A Fast Robust approach to Optimal Control

Dhruv Shah

Department of Electrical Engineering, IIT Bombay

May 25, 2022

What is Optimal Control

Consider the system $\Sigma : \dot{x} = f(x, u, t)$ where $x \in \chi$ and $u \in \mathcal{U}$. Suppose the control objective is to steer the state x to the target set χ_f in some fixed time T_f satisfying certain control and state constraints. Let $J = \varphi(x(T_f), T_f) + \int_0^{T_f} L(x, u, t)dt$ be the performance index for this task. The optimal control u^* is the control that minimizes

$$
\begin{aligned}
\min_{u} \quad & \varphi(x(T_f), T_f) + \int_0^{T_f} L(x, u, t)dt \\
\text{s.t.} \quad & \dot{x} = f(x, u, t), \\
& u \in \mathcal{U}, \\
& x(T_f) \in \chi_f
\end{aligned}
$$

Necessary conditions for optimality are obtained by introducing a co-state λ which has the same dimension as x and putting the first variation of J while the constraints are being satisfied to zero. This leads to the Two Point Boundary Value Problem (TPBVP).

Two Point Boundary Value Problem (TPBVP)

The solution to the Optimal Control Problem (OCP) using indirect methods boils down to solving the TPBVP which for a specific case (Fixed Endpoint, Fixed Time) takes the form

$$\begin{aligned}
\dot{x} &= f(x, u, t) \\
\dot{\lambda} &= -\left(\frac{\partial f}{\partial x}\right)^T \lambda - \left(\frac{\partial L}{\partial x}\right)^T \\
\left(\frac{\partial f}{\partial u}\right)^T \lambda &+ \left(\frac{\partial L}{\partial u}\right)^T = 0
\end{aligned}$$

with the boundary conditions

$$\begin{aligned}
x(0) &= x_0 \\
x(T_f) &= x_f
\end{aligned}$$

There are various numerical methods to solve the TPBVP but they can get computationally heavy for complicated systems

Problem Statement

We are looking for a quick way to solve the TPBVP where the solution need not be optimal but is good enough for practical applications i.e we are searching for a fast suboptimal solution to the TPBVP. The consequences of succeeding in doing this would include

1. A fast solution to the TPBVP will be useful in problems where time is a critical factor (Missile Guidance Problems, Collision Avoidance Problems, etc)
2. The robustness of the system against large disturbances can be guaranteed by quickly recomputing a solution after a disturbance.
3. An iterative algorithm is possible where the solution is recomputed after every k time steps to make the solution more optimal and robust against disturbances. (Similar to MPC)

Interpretation of the Solution to the Discretised TPBVP

We use the euler discretisation and formulate the TPBVP as:

$$x(t+1) = x(t) + \frac{1}{2}\Delta t(F(x(t), \lambda(t), t) + F(x(t-1), \lambda(t-1), t-1))$$
$$\lambda(t+1) = \lambda(t) + \frac{1}{2}\Delta t(G(x(t), \lambda(t), t) + G(x(t-1), \lambda(t-1), t-1))$$

with boundary conditions : $x(0) = x_0 \quad x(T_f) = x_f$
Let the solution to the above set of equations using some optimisation algorithm be $\{x_N[k]\}_{k=0}^{N-1}, \{\lambda_N[k]\}_{k=0}^{N-1}$ and let the true optimal solution be $x^*(t), y^*(t)$

$$\varepsilon 1_N(k) = ||(x_N[k], \lambda_N[k])^T - (x^*(kh), \lambda^*(kh))^T||$$

where h is the time step.
How does this error $\varepsilon 1_N$ look like? Can we obtain upper bounds for this error? Furthermore, let T(N) be the computation time to compute the solution to the Discretised TPBVP. How does this scale with N?

Behaviour of the Discretised solution on the true dynamics

Let $u_N(t)$ be some interpolation of $\{u_N[k]\}_{k=0}^{N-1}$. Let $x_N(t)$ be the trajectory generated by the system under action $u_N(t)$

$$\varepsilon 2_N(k) = ||x_N(hk) - x_N[k]||$$

where h is the time step

How does the error $\varepsilon 2_N$ look like and can we design some feedback law to minimize this error?

Feedback Strategies

1. Motivation for using feedback : Small errors in the state can lead to poor performance when operating in open loop
2. Adding feedback provides robustness to the solution

Various feedback strategies can be tried out. The idea behind the most standard algorithm is as follows: If the initial point x_0 is perturbed (1), how does the optimal control $u(t)$ change so that the new OCP is solved.

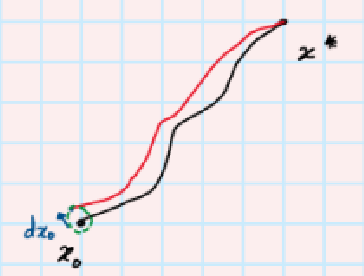


Figure: Perturbing the initial conditions

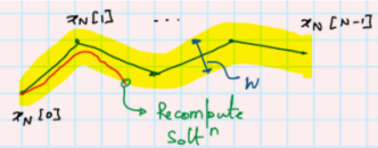
This gives rise to solving the following Ricatti equation for S

$$\dot{S} = -Q + MR^{-1}M^T - A^TS - SA + SBR^{-1}B^TS$$

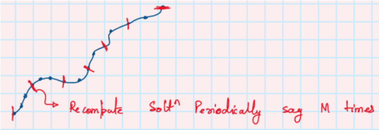
Let $K(t) = R^{-1}(B^TS + M^T)$. then, $u(t) = u_{ff}(t) + K(t)[x_{ff}(t) - x(t)]$

If we put $\dot{S} = 0$ in the Ricatti equation and solve for S, the feedback is termed as Quasi-Stationary LQR feedback. Solving the time dependent Ricatti equation gives rise to the time-dependent LQR feedback.
One can also choose a constant gain K heuristically, this is termed as heuristic PD feedback.

Recomputing the Optimal Solution : Periodic vs Tube based



(a) Tube based



(b) Periodic

How to choose W (tube width) and M (frequency of recomputing the solution)?
Furthermore for periodic recomputing, is there some optimal choice of the pair (N,M) ?

Next Steps : Simulation Studies

1. Explore using simulations, T(N) and error from the optimal solution. (Plot J vs N)
2. Explore various feedback strategies and compare their performance
3. Use different discretisations (specifically symplectic discretisations) and compare their performance
4. Compare the two strategies of recomputing the solutions : Periodic and Tube based

Next Steps: Theoretical Studies

1. Come up with error bounds due to discretisations
2. Come up with error bounds due to true dynamics after adding feedback
3. Merge the above to obtain the final optimality bounds
4. Analyze the algorithm of recomputing the solution and obtain performance bounds

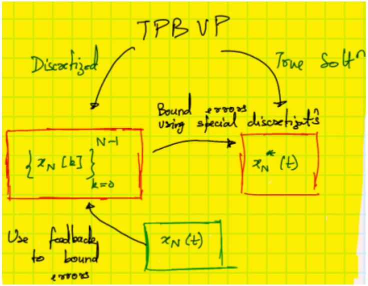


Figure: Error Analysis