امتحان پایان‌ترم کنترل بهینه ۱

علی بنی اسد ۹۶۱۰۸۳۷۸

۱۵ تیر ۱۴۰۰

سوال اول

برای حل این سوال از کد ارسالی درس استفاده شده‌است ولی برای این سوال تغییراتی اعمال شده‌است که در ادامه به بررسی آن پرداخته می‌شود.

سیستم به فرم زیر نوشته شده‌است که در کد با اتفاده از تابع ode45 شبیه‌سازی می‌شود.

$$a = \begin{bmatrix} \dot{x_1} \\ \dot{x_2} \end{bmatrix} = \begin{bmatrix} x_2 \\ -0.4x_1^2 - 0.2x_2^2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$$

برای افزودن قیود صورت سوال از توابع جریمه به شکل Quadratic Extended استفاده شده‌است. این کار باعث عوض شدن ماتریس هملیتونین می‌شود.

$$\begin{aligned} g_1(x_2) &= x_2 - 0.5 \\ g_1(u) &= u - 0.5 \\ g_2(x_2) &= -x_2 - 0.5 \\ g_2(u) &= -u - 0.5 \end{aligned}$$

$$G_1(x_2) = \begin{cases} -\dfrac{1}{g_1(x_2)} & g_1(x_2) \leq \epsilon \\ -\dfrac{1}{\epsilon}\left(3 - \dfrac{3g_1(x_2)}{\epsilon} + \left(\dfrac{g_1(x_2)}{\epsilon}\right)^2\right) & g_1(x_2) > \epsilon \end{cases}$$

$$G_1'(x_2) = \begin{cases} \dfrac{1}{(x_2 - 0.5)^2} & g(x_2) \leq \epsilon \\ -\dfrac{1}{\epsilon}\left(-\dfrac{3}{\epsilon} + \dfrac{2x_2 - 1}{\epsilon^2}\right) & g_1(x_2) > \epsilon \end{cases}$$

$$G_2(x_2) = \begin{cases} -\dfrac{1}{g_2(x_2)} & g_2(x_2) \leq \epsilon \\ -\dfrac{1}{\epsilon}\left(3 - \dfrac{3g_2(x_2)}{\epsilon} + \left(\dfrac{g_2(x_2)}{\epsilon}\right)^2\right) & g_2(x_2) > \epsilon \end{cases}$$

$$G_2'(x_2) = \begin{cases} \dfrac{1}{(x_2 + 0.5)^2} & g(x_2) \leq \epsilon \\ -\dfrac{1}{\epsilon}\left(\dfrac{3}{\epsilon} + \dfrac{2x_2 + 1}{\epsilon^2}\right) & g_2(x_2) > \epsilon \end{cases}$$

$$G_1(u) = \begin{cases} -\dfrac{1}{g_1(u)} & g_1(u) \leq \epsilon \\ -\dfrac{1}{\epsilon}\left(3 - \dfrac{3g_1(u)}{\epsilon} + \left(\dfrac{g_1(u)}{\epsilon}\right)^2\right) & g_1(u) > \epsilon \end{cases}$$

$$G_1'(u) = \begin{cases} \dfrac{1}{(u - 0.5)^2} & g_1(u) \leq \epsilon \\ -\dfrac{1}{\epsilon}\left(-\dfrac{3}{\epsilon} + \dfrac{2u - 1}{\epsilon^2}\right) & g_1(u) > \epsilon \end{cases}$$

$$G_2(x_2) = \begin{cases} -\dfrac{1}{g_2(u)} & g_2(u) \leq \epsilon \\ -\dfrac{1}{\epsilon}\left(3 - \dfrac{3g_2(u)}{\epsilon} + \left(\dfrac{g_2(u)}{\epsilon}\right)^2\right) & g_2(u) > \epsilon \end{cases}$$

$$G_2'(u) = \begin{cases} \dfrac{1}{(u + 0.5)^2} & g_2(u) \leq \epsilon \\ -\dfrac{1}{\epsilon}\left(\dfrac{3}{\epsilon} + \dfrac{2u + 1}{\epsilon^2}\right) & g_2(u) > \epsilon \end{cases}$$

توابع جریمه دارای ضریب نیز هستند که در هر دوره محاسبات به روز می‌شود البته در کد حداقل ضریب 0.001 در نظر گرفته شده‌است که از این مقدار کوچکتر نشود.

$$\epsilon = -c\left(r_k\right)^2, \qquad a = 0.5, \qquad r_{k+1} = cr_k, \qquad c = 0.9, \qquad \min(r_k) = 0.001$$

$$\mathcal{H} = \vec{P}^T a(\vec{X}, u, t) + \frac{1}{2}\left(x_1^2 + x_2^2 + u^2 + r_k G(u) + r_k G(x_2)\right)$$

با تغییر ماتریس همیلتونین مشتق آن نسبت به $\vec{X}$ تغییر می‌کند که در کد لحاظ شده‌است.

$$\dot{\vec{P}} = -\frac{\partial \mathcal{H}}{\partial \vec{X}} = \begin{bmatrix} -x_1 + 0.4p_2 \\ -x_2 - p_1 + 0.4p_2x_2 - \frac{r_k}{2}G'(x_2) \end{bmatrix}$$

$$\begin{bmatrix} \dot{p_1} \\ \dot{p_2} \end{bmatrix} = \begin{bmatrix} -x_1 + 0.4p_2 \\ x_2(0.4p_2 - 1) - p_1 - \frac{r_k}{2}G'(x_2) \end{bmatrix}$$

بر اساس روند بالا و موارد قبلی که در کد پیاده سازی شده‌بود می‌توان مشتق همیلتونین نسبت به تلاش کنترلی(u) بدست آورد، سپس مشتق ترم جریمه تلاش کنترلی را نیز اضافه می‌کنیم.

$$\frac{\partial \mathcal{H}}{\partial u} = Ru + PB + r_k G'(u)$$

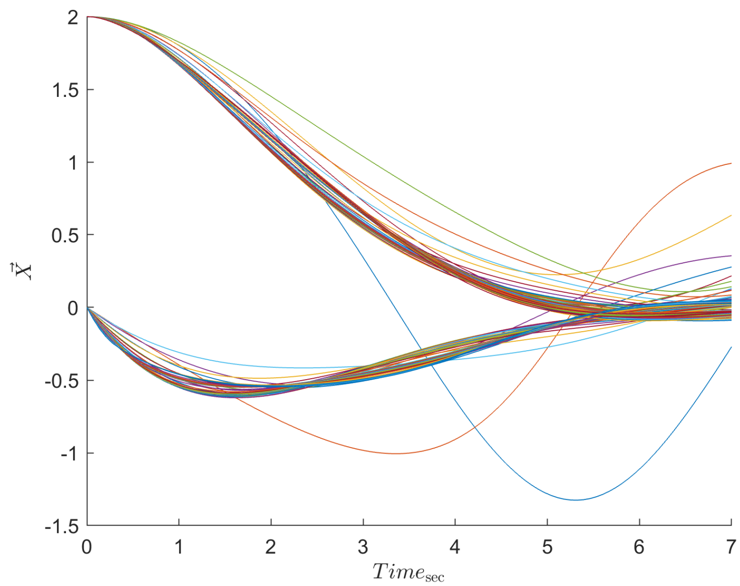
در بالا روش تحلیلی بدست آوردن گرادیان توضیح داده شد. تابع هزینه هم از انتگرال زیر بدست می‌آید.

$$J = 10(x_{1_{(t_f)}}^2 + x_{2_{(t_f)}}^2) + \frac{1}{2}\int_0^7 x_1^2 + x_2^2 + u^2 + r_k G(x_2) + r_k G(u)$$

کد پیوست شده توانایی حل با چهار روش را دارد اگه از منو می‌توان روش را انتخاب کرد. نتایج چهار روش آورده شده‌است.

- Steepest Descent + Quadratic Interpolation

Figure 1: Steepest Descent + Quadratic Interpolation



- Steepest Descent + Golden Section

Figure 2: Steepest Descent + Golden Section

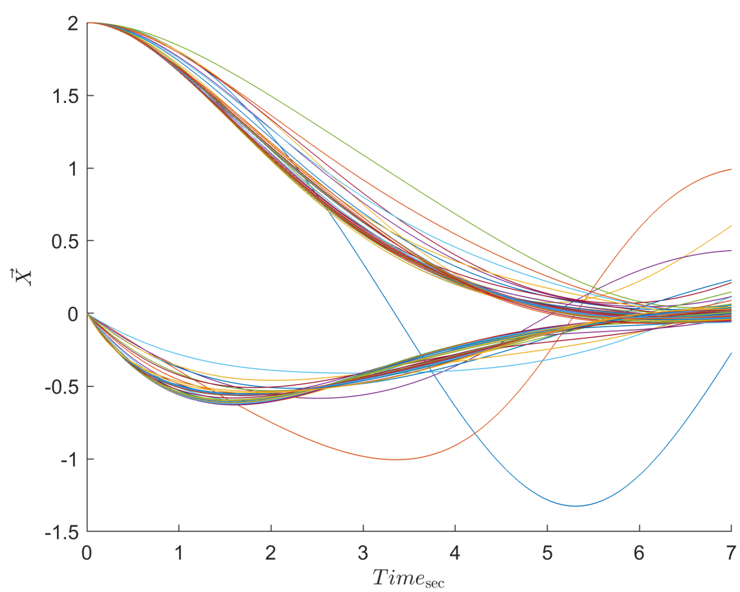


Figure 3: BFGS + Quadratic Interpolation

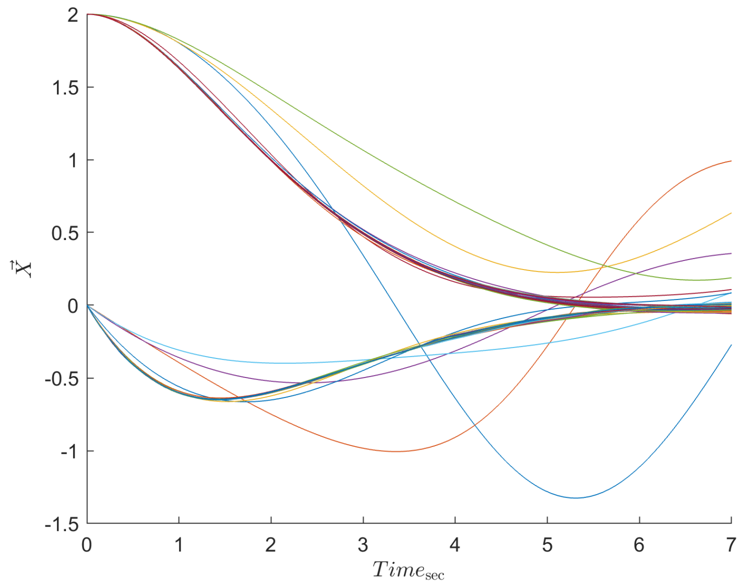
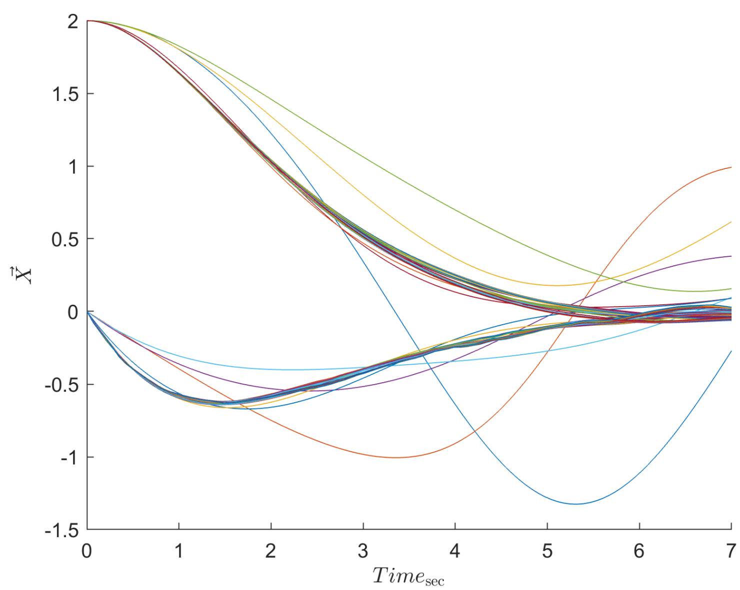


Figure 4: BFGS + Golden Section



سوال دوم

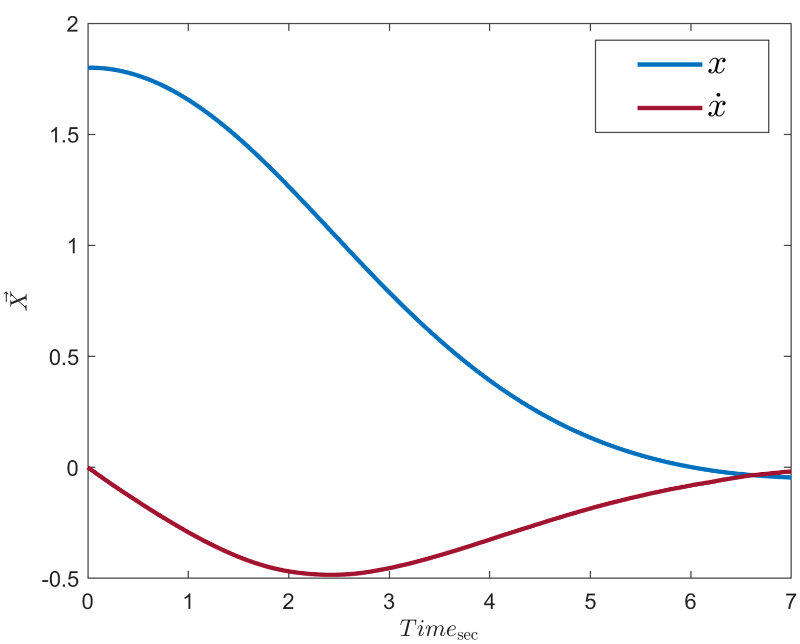
در این قسمت با توجه به نقطه اولیه ممکن است جواب درستی وجود نداشته باشد برای این کار اگر از قید رد شد تابع هزینه به شدت افزایش می‌یابد که با اینکار می‌توان به جواب حدودی همراه با قید رسید. مزیت اینکار نسبت به عدم در نظرگرفتن حالت خارج از آن یا قرار دادن Inf برای آن این است که تابع interpolation عملکرد بهتری خواهد داشت و اگر شرایط اولیه مطلوب نبود برنامه دچارگمراهی نمی‌شود و بازهم بهترین مسیر را انتخاب می‌کند ولی اگر شرایط اولیه درست باشد با توجه به تفاوت بسیار زیاد تابع هزینه در شرایط خارج از قید اصلا از قید خارج نمی‌شود.

معدلات سیستم به فرم زیر نوشته شده‌است.

$$a = \begin{bmatrix} \dot{x_1} \\ \dot{x_2} \end{bmatrix} = \begin{bmatrix} x_2 \\ -0.4x_1^2 - 0.2x_2^2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$$

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \end{bmatrix} = \begin{bmatrix} x_2(k) \\ -0.4x_1(k)^2 - 0.2x_2^2(k) + u(k) \end{bmatrix} \Delta t + \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix}$$

Figure 5: Dynamic Programming result



گسسته سازی به صورت زیر انجام شد و کل برنامه در ۸ ثانیه اجرا شد. البته با توجه به گسسته سازی محدود interpolation با ماتریس‌های نسبتا بزرگی انجام شده‌است و دقت کار و سرعت را به شکل خوبی بالا می‌برد.

- $\Delta t = 0.05\,\text{sec}$
- $u = -0.5 : 0.5 : 0.5$

- $x = -1 : 1 : 3$
- $\dot{x} = -1 : 0.5 : 1$

حال با توجه به قانون کنترل بدست آمده می‌توان بدون انجام محاسبات زیاد قبلی خیلی سرعتر به جواب رسید.