algorithm for computing the observability gramian update

zh296

March 2021

1 Derivation

We start with the heat equation from equation 1.3 in "Drew's First Writeup". That equation states

$$M\dot{u} + Au = Bz \tag{1}$$

To make the notation more compatible with our current paper, we rewrite the equation with different letters and invert the mass matrix. Now, the differential equation is,

$$\begin{aligned} \dot{x}(t) &= -M^{-1}Ax(t) + M^{-1}Bu(t) \\ y(t) &= Cx(t) \end{aligned} \tag{2}$$

Based on the above equation, the Lyapunov equation we want to solve is

$$(-M^{-1}A)P + P(-M^{-1}A)^T + CC^T = 0 \tag{3}$$

The P in the above equation is the observability gramian.

Now, based on the "low rank smith" paper, we can rewirte the above equation as

$$\begin{aligned} P = {} & (-M^{-1}A - \mu I)(-M^{-1}A + \mu I)^{-1}P(-M^{-1}A + \mu I)^{-1T}(-M^{-1}A - \mu I)^T \\ & - 2\mu(-M^{-1}A + \mu I)^{-1}CC^T((-M^{-1}A)^T + \mu I)^{-1} \end{aligned} \tag{4}$$

Let's denote $A_\mu = (-M^{-1}A - \mu I)(-M^{-1}A + \mu I)^{-1}$. Then, we can simplify the above equation as

$$\begin{aligned} P = {} & A_\mu P A_\mu^T \\ & - 2\mu(-M^{-1}A + \mu I)^{-1}CC^T((-M^{-1}A)^T + \mu I)^{-1} \end{aligned} \tag{5}$$

Now, suppose we are inside the greedy algorithm and want to update the observability gramian with just one more sensor. We can write $P_{next} = P_{pre} + D$.

$$\begin{aligned} P_{pre} + D = {} & A_\mu(P_{pre} + D)A_\mu^T \\ & - 2\mu(-M^{-1}A + \mu I)^{-1}(C_{pre} + C_{rank1})(C_{pre} + C_{rank1})^T((-M^{-1}A)^T + \mu I)^{-1} \end{aligned} \tag{6}$$

When we do $(C_{pre} + C_{rank1})(C_{pre} + C_{rank1})^T$, the cross terms will cancel out. Furthermore, we know P_{pre} satisfies the Stein equation, which is equation 3.4 in the "low rank smith" paper. Therefore, we have

$$\begin{aligned} D = &A_\mu D A_\mu^T \\ &- 2\mu(-M^{-1}A + \mu I)^{-1} C_{rank1} C_{rank1}^T ((-M^{-1}A)^T + \mu I)^{-1} \end{aligned} \tag{7}$$

Now, we use the infinite series expansion given in the stanford lecture,

$$D = \sum_{t=0}^{\infty} A_\mu^t (-2\mu(-M^{-1}A + \mu I)^{-1} C_{rank1} C_{rank1}^T ((-M^{-1}A)^T + \mu I)^{-1})(A_\mu^T)^t \tag{8}$$

or,

$$D = -2\mu \sum_{t=0}^{\infty} A_\mu^t (-M^{-1}A + \mu I)^{-1} C_{rank1} C_{rank1}^T ((-M^{-1}A)^T + \mu I)^{-1} (A_\mu^T)^t \tag{9}$$

2 Implementation Detail

We notice that $A_\mu, -M^{-1}A + \mu I)^{-1}, and (-M^{-1}A)^T + \mu I)^{-1}$ are the same throughout the entire process. Therefore, we can compute those three matrices in advance. Furthermore, we can diagonalize $A_\mu = A_{\mu v} A_{\mu d} A_{\mu v}^{-1}$

Now, we can write the D as

$$D = -2\mu(aa^T + bb^T + ...) \tag{10}$$

Furthermore, we have

$$D = -2\mu A_{sylvester} B_{sylvester} \tag{11}$$

$$A_{sylvester} = \begin{bmatrix} | & | & \\ a & b & \dots \\ | & | & \end{bmatrix}$$

$$B_{sylvester} = \begin{bmatrix} — & a^T & — \\ — & b^T & — \\ & \vdots & \end{bmatrix}$$

Then, the time complexity of computing one of those vectors is $O(n^2)$. The time complexity for forming the $A_{sylester} or B_{sylvester}$ is $O(n^2 * terms)$

Now, we have

$$\begin{aligned} det(P - 2\mu A_{sylester} B_{sylvester}) &= det(P(I - 2\mu P^{-1} A_{sylvester} B_{sylvester})) \\ &= det(P) det(I - 2\mu P^{-1} A_{sylvester} B_{sylvester}) \\ &= det(P) det(I - 2\mu B_{sylvester} P^{-1} A_{sylvester}) \end{aligned} \tag{12}$$

Now, we have

$$det(P - 2\mu A_{sylester} B_{sylvester}) = det(P) det(I + B_{sylvester} P^{-1} A_{sylvester}) \tag{13}$$