x^3$, where $P(t)$ is given by

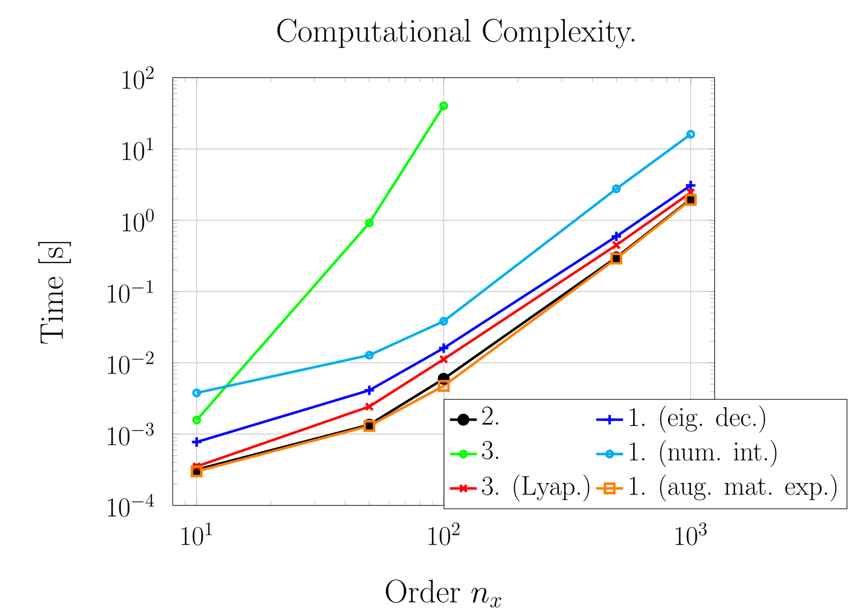
$$P(t) = e^{At}P(0)e^{A^{\mathsf{T}}t} + Q_d(t), \qquad AQ_d(t) + Q_d(t)A^{\mathsf{T}} + \widetilde{Q} - e^{At}\widetilde{Q}e^{A^{\mathsf{T}}t} = 0.$$

Numerical Properties (MC sim. with $A \in \mathbb{R}^{2\times 2}$ randomly chosen)

1) A large enough value of t should give that $P(t)$ equals the stationary solution given from the stationary Lyapunov equation $AP^{\text{stat}} + P^{\text{stat}}A^{\mathsf{T}} + \widetilde{Q} = 0$. **2)** The recursive updates should approach P^{stat} when $k \to \infty$.

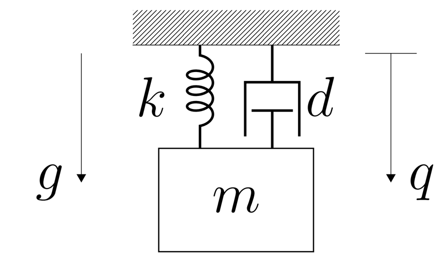
1. **Exact solution:** 1) $P(100) \neq P^{\text{stat}}$. 2) $\lim_{k\to\infty} P(t) = P^{\text{stat}}$
2. **Matrix frac. decomp.:** 1) $P(100) \neq P^{\text{stat}}$. 2) $\lim_{k\to\infty} P(t) \neq P^{\text{stat}}$
3. **Vectorization:** 1) $P(100) = P^{\text{stat}}$. 2) $\lim_{k\to\infty} P(t) = P^{\text{stat}}$



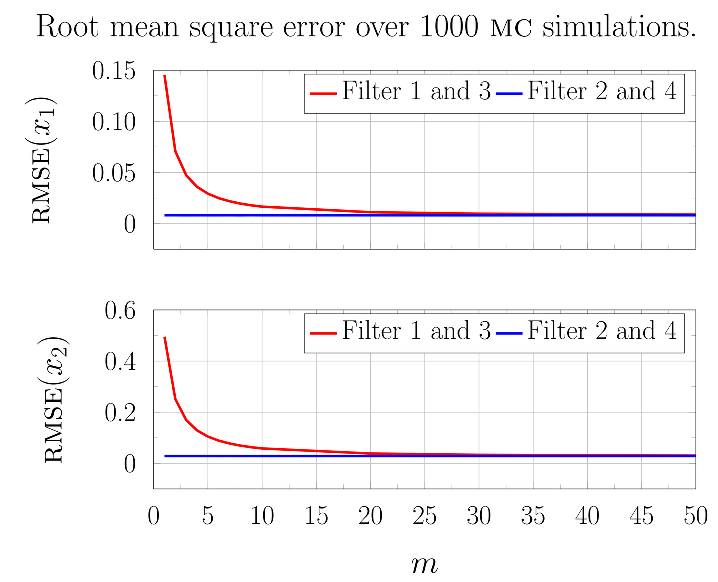


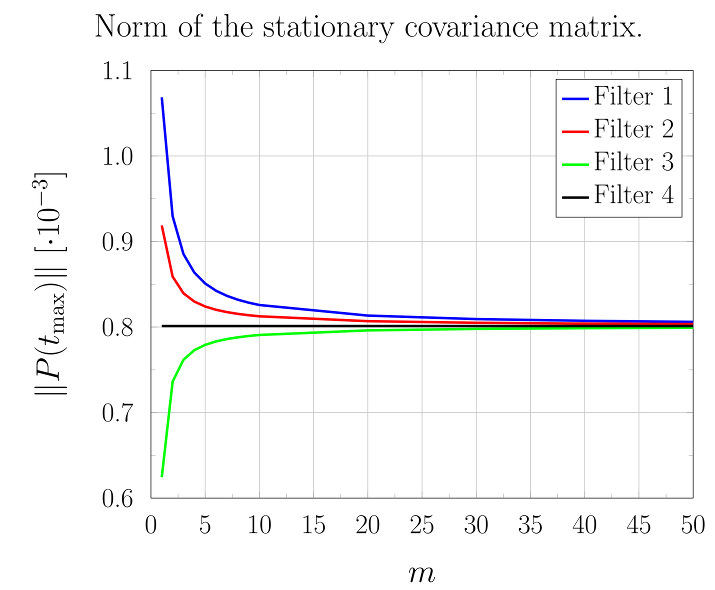
Linear Spring-damper Example

Continuous-time solutions of the SDE will be compared to discrete-time solutions where the model has been discretized. Four Kalman filters are used with $h = 0.09$ s.



1. $F_h = e_{1,m}(Ah)$ and $P(k+1) = F_h P(k) F_h^{\mathsf{T}} + G_h Q_h G_h^{\mathsf{T}}$,
2. F_h is given by `expm` in MATLAB and $P(k+1) = F_h P(k) F_h^{\mathsf{T}} + G_h Q_h G_h^{\mathsf{T}}$,
3. $F_h = e_{1,m}(Ah)$ and $P(k+1) = F_h P(k) F_h^{\mathsf{T}} + Q_d(h)$,
4. F_h is given by `expm` in MATLAB and $P(k+1) = F_h P(k) F_h^{\mathsf{T}} + Q_d(h)$,





- A factor of $m = 20$ or higher is required for the discretized methods.
- The execution time increases when m increases, hence the continuous-time solution is to prefer.
- The covariance matrix is important in e.g. target tracking, hence essential not to get too high or too low values.

Conclusions

- Kalman filtering is improved if the continuous-time update is solved directly instead of first discretizing the model.
- A novel solution to the CDLE was proposed and compared to existing methods.
- Extensions to nonlinear systems are also possible.