Discretisation of linear state-space processes with noise: Van Loan's variance formula

Antonio Sala

Video-presentación disponible en:

http://personales.upv.es/asala/YT/V/**vanloanEN**.html

Outline

Motivation:

Process with deterministic input in computer control have an exact ZOH discretization. What about noise input discretisation?

Objectives:

Understand Van Loan's formula to obtain the variance accumulation in one sampling period for a linear process. Only "expm" is needed, no integrals.

Contents:

Modelling. Mean/Variance equations. ZOH Discretization. Variance Discretization. Van Loan's formula. Conclusions.

Modelling

Let us have a linear process $\dot{x} = Ax + Bu + Fw$ being u a deterministic known input,x the state and w a normally distributet white noise, with variance parameter I (constant PSD).

The above equation is an "informal" representation of the Itó process $dx = (Ax + Bu)dt + Fd\mathcal{W}$ being $\mathcal{W}$ the standarised random walk (Wiener process).

*If $d\mathcal{W}$ had variance $H \cdot dt$ instead of $I \cdot dt, H > 0$, Choleski factorisation $H = QQ^T$ would convert it to $F' = FQ$ excited with white noise of unit PSD.

Simulation

If $d\mathcal{W}$ is Gaussian and so they are the initial conditions, then x is a Gaussian process, and with the knowledge of mean and variance, its time evolution can be fully characterised.

Mean equation:

$$\frac{d\bar{x}}{dt} = A\bar{x} + Bu$$

Variance equation:

$$\frac{dP}{dt} = AP + PA^T + W$$

being $W = FHF^T$ a known instantaneous variance parameter (power spectral density).

ZOH Discretization of the mean equation

As mean equation does not have noise, it's just plain ZOH discretisation, adding $\frac{du}{dt} = 0$. We end up with a discretised equation for the mean given by $\bar{x}_{k+1} = A_d(T)\bar{x}_k + B_d(T)u_k$, with

$$\begin{pmatrix} A_d(T) & B_d(T) \\ 0 & I \end{pmatrix} = exp\left[\begin{pmatrix} A & B \\ 0 & 0 \end{pmatrix} \cdot T\right]$$

Actually, $B_d(T)$ is an integral arising from the well-known solution of 1st-order ODE:

$$\bar{x}(T) = e^{AT}\bar{x}(0) + \int_0^T e^{A(T-\tau)} B u(0)\, d\tau = \underbrace{e^{AT}}_{A_d(T)} \bar{x}(0) + \underbrace{\left(\int_0^T e^{A\tau}\, d\tau\right) \cdot B}_{B_d(T)} \cdot u(0)$$

Discretization of variance equation

The exponential solution (proof elsewhere) is:

$$P(t) = e^{At} \cdot P(0) \cdot e^{A^T t} + \int_0^t e^{A\zeta} W e^{A^T\zeta}\, d\zeta$$

Setting $t = T$, the sampling period, we have

$$P_{k+1} = A_d(T) \cdot P_k \cdot A_d^T(T) + W_d(T)$$

with $W_d(T) = \int_0^T e^{A\tau} W e^{A^T\tau}\, d\tau$.

*The above expression is the variance equation of the discrete-time (sampled) stochastic process

$$x_{k+1} = A_d(T)x_k + B_d(T)u_k + w_k, \qquad E[w_k w_k^T] = W_d(T)$$

Numerical methods to compute $W_d(T)$

With $W_d(T) = \int_0^T e^{A\tau} W e^{A^T\tau}\, d\tau$, and $\mathcal{G}_c = W_d(\infty)$ being the gramian (stationary variance), we have several options:

- Direct computation with Matlab Symbolic toolbox (small matrices)
- Some formulae for stable systems, for non-integrating ones, using Jordan forms, ...
- Van Loan's formula, valid for any system but maybe ill-conditioned for large T.

Van Loan's formula is the chosen implementation in `kalmd` (and `lqrd` for control problems, dual to estimation), so we'll concentrate on it in this material. Good if sampling period is reasonably chosen and model is not "stiff" (i.e., very fast poles are elliminated via model reduction).

Van Loan's formula for W_d

Let us consider

$$\Theta := \begin{pmatrix} -A & W \\ 0 & A^T \end{pmatrix}$$

It's easily seen that $\Theta^2 = \begin{pmatrix} A^2 & WA^T - AW \\ 0 & (A^T)^2 \end{pmatrix}$, Θ^3, Θ^4, ... are upper triangular; so it is (via Taylor series argumentation) the exponential:

$$e^{\Theta \cdot t} = \begin{pmatrix} F_1(t) & G(t) \\ 0 & F_2(t) \end{pmatrix}$$

Being $F_1(0) = F_2(0) = I$, $G(0) = 0$, because $e^{\Theta \cdot 0} = \begin{pmatrix} I & 0 \\ 0 & I \end{pmatrix}$

Van Loan's formula (2)

From

$$\begin{pmatrix} \frac{dF_1}{dt} & \frac{dG}{dt} \\ 0 & \frac{dF_2}{dt} \end{pmatrix} = \frac{d}{dt} e^{\Theta \cdot t} = \Theta e^{\Theta t} = \begin{pmatrix} -A & W \\ 0 & A^T \end{pmatrix} \begin{pmatrix} F_1(t) & G(t) \\ 0 & F_2(t) \end{pmatrix} = \begin{pmatrix} -AF_1 & -AG + WF_2 \\ 0 & F_2 A^T \end{pmatrix}$$

we can write (using initial conditions in previous slide):

$$\frac{dF_1}{dt} = -AF_1 \qquad \Rightarrow \qquad F_1(t) = e^{-At} F_1(0) = e^{-At}$$

$$\frac{dF_2}{dt} = F_2 A^T \qquad \Rightarrow \qquad F_2(t) = F_2(0) e^{A^T t} = e^{A^T t}$$

$$\frac{dG}{dt} = -AG + WF_2 \qquad \Rightarrow \qquad G(t) = e^{-At} \underbrace{G(0)}_{zero} + \int_0^t e^{-A(t-\tau)} WF_2(\tau)\, d\tau$$

Van Loan's formula (3)

As $G(0) = 0$, replacing $F_2(\tau) = e^{A^T\tau}$, we have:

$$G(t) = e^{-At} \int_0^t e^{A\tau}\, W\, e^{A^T\tau}\, d\tau$$

Hence, as $F_2(t)^T = e^{At}$, we have the main result:

$$F_2(t)^T \cdot G(t) = e^{At} \cdot e^{-At} \int_0^t e^{A\tau}\, W\, e^{A^T\tau}\, d\tau = \int_0^t e^{A\tau}\, W\, e^{A^T\tau}\, d\tau = W_d(t)$$

Conclusions

The integral $W_d(T) = \int_0^T e^{A\tau}\, W\, e^{A^T\tau}\, d\tau$ can be computed from a matrix exponential (`expm` in Matlab):

$$e^{\Theta \cdot T} = e^{\begin{pmatrix} -A & W \\ 0 & A^T \end{pmatrix} \cdot T} = \begin{pmatrix} F_1(T) & G(T) \\ 0 & F_2(T) \end{pmatrix}$$

so $W_d = F_2(T)^T \cdot G(T)$. No need to carry out "integrals", bingo!.
Hence, implementation can be done in a couple of lines of Matlab code.

As both e^{AT} and e^{-AT} appear, there may be conditioning problems for large T or stiff poles. Reasonably chosen sampling periods and elliminating extremely non-dominant poles will give no problems, and it's the default implementation in `kalmd` (Matlab).