CS 287: Advanced Robotics
Fall 2009

Lecture 8: Control 7: MPC --- Feedback Linearization --- Controllability --- Lagrangian Dynamics

Pieter Abbeel
UC Berkeley EECS

Model predictive control (MPC)

- Optimal control problem

  Given a system with (stochastic) dynamics: $x_{t+1} = f(x_t, u_t, w_t)$ Find the optimal policy π which minimizes the expected cost:

$$\min_{\pi} \mathrm{E}[\sum_{t=0}^{H} g(x_t, u_t)|\pi]$$

- MPC:

  For $t = 0, 1, 2, \ldots$

  1. Solve

$$\begin{aligned} \min_{u_t, u_{t+1}, \ldots, u_H} \quad & \sum_{k=t}^{H} g(x_k, u_k) \\ \text{s.t.} \quad & x_{k+1} = f(x_k, u_k, 0) \ \ \forall k = t, t+1, \ldots, H-1 \end{aligned}$$

  2. Execute u_t from the solution found in (1).

- In practice, one often ends up having to solve:

$$\begin{aligned} \min_{u_t, u_{t+1}, \ldots, u_{t+h-1}} \quad & \sum_{k=t}^{t+h} g(x_k, u_k) + \bar{g}(x_{t+h}, u_{t+h}) \\ \text{s.t.} \quad & x_{k+1} = f(x_k, u_k, 0) \ \ \forall k = t, t+1, \ldots, t+h-1 \end{aligned}$$

Single shooting

- At core of MPC, need to quickly solve problems of the form:

$$\begin{aligned} \min_{u_t, u_{t+1}, \ldots, u_H} \quad & \sum_{k=t}^{H} g(x_k, u_k) \\ \text{s.t.} \quad & x_{k+1} = f(x_k, u_k, 0) \ \ \forall k = t, t+1, \ldots, H-1 \end{aligned}$$

- Single shooting methods directly solve for

  i.e., they solve:

- Underneath, this typically boils down to iterating:
  - For the current simulate and find the state sequence
  - Take the 1st (and 2nd) derivatives w.r.t.
- Note: When taking derivatives, one ends up repeatedly applying the chain rule and the same Jacobians keep re-occurring
- → Beneficial to not waste time re-computing same Jacobians; pretty straightforward, various specifics with their own names. (E.g., back-propagation.)

Single shooting drawback

- Numerical conditioning of the problem:
  - Influence on objective function of earlier actions vs. later actions
- What happens in case of a non-linear, unstable system?

Multiple shooting/Direct collocation

- Keep the state at each time in the optimization problem:

$$\begin{aligned} \min_{u_t, u_{t+1}, \ldots, u_H, x_t, x_{t+1}, \ldots, x_H} \quad & \sum_{k=t}^{H} g(x_k, u_k) \\ \text{s.t.} \quad & x_{k+1} = f(x_k, u_k, 0) \ \ \forall k = t, t+1, \ldots, H-1 \\ & h_k(x_k, u_k) \leq 0 \ \ \forall k = t, t+1, \ldots, H-1 \end{aligned}$$

- Larger optimization problem, yet sparse structure.
- Special case: Linear MPC: f linear, h, g convex → convex opt. problem, "easily" solved

Sequential Quadratic Programming (SQP)

- Goal: solve

$$\begin{aligned} \min_{u_t, u_{t+1}, \ldots, u_H, x_t, x_{t+1}, \ldots, x_H} \quad & \sum_{k=t}^{H} g(x_k, u_k) \\ \text{s.t.} \quad & x_{k+1} = f(x_k, u_k, 0) \ \ \forall k = t, t+1, \ldots, H-1 \\ & h_k(x_k, u_k) \leq 0 \ \ \forall k = t, t+1, \ldots, H-1 \end{aligned}$$

- SQP: Iterates over
  - Linearize f around current point $(\boldsymbol{u}, \boldsymbol{x})$, quadraticize g, h around current point
  - Solve the resulting Quadratic Programming problem to find the updated "current point" (u, x)
- Corresponds to:
  - Write out the first-order necessary conditions for optimality (the Karuhn-Kuhn-Tucker (KKT) conditions)
  - Apply Newton's method to solve the (typically non-linear) KKT equations

Sequential Quadratic Programming (SQP)

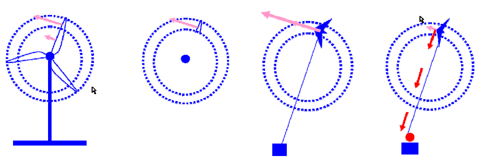
- Not only method, but happens to be quite popular
- Packages available, such as SNOPT, SOCS.
- Many choices underneath:
  - Quasi-Newton methods
- Compared to single shooting:
  - Easier initialization (single shooting relies on control sequence)
  - Easy to incorporate constraints on state / controls
  - More variables, yet good algorithms leverage sparsity to offset this

Further readings

- Tedrake Chapter 9.
- Diehl, Ferreau and Haverbeke, 2008, Nonlinear MPC overview paper
- Francesco Borelli (M.E., UC Berkeley): taught course in Spring 2009 on (linear) MPC
- Packages:
  - SNOPT, ACADO, SOCS, …
- We have ignored:
  - Continuous time aspects
  - Details of optimization methods underneath --- matters in practice b/c the faster the longer horizon
  - Theoretical guarantees

Related intermezzo: Nonlinear control applied to kite-based power generation

[Diehl + al.]



- Many companies pursuing this: Makani, KiteGen, SkySails, AmpyxPower, …
- Number from Diehl et al.: For a 500m2 kite and 10m/s wind speed (in sim) can produce an average power of more than 5MW
- Technically interesting aspect in particular work of Diehl et al.: incorporate open-loop stability into the optimization problem.
  - Only possible for non-linear systems
  - The criterion quantifies how much deviation from the nominal trajectory would amplify/decrease in one cycle

Non-minimum phase example

[Slotine and Li, p. 195, Example II.2]

Consider the linear system

$$\ddot{y} + 2\dot{y} + 2y = -\dot{u} + u$$

The system is non-minimum phase because it has a zero at $p = 1$. Assume that perfect tracking is achieved, *i.e.*, that $y(t) = y_d(t)$, $\forall\, t \geq 0$. Then, the input u satisfies

$$\dot{u} - u = -(\ddot{y}_d + 2\dot{y}_d + 2y_d)$$

Since this represents an unstable dynamics, u diverges exponentially. Note that the above dynamics has a pole which exactly coincides with the unstable zero of the original system, *i.e.*, perfect tracking for non-minimum phase systems can be achieved only by infinite control inputs. By writing u as

$$u = -\frac{p^2 + 2p + 2}{p - 1}\, y_d$$

we see that the perfect-tracking controller is actually inverting the plant dynamics. $\square$

Feedback linearization

Feedback linearization

Feedback linearization

Feedback linearization

Feedback linearization

- Further readings:
  - Slotine and Li, Chapter 6
  - Isidori, Nonlinear control systems, 1989.

Announcements

- Reminder: No office hours today.
  - [Feel free to schedule over email instead]

Controllability [defn., linear systems]

- A system $x_{t+1} = f(x_t, u_t)$ if for all x_0 and all x, there exists a time k and a control sequence $u_0, \ldots, u_{k-1}$ such that $x_t = x$.

Fact. The linear system $x_{t+1} = Ax_t + Bu_t$ with $x_t \in \Re^n$ is controllable iff $[B \ AB \ A^2B \ \ldots \ A^nB]$ is full rank.

Lagrangian dynamics

[From: Tedrake Appendix A]

Lagrangian dynamics: example