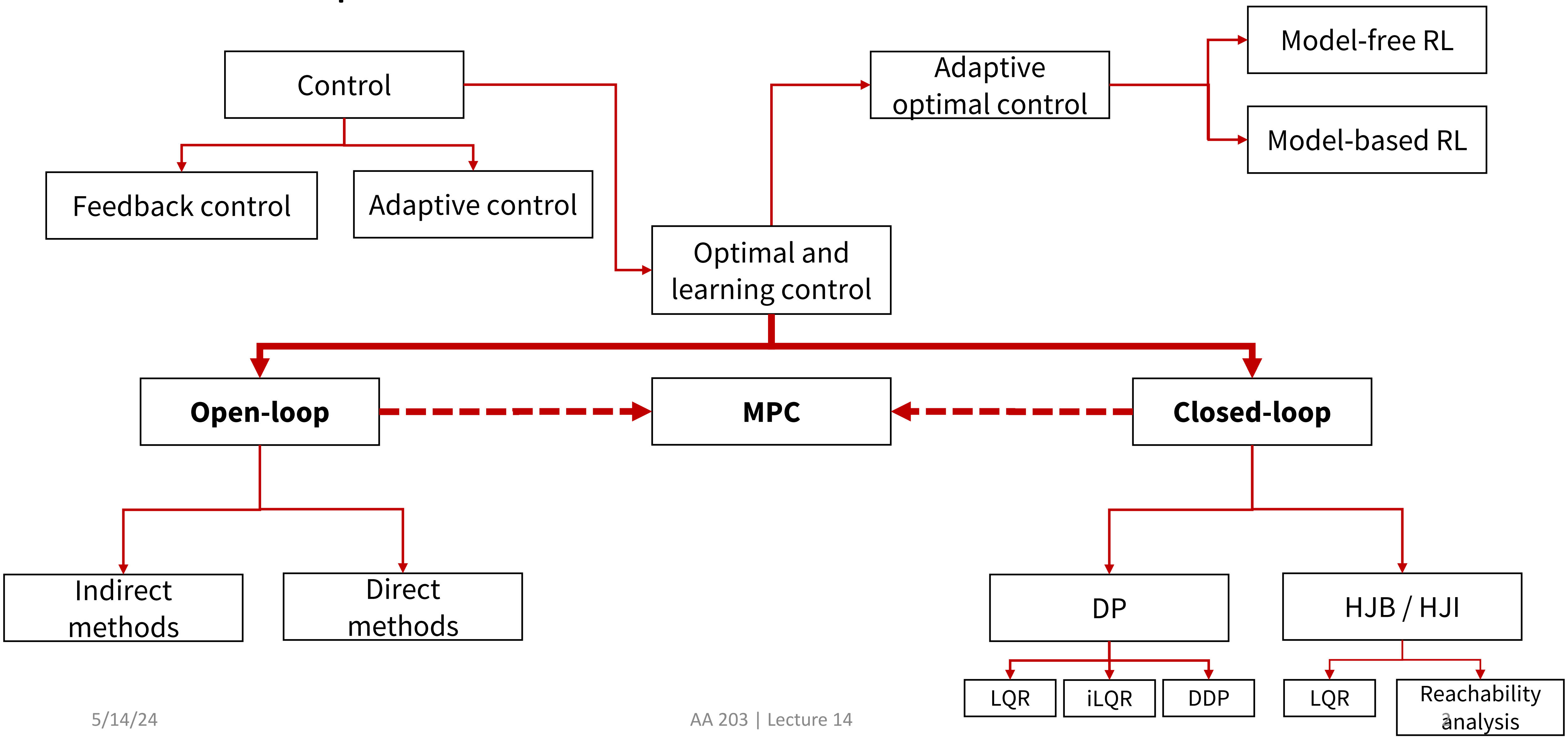
AA203
Optimal and Learning-based Control

Explicit MPC, practical considerations, Robust MPC

Stanford University

Autonomous Systems Lab

Roadmap



Control
Feedback control
Adaptive control
Optimal and learning control
Adaptive optimal control
Model-free RL
Model-based RL
Open-loop
MPC
Closed-loop
Indirect methods
Direct methods
DP
HJB / HJI
LQR
iLQR
DDP
LQR
Reachability analysis

Model predictive control

- Explicit MPC
- Implementation aspects of MPC
- Robust MPC

- Reading:
  - F. Borrelli, A. Bemporad, M. Morari. *Predictive Control for Linear and Hybrid Systems*, 2017.
  - J. B. Rawlings, D. Q. Mayne, M. M. Diehl. *Model Predictive Control: Theory, Computation, and Design*, 2017.

Explicit MPC

- In some cases, the MPC law can be *pre-computed* → no need for online optimization
- Important case: constrained LQR

$$J_0^*(\mathbf{x}) = \min_{\mathbf{u}_0,\dots,\mathbf{u}_{N-1}} \mathbf{x}_N^T P \mathbf{x}_N + \sum_{k=0}^{N-1} \mathbf{x}_k^T Q \mathbf{x}_k + \mathbf{u}_k^T R \mathbf{u}_k$$

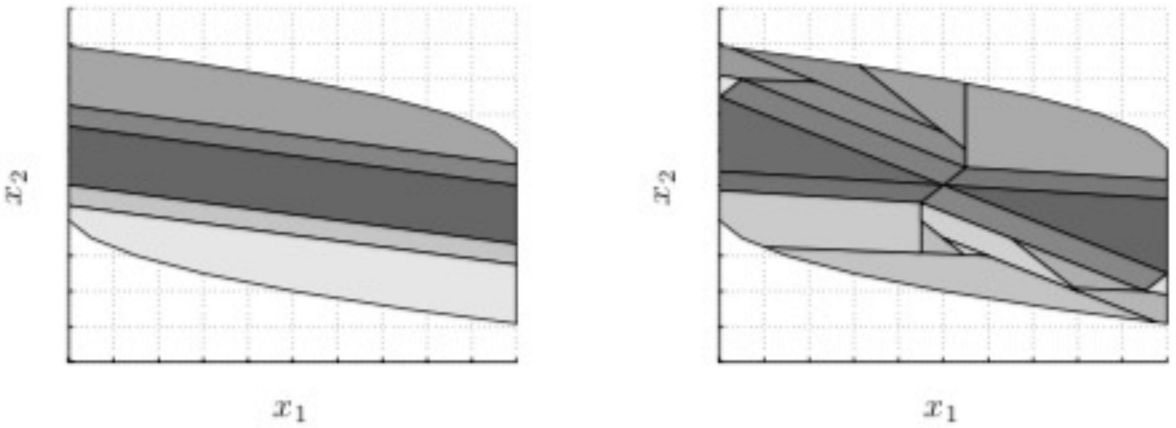
$$\begin{aligned} \text{subject to} \quad & \mathbf{x}_{k+1} = A\mathbf{x}_k + B\mathbf{u}_k, \quad k = 0, \dots, N-1 \\ & \mathbf{x}_k \in X, \quad \mathbf{u}_k \in U, \quad k = 0, \dots, N-1 \\ & \mathbf{x}_N \in X_f \\ & \mathbf{x}_0 = \mathbf{x} \end{aligned}$$

Explicit MPC

- The solution to the constrained LQR problem is a control which is a continuous piecewise affine function on polyhedral partition of the state space X, that is $\mathbf{u}_k^* = \pi_k(\mathbf{x}_k)$ where

$$\pi_k(\mathbf{x}) = F_k^j \mathbf{x} + g_k^j \quad \text{if} \quad H_k^j \mathbf{x} \le K_k^j, \quad j = 1, \dots, N_k^r$$

- Thus, online, one has to locate in which cell of the polyhedral partition the state $\mathbf{x}$ lies, and then one obtains the optimal control via a look-up table query



Tuning and practical use

- At present there is no other technique other than MPC to design controllers for general large linear multivariable systems with input and output constraints with a stability guarantee
- Design approach (for squared 2-norm cost):
  - Choose horizon length N and the control invariant target set X_f
  - Control invariant target set X_f should be as large as possible for performance
  - Choose the parameters Q and R freely to affect the control performance
  - Adjust P as per the stability theorem
  - Useful toolbox (MATLAB): https://www.mpt3.org/
- In practice, sometimes choosing a good terminal cost is enough (i.e., don’t need to enforce a terminal control invariant condition), though you may be sacrificing guarantees

MPC for reference tracking

- Usual cost

$$\sum_{k=0}^{N-1} \mathbf{x}_k^T Q \mathbf{x}_k + \mathbf{u}_k^T R \mathbf{u}_k$$

  does not work, as in steady state control does not need to be zero
- $\delta \mathbf{u}$- formulation: reason in terms of *control changes*

$$\mathbf{u}_k = \mathbf{u}_{k-1} + \delta \mathbf{u}_k$$

MPC for reference tracking

- The MPC problem is readily modified to

$$J_0^*(\mathbf{x}(t)) = \min_{\delta\mathbf{u}_0,\ldots,\delta\mathbf{u}_{N-1}} \sum_k \|\mathbf{y}_k - \mathbf{r}_k\|_Q^2 + \|\delta\mathbf{u}_k\|_R^2$$

$$\begin{aligned}
\text{subject to} \quad & \mathbf{x}_{k+1} = A\mathbf{x}_k + B\mathbf{u}_k, && k = 0,\ldots,N-1 \\
& \mathbf{y}_k = C\mathbf{x}_k, && k = 0,\ldots,N-1 \\
& \mathbf{x}_k \in X, \quad \mathbf{u}_k \in U, && k = 0,\ldots,N-1 \\
& \mathbf{x}_N \in X_f \\
& \mathbf{u}_k = \mathbf{u}_{k-1} + \delta\mathbf{u}_k, && k = 0,\ldots,N-1 \\
& \mathbf{x}_0 = \mathbf{x}(t), \quad \mathbf{u}_{-1} = \mathbf{u}(t-1)
\end{aligned}$$

- The control input is then $\mathbf{u}(t) = \delta\mathbf{u}_0^* + \mathbf{u}(t-1)$

Robust MPC

- We have so far not explicitly considered disturbances in constraint satisfaction
- Consider system of the form

$$\mathbf{x}_{k+1} = A\mathbf{x}_k + B\mathbf{u}_k + \mathbf{w}_k$$
$$\mathbf{w}_k \in W \quad \forall k$$

  with constraints $\mathbf{x} \in X, \mathbf{u} \in U$, and W is bounded.

- Can we guarantee stability and persistent feasibility for this system?

Robust optimal control problem

$$J_0^*(\mathbf{x}(t)) = \max_{\mathbf{w}_0,\ldots,\mathbf{w}_{N-1}} \min_{\mathbf{u}_0,\ldots,\mathbf{u}_{N-1}} p(\mathbf{x}_N) + \sum_{k=0}^{N-1} c(\mathbf{x}_k, \mathbf{u}_k)$$

subject to

$$\mathbf{x}_{k+1} = A\mathbf{x}_k + B\mathbf{u}_k + \mathbf{w}_k, \quad k = 0, \ldots, N-1$$

$$\mathbf{x}_k \in X,\ \mathbf{u}_k \in U,\ \mathbf{w}_k \in W \quad k = 0, \ldots, N-1$$

$$\mathbf{x}_N \in X_f$$

$$\mathbf{x}_0 = \mathbf{x}(t)$$

Robust MPC

- Key idea: consider forward reachable sets at each time

$$S_0(\mathbf{x}_0) = \{\mathbf{x}(0)\}$$
$$S_k(\mathbf{x}_0, \mathbf{u}_{0:k-1}) = AS_{k-1}(\mathbf{x}_0, \mathbf{u}_{0:k-2}) + B\mathbf{u}_{k-1} + W$$

All trajectories in these “tubes” must satisfy constraints.

Robust MPC

$$J_0^*(\mathbf{x}(t)) = \max_{\mathbf{w}_0,\ldots,\mathbf{w}_{N-1}} \quad \min_{\mathbf{u}_0,\ldots,\mathbf{u}_{N-1}} p(\mathbf{x}_N) + \sum_{k=0}^{N-1} c(\mathbf{x}_k, \mathbf{u}_k)$$

$$\begin{aligned} \text{subject to} \quad & \mathbf{x}_{k+1} = A\mathbf{x}_k + B\mathbf{u}_k + \mathbf{w}_k, \qquad k = 0, \ldots, N-1 \\ & S_k \in X,\ \mathbf{u}_k \in U,\ \mathbf{w}_k \in W \qquad k = 0, \ldots, N-1 \\ & S_N \in X_f \\ & \mathbf{x}_0 = \mathbf{x}(t) \end{aligned}$$

Where $p(\mathbf{x}_N)$ is *robustly stable* and X_f is *robust control invariant.*

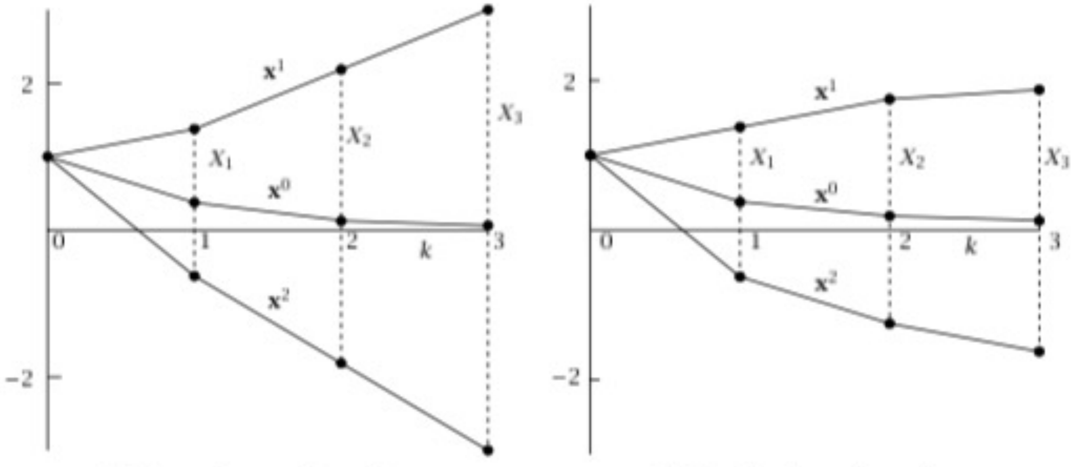
Tube MPC

- Forward tubes in robust MPC can be prohibitively large, motivating techniques to reduce their size
- Introduce nominal trajectory:

Nominal trajectory: $\bar{\mathbf{x}}_{k+1} = A\bar{\mathbf{x}}_k + B\mathbf{u}_k$

Error: $\mathbf{e}_k = \mathbf{x}_k - \bar{\mathbf{x}}_k$

Yields dynamics: $\mathbf{e}_{k+1} = A\mathbf{e}_k + \mathbf{w}_k$



(a) Open-loop trajectories.
(b) Feedback trajectories.

- Consider feedback law: $\mathbf{u}_k = \bar{\mathbf{u}}_k + F_\infty \mathbf{e}_k$

Tube MPC

- Adding error feedback gives dynamics

$$\bar{\mathbf{x}}_{k+1} = A\bar{\mathbf{x}}_k + B\bar{\mathbf{u}}_k$$
$$\mathbf{e}_{k+1} = (A + BF_\infty)\mathbf{e}_k + \mathbf{w}_k$$

Must choose $\bar{\mathbf{u}}_k$ to guarantee that $\bar{\mathbf{x}}_k + \mathbf{e}_k$ satisfy state, action, and terminal constraints for $k = 1, \ldots, N$.

What about nonlinearity?

- A very active field of research today!
- Control Barrier Functions (CBFs)
  - Analogous to Control Lyapunov Functions (CLFs) but for constraints
  - For general nonlinear dynamics $\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u})$, if we can construct a function $B(\mathbf{x})$ satisfying

$$\max_{\mathbf{u} \in U} \nabla_{\mathbf{x}} \mathrm{B}(\mathbf{x})^{\mathrm{T}} f(\mathbf{x}, \mathbf{u}) \geq -\alpha\big(B(\mathbf{x})\big)$$

    then $C := \{\mathbf{x} \in \mathbb{R}^n \mid \mathrm{B}(\mathbf{x}) \geq 0\}$ is control invariant.
- Combining CBFs for persistent feasibility, CLFs for stability, horizon $N = 1$ results in a quadratic program for control-affine systems: CLF-CBF QPs
  - Ames, et al., “Control Barrier Function Based Quadratic Programs for Safety Critical Systems,” TAC, 2017.
- In practice, guarantees of persistent feasibility or stability are often sacrificed; heuristic choices of terminal constraint, cost are employed

Next time

- Back to learning!
  Intro to learning, sys ID,
  adaptive control