Trajectory Optimization Methods for Dynamical Systems

Nabila Adawy
B19-RO

Research Questions

- How does the form of representing the dynamical system affect the speed of convergence to the optimal trajectory?
- What is the impact of designing a Time-Variant LQR controller in tracking the optimal trajectory and stabilizing the dynamical system?

Literature Review

Introduction:

We begin this chapter in section 2.1 with an overview of the trajectory optimization problem and by explaining the approach and the techniques we intend to implement. After that, we implement those techniques on the Cart-Pole system, which we explain in Section 2.2. Then, in section 2.3, we explain the design of the time-variant LQR controller used for tracking the optimal trajectory and stabilizing the dynamical system. In the end, we conclude by discussing the possible novelty we intend to bring.

2.1 Trajectory Optimization Problem:

Trajectory optimization is a set of mathematical methods used to achieve the ideal or desired behavior for a dynamical system. They are widely used in many applications, such as aerospace, industrial, and robotics applications. The trajectory optimization problem is the technique used to solve an open loop optimal control problem. It is a way to find how the system should behave, starting from the initial condition to reach the desired condition with the minimum possible effort (to save power/time/cost). Trajectory optimization methods can be mainly categorized into two sub-categories, as Matthew [1] explained:

- **Direct** Collocation Method: which converts the trajectory optimization problem (continuous-time) into a non-linear program (NLP) and then tries to find the optimal solution/trajectory.
- **Indirect** Collocation Method: which tries to find the optimal conditions analytically and then tries to solve them numerically after discretization.

Since the direct collocation method has many techniques, we focus on Direct Trapezoidal Collocation method. Matthew [1] shows that this method is done by using the trapezoidal quadrature (trapezoid rule for integration) to convert the continuous aspect of the NLP into a discrete approximation form. This trapezoidal rule is applied between each collocation point along the trajectory. Which yields the following equations:

- For the objective function (we try to minimize):

$$\int_{t_0}^{t_f} w(\tau,\ x(\tau),\ u(\tau))\, d\tau \ \approx\ \sum_{k=0}^{N-1} \frac{1}{2}\, h_k\ .\ (w_k + w_{k+1})$$

Where $h_k = t_{k+1} - t_k$, $t_k = k\frac{T}{N}$, and N is the number of segments, T is the total time.

- For the system dynamics (the collocation constraints) in the state space **(implicit)** form :

$$\dot{x} \ = f(t, x, u)$$

$$\int_{t_k}^{t_{k+1}} \dot{x}\, dt \ = \int_{t_k}^{t_{k+1}} f(x, u)\, dt$$

$$x_{k+1} - x_k \ = \frac{1}{2} h_k\ (f_k + f_{k+1}),\ k \in 0, .., (N-1)$$

Where x_k is a decision variable.

2.1.1 Discrete Explicit form:

We implement three different techniques for representing the system dynamics. The first technique represents the system dynamics in **implicit** form (as mentioned above). The second technique uses the **explicit** form of the dynamics as follows:

$$M(q)\,\ddot{q} \ +\ C(q,\ \dot{q})\dot{q} \ +\ G(q) \ =\ B(q)\, u$$
$$or$$
$$M(q)\,\ddot{q} \ +\ h(q,\ \dot{q}) \ =\ B(q)\, u$$

Here, we need to change the collocation constraints by adding two integral constraints: one on the relation between the state and the velocity and another between the velocity and the acceleration:

$$q_{k+1} \ -\ q_k \ =\ \frac{1}{2}\, h_k\ (\dot{q}_{k+1} + \dot{q}_k)$$
$$\dot{q}_{k+1} \ -\ \dot{q}_k \ =\ \frac{1}{2}\, h_k\ (\ddot{q}_{k+1} + \ddot{q}_k)$$

This technique has more variables than the previous one, since the acceleration variables are included.

2.1.2 Variational Integrator :

We are interested in deriving integrators from discrete lagrangian mechanics for mechanical systems. Matthew West [3] shows that any integrator which is the discrete Euler-Lagrange equation for some discrete Lagrangian is called a variational integrator. Since our system is nonconservative and has force acts on it. We can use the following forced discrete Euler-Lagrange equations:

$$M\left(\frac{q_{k+1} - 2q_k + q_{k-1}}{\Delta t^2}\right) \ =\ -\nabla V(q_k)\ +\ F(q_k)$$

Where M is the mass matrix, q is the state, V is the potential energy, F is the force applied to the system.

We need to modify this equation because Matthew [3] considers the M matrix as constant (does not depend on the state variables), but in our system, the M matrix depends on the state q_k. The derivation of the modified equation is explained later in the methodology chapter. The advantage of this technique is that it has fewer variables than the previous two techniques.

2.2 Cart-Pole System Dynamics:

Matthew [1] applied the Trapezoidal Collocation method on a Cart-Pole system. Cart-Pole system is a second-order under-actuated dynamical system. This system consists of a cart that travels along a horizontal rail, and it is powered by a motor that can apply a force along the rail; the cart is attached to a pendulum that can swing freely. The swing-up problem for the Cart-Pole system is to find the force as a function of time that will move the cart and have this pendulum - which has no motor actuated on it - swing up to be balanced above the cart. Fig. 1 below shows the Cart-Pole system, as demonstrated by Mathew [1].

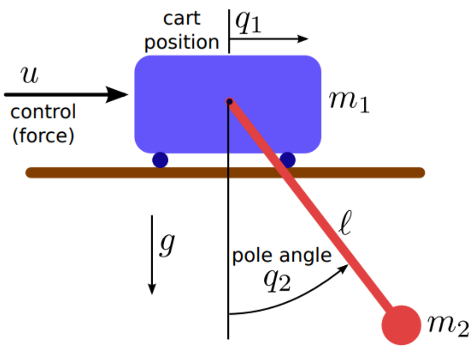

Fig. 1.: Cart-Pole system

Although the system looks simple and easy to derive its equation of motion, many mistakes in its dynamics are found in many papers. After searching and diving into the dynamics more, we finally found some resources which use the correct equations of motion of the Cart-Pole system. Razvan [2] proved those equations and their derivation.

- The dynamics equation in the state-space (Implicit) form [2]:

$$\dot{\boldsymbol{x}} = f(\boldsymbol{x}, u)\ ,\ \boldsymbol{x} = \begin{bmatrix} x \\ \theta \\ \dot{x} \\ \dot{\theta} \end{bmatrix},\ \dot{\boldsymbol{x}} = \begin{bmatrix} \dot{x} \\ \dot{\theta} \\ \ddot{x} \\ \ddot{\theta} \end{bmatrix}$$

$$\dot{\boldsymbol{x}} = f(\boldsymbol{x}, u) = \begin{bmatrix} \dot{x} \\ \dot{\theta} \\ \ddot{x} \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} \dot{x} \\ \dot{\theta} \\ \dfrac{m_p\, sin\theta\, \left(l\, \dot{\theta}^2 + g\, cos\theta\right)}{m_c + m_p\, sin^2\theta} \\ \\ \dfrac{-m_p\, l\dot{\theta}^2\, cos\theta\, sin\theta\, - (m_c + m_p)\, g\, sin\theta}{l\left(m_c + m_p\, sin^2\theta\right)} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \dfrac{1}{m_c + m_p\, sin^2\theta} \\ \\ \dfrac{-cos\theta}{l\left(m_c + m_p\, sin^2\theta\right)} \end{bmatrix} u \qquad (1)$$

- The dynamics equation in the Explicit form:

$$\begin{bmatrix} m_c + m_p & m_p l\, cos\theta \\ m_p l\, cos\theta & m_p l^2 \end{bmatrix} \begin{bmatrix} \ddot{x} \\ \ddot{\theta} \end{bmatrix} + \begin{bmatrix} 0 & -m_p l\, \dot{\theta}\, sin\theta \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ m_p\, gl\, sin\theta \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} u \tag{2}$$

$$Where$$

$$M(q) = \begin{bmatrix} m_c + m_p & m_p l\, cos\theta \\ m_p l\, cos\theta & m_p l^2 \end{bmatrix}$$

$$h(q, \dot{q}) = \begin{bmatrix} -m_p l\, \dot{\theta}\, sin\theta \\ m_p\, gl\, sin\theta \end{bmatrix}$$

$$B(q) = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

- The dynamics equation using Discrete Variational Integrator:

$$\begin{bmatrix} m_c + m_p & m_p l\, cos(q_2) \\ m_p l\, cos(q_2) & m_p l^2 \end{bmatrix} \left(\frac{q_{k+1} - 2q_k + q_{k-1}}{\Delta t^2} \right) + \begin{bmatrix} -m_p l\;\; sin(q_2) \left(\dfrac{q_k \;-\; q_{k-1}}{\Delta t} \right) \\ m_p\, gl\, sin(q_2) \end{bmatrix} - \begin{bmatrix} 1 \\ 0 \end{bmatrix} u = 0 \tag{3}$$

2.3 Time-Variant LQR Controller:

Since the trapezoidal collocation approach is an open-loop control problem, it should be combined with the closed-loop controller to follow the optimal trajectory obtained and to achieve stability of the system at the desired trajectory. Banerjee, *et al.* [4] reached to the conclusion that the performance of LQR is more reliable and satisfactory than the Pole Placement technique and PD controller. So we implement the time-variant LQR controller as follows:

- A feedback controller whose equations are given below:

$$J_c = \sum_{k=0}^{\infty} \left(x_k^T\, Q\, x_k + u_k^T\, R\, u_k \right) + x_k^T\, H\, x_k$$

- The optimal control sequence minimizing the performance index is given by:

$$u_k = -K_{LQR}\, (x_k - x_{d_k}) + u_{d_k}$$

- Where K_{LQR} is time-variant gain given by:

$$K_{N-K} = \left(R_{N-k} + B_{N-k}^T\, P_{N-k+1}\, B_{N-k} \right) B_{N-k}^T\, P_{N-k+1}\, A_{N-k}$$

- P updates with difference Riccati equation (DRE):

$$P_{N-k} = Q_{N-k} + K_{N-k}^T\, R_{N-k} K_{N-k} + (A_{N-k} - B_{N-k}\, K_{N-k})^T\, P_{N-k+1}\, (A_{N-k} - B_{N-k}\, K_{N-k})$$

- Where A and B are the discretized Jacobians in N points of the desired trajectory.

$$A = \frac{\partial f}{\partial x}\Big|_{x_{d_k},\, u_{d_k}},\; B = \frac{\partial f}{\partial u}\Big|_{x_{d_k},\, u_{d_k}}$$

$$\dot{x} = f(x_d, u_d) = \frac{\partial f}{\partial x}\Big|_{x_{d_k},\, u_{d_k}} (x - x_d) + \frac{\partial f}{\partial u}\Big|_{x_{d_k},\, u_{d_k}} (u - u_d)$$

Introducing the tracking error $\tilde{x}$

$$\dot{\tilde{x}} = A\, \tilde{x} + B\, \tilde{u}$$

Conclusion:

Our goal is to implement a trajectory optimization method with three different techniques to represent the system dynamics. All the mentioned works are based on simulations using Matlab and other tools. In our case, we use CasADi python (CasADi is an open-source framework for nonlinear optimization and algorithmic differentiation) to do our simulations. The solution is obtained using IPOPT (Interior Point Optimizer) solver (which is a class to solve the NLP program).

The novelty we seek is to compare between the three mentioned techniques, and prove that the number of the decision variables used in the trajectory optimization problem affects on the speed of the solution convergence. The less the number of decision variables the faster the solution converges. This could be useful for dynamical systems with larger DoF such as quadcopter or quadruped. We also combine all the processes, from calculating the optimal trajectory, to designing the appropriate controller to follow this trajectory and then stabilize the system at the desired conditions.

References:

[1] Kelly, Matthew. "An introduction to trajectory optimization: How to do your own direct collocation." SIAM Review 59.4 (2017): 849-904. https://epubs.siam.org/doi/pdf/10.1137/16M1062569

[2] Florian, Razvan V. "Correct equations for the dynamics of the cart-pole system." Center for Cognitive and Neural Studies (Coneural), Romania (2007). https://coneural.org/florian/papers/05_cart_pole.pdf

[3] West, Matthew. "Variational Integrators". California Institute of Technology, 2004. https://thesis.library.caltech.edu/2492/1/west_thesis.pdf

[4] Banerjee, Ramashis, and Arnab Pal. "Stabilization of inverted pendulum on cart based on pole placement and lqr." 2018 International Conference on Circuits and Systems in Digital Enterprise Technology (ICCSDET). IEEE, 2018. https://ieeexplore.ieee.org/abstract/document/8821196