Model Predictive Control

- linear convex optimal control
- finite horizon approximation
- model predictive control
- fast MPC implementations
- supply chain management

Linear time-invariant convex optimal control

$$
\begin{array}{ll}
\text{minimize} & J = \sum_{t=0}^{\infty} \ell(x(t), u(t)) \\
\text{subject to} & u(t) \in \mathcal{U}, \quad x(t) \in \mathcal{X}, \quad t = 0, 1, \ldots \\
& x(t+1) = Ax(t) + Bu(t), \quad t = 0, 1, \ldots \\
& x(0) = z.
\end{array}
$$

- variables: state and input trajectories $x(0), x(1), \ldots \in \mathbf{R}^n$, $u(0), u(1), \ldots \in \mathbf{R}^m$
- problem data:
  - dynamics and input matrices $A \in \mathbf{R}^{n \times n}$, $B \in \mathbf{R}^{n \times m}$
  - convex stage cost function $\ell : \mathbf{R}^n \times \mathbf{R}^m \to \mathbf{R}$, $\ell(0, 0) = 0$
  - convex state and input constraint sets $\mathcal{X}$, $\mathcal{U}$, with $0 \in \mathcal{X}$, $0 \in \mathcal{U}$
  - initial state $z \in \mathcal{X}$

Greedy control

- use $u(t) = \operatorname{argmin}_w \{\ell(x(t), w) \mid w \in \mathcal{U},\ Ax(t) + Bw \in \mathcal{X}\}$

- minimizes current stage cost only, ignoring effect of $u(t)$ on future, except for $x(t+1) \in \mathcal{X}$

- typically works very poorly; can lead to $J = \infty$ (when optimal u gives finite J)

‘Solution’ via dynamic programming

- (Bellman) **value function** $V(z)$ is optimal value of control problem as a function of initial state z

- can show V is convex

- V satisfies Bellman or dynamic programming equation

$$V(z) = \inf \{\ell(z, w) + V(Az + Bw) \mid w \in \mathcal{U},\ Az + Bw \in \mathcal{X}\}$$

- optimal u given by

$$u^\star(t) = \underset{w \in \mathcal{U},\ Ax(t)+Bw \in \mathcal{X}}{\operatorname{argmin}} \left(\ell(x(t), w) + V(Ax(t) + Bw)\right)$$

- intepretation: term $V(Ax(t) + Bw)$ properly accounts for future costs due to current action w

- optimal input has 'state feedback form' $u^\star(t) = \phi(x(t))$

Linear quadratic regulator

- special case of linear convex optimal control with
  - $\mathcal{U} = \mathbf{R}^m$, $\mathcal{X} = \mathbf{R}^n$
  - $\ell(x(t), u(t)) = x(t)^T Q x(t) + u(t)^T R u(t)$, $Q \succeq 0$, $R \succ 0$

- can be solved using DP
  - value function is quadratic: $V(z) = z^T P z$
  - P can be found by solving an algebraic Riccati equation (ARE)

$$P = Q + A^T P A - A^T P B (R + B^T P B)^{-1} B^T P A$$

  - optimal policy is linear state feedback: $u^\star(t) = K x(t)$, with $K = -(R + B^T P B)^{-1} B^T P A$

Finite horizon approximation

- use finite horizon T, impose terminal constraint $x(T) = 0$:

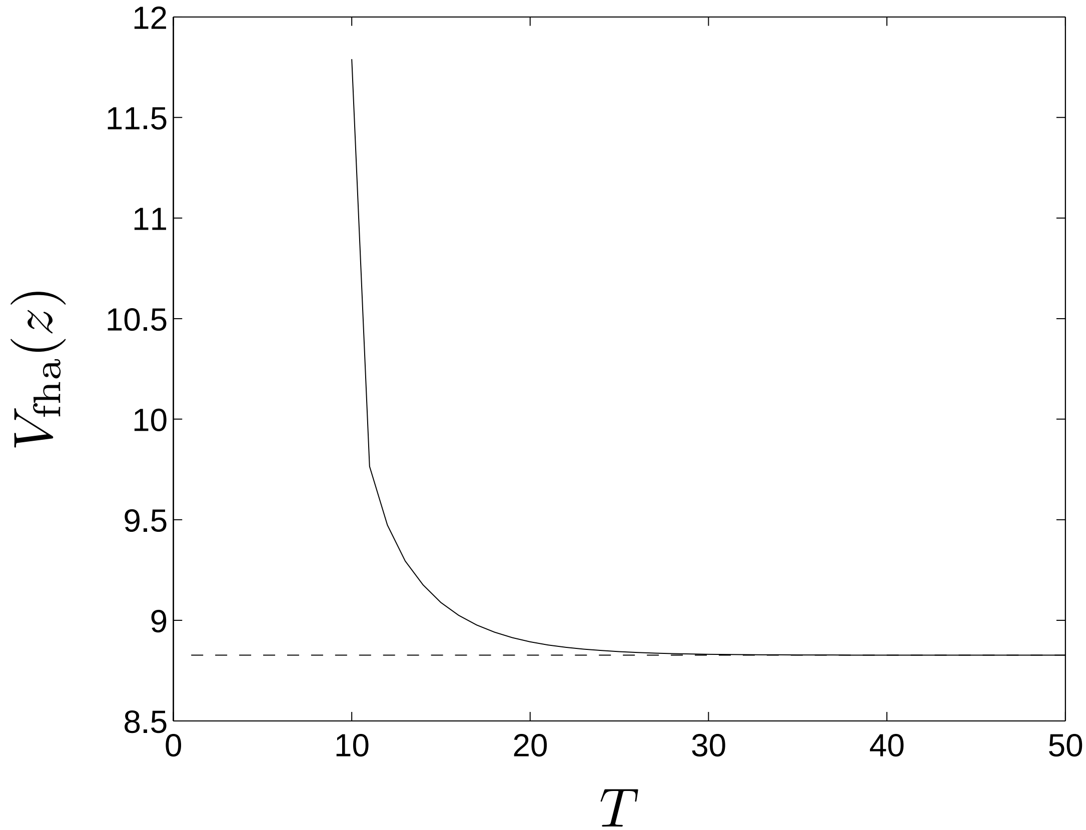
$$
\begin{array}{ll}
\text{minimize} & \sum_{\tau=0}^{T-1} \ell(x(t), u(t)) \\
\text{subject to} & u(t) \in \mathcal{U}, \quad x(t) \in \mathcal{X} \quad \tau = 0, \ldots, T \\
& x(t+1) = Ax(t) + Bu(t), \quad \tau = 0, \ldots, T-1 \\
& x(0) = z, \quad x(T) = 0.
\end{array}
$$

- apply the input sequence $u(0), \ldots, u(T-1), 0, 0, \ldots$
- a finite dimensional convex problem
- gives suboptimal input for original optimal control problem

Example

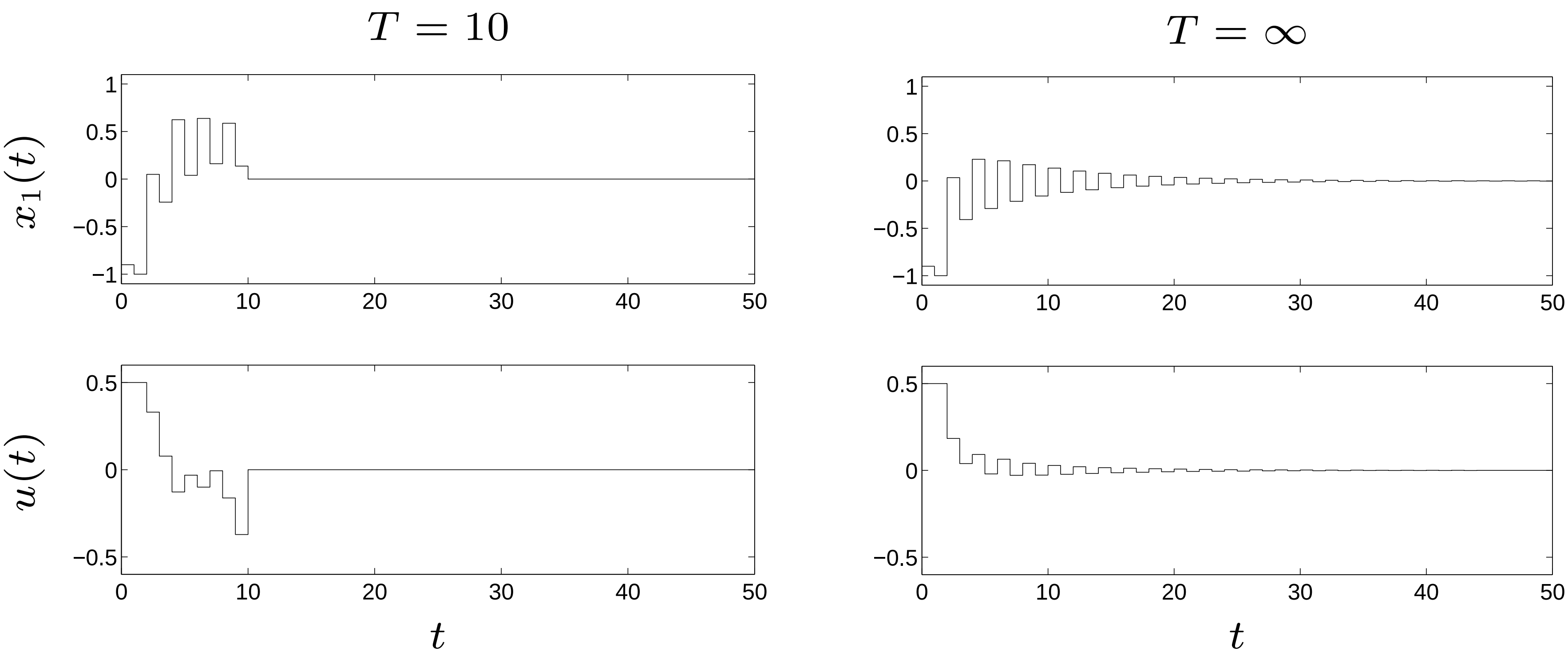
- system with $n = 3$ states, $m = 2$ inputs; A, B chosen randomly
- quadratic stage cost: $\ell(v, w) = \|v\|^2 + \|w\|^2$
- $\mathcal{X} = \{v \mid \|v\|_\infty \leq 1\}$, $\mathcal{U} = \{w \mid \|w\|_\infty \leq 0.5\}$
- initial point: $z = (0.9, -0.9, 0.9)$
- optimal cost is $V(z) = 8.83$

Cost versus horizon



dashed line shows $V(z)$; finite horizon approximation infeasible for $T \leq 9$

Trajectories



Model predictive control (MPC)

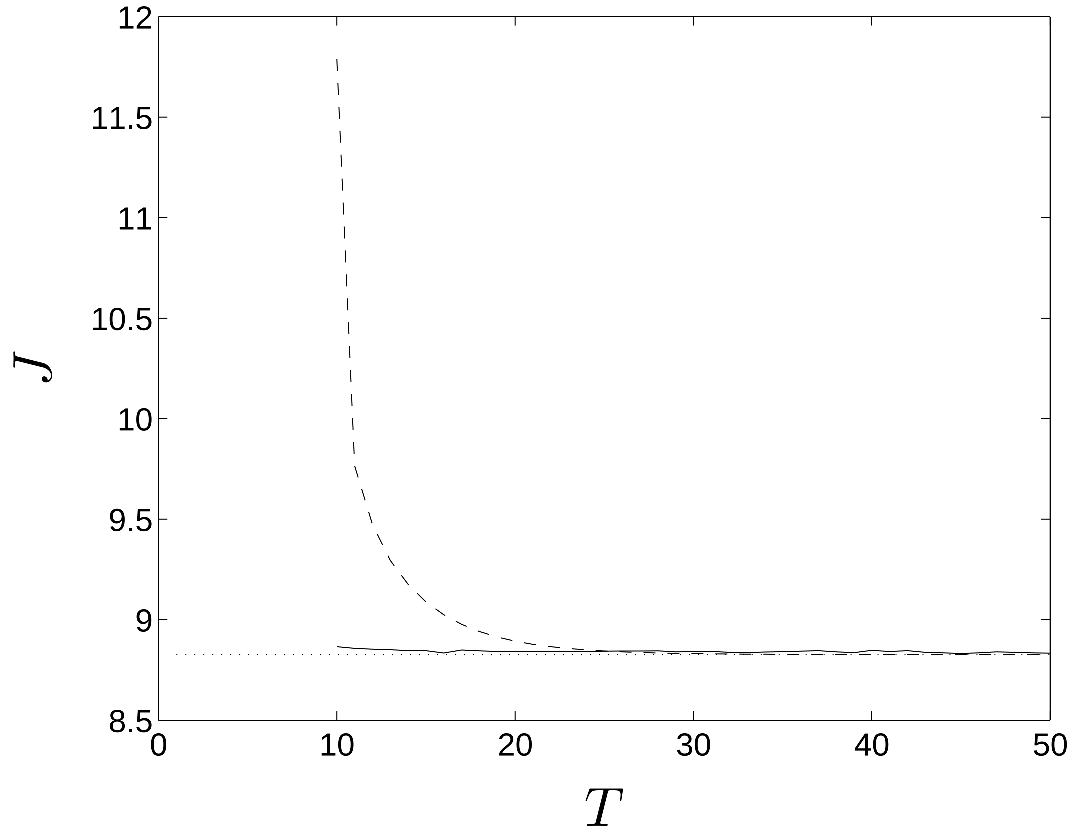
- at each time t solve the (planning) problem

$$
\begin{array}{ll}
\text{minimize} & \sum_{\tau=t}^{t+T} \ell(x(\tau), u(\tau)) \\
\text{subject to} & u(\tau) \in \mathcal{U}, \quad x(\tau) \in \mathcal{X}, \quad \tau = t, \ldots, t+T \\
& x(\tau+1) = Ax(\tau) + Bu(\tau), \quad \tau = t, \ldots, t+T-1 \\
& x(t+T) = 0
\end{array}
$$

  with variables $x(t+1), \ldots, x(t+T)$, $u(t), \ldots, u(t+T-1)$ and data $x(t)$, A, B, ℓ, $\mathcal{X}$, $\mathcal{U}$

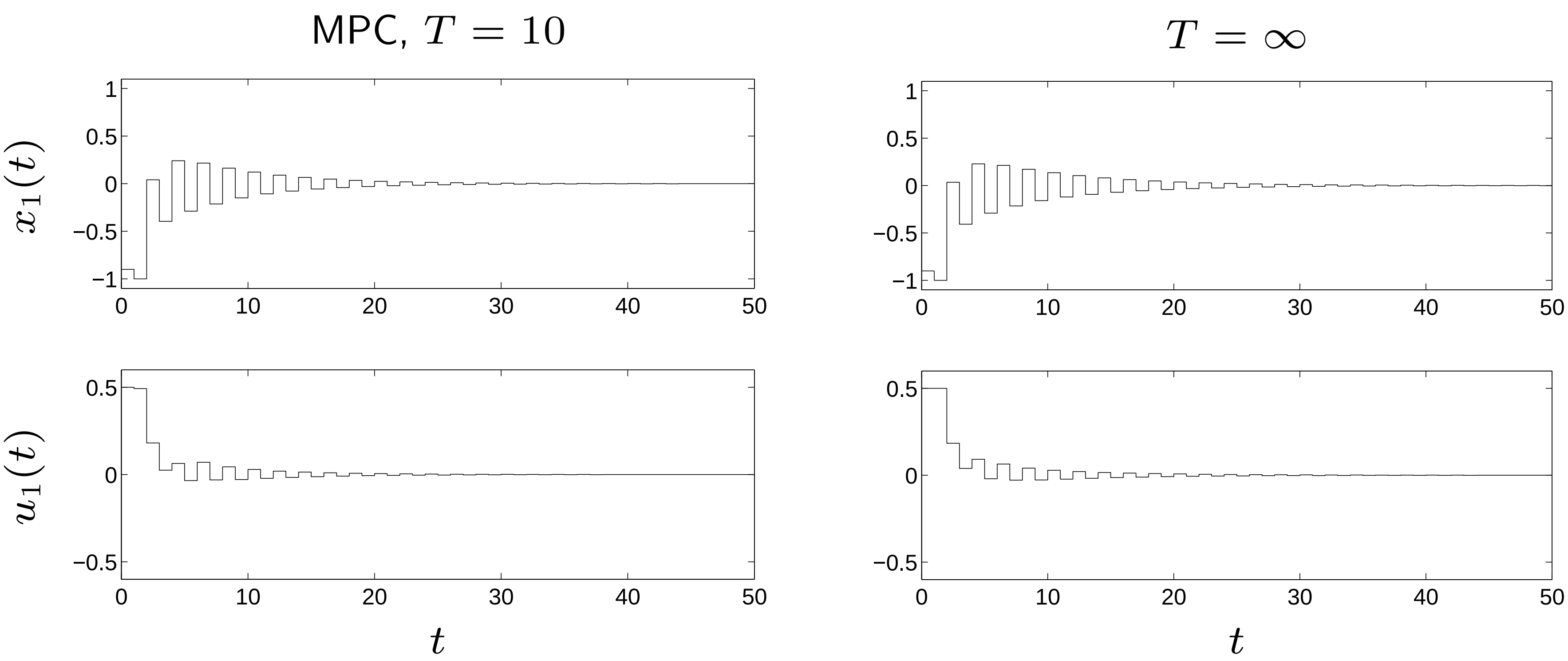
- call solution $\tilde{x}(t+1), \ldots, \tilde{x}(t+T)$, $\tilde{u}(t), \ldots, \tilde{u}(t+T-1)$
- we interpret these as *plan of action* for next T steps
- we take $u(t) = \tilde{u}(t)$
- this gives a complicated state feedback control $u(t) = \phi_{\text{mpc}}(x(t))$

MPC performance versus horizon



solid: MPC, dashed: finite horizon approximation, dotted: $V(z)$

MPC trajectories



MPC

- goes by many other names, $e.g.$, dynamic matrix control, receding horizon control, dynamic linear programming, rolling horizon planning
- widely used in (some) industries, typically for systems with slow dynamics (chemical process plants, supply chain)
- MPC typically works very well in practice, even with short T
- under some conditions, can give performance guarantees for MPC

Variations on MPC

- add final state cost $\hat{V}(x(t+T))$ instead of insisting on $x(t+T) = 0$
  - if $\hat{V} = V$, MPC gives optimal input
- convert hard constraints to violation penalties
  - avoids problem of planning problem infeasibility
- solve MPC problem every K steps, $K > 1$
  - use current plan for K steps; then re-plan

Explicit MPC

- MPC with ℓ quadratic, $\mathcal{X}$ and $\mathcal{U}$ polyhedral

- can show ϕ_{mpc} is piecewise affine

$$\phi_{\mathrm{mpc}}(z) = K_j z + g_j, \quad z \in \mathcal{R}_j$$

  $\mathcal{R}_1, \ldots, \mathcal{R}_N$ is polyhedral partition of $\mathcal{X}$

  (solution of *any* QP is PWA in righthand sides of constraints)

- ϕ_{mpc} ($i.e.$, K_j, g_j, $\mathcal{R}_j$) can be computed explicitly, off-line

- on-line controller simply evaluates $\phi_{\mathrm{mpc}}(x(t))$
  (effort is dominated by determining which region $x(t)$ lies in)

- can work well for (very) small n, m, and T
- number of regions N grows exponentially in n, m, T
  - needs lots of storage
  - evaluating $\phi_{\rm mpc}$ can be slow
- simplification methods can be used to reduce the number of regions, while still getting good control

MPC problem structure

- MPC problem is highly structured (see *Convex Optimization*, §10.3.4)
  - Hessian is block diagonal
  - equality constraint matrix is block banded
- use block elimination to compute Newton step
  - Schur complement is block tridiagonal with $n \times n$ blocks
- can solve in order $T(n+m)^3$ flops using an interior point method

Fast MPC

- can obtain further speedup by solving planning problem approximately
  - fix barrier parameter; use warm-start
  - (sharply) limit the total number of Newton steps

- results for simple C implementation

problem size			QP size		run time (ms)	
n	m	T	vars	constr	fast mpc	SDPT3
4	2	10	50	160	0.3	150
10	3	30	360	1080	4.0	1400
16	4	30	570	1680	7.7	2600
30	8	30	1110	3180	23.4	3400

- can run MPC at **kilohertz** rates

Supply chain management

- n nodes (warehouses/buffers)
- m unidirectional links between nodes, external world
- $x_i(t)$ is amount of commodity at node i, in period t
- $u_j(t)$ is amount of commodity transported along link j
- incoming and outgoing node incidence matrices:

$$A_{ij}^{\mathrm{in(out)}} = \begin{cases} 1 & \text{link } j \text{ enters (exits) node } i \\ 0 & \text{otherwise} \end{cases}$$

- dynamics: $x(t+1) = x(t) + A^{\mathrm{in}}u(t) - A^{\mathrm{out}}u(t)$

Constraints and objective

- buffer limits: $0 \leq x_i(t) \leq x_{\max}$
  (could allow $x_i(t) < 0$, to represent back-order)

- link capacities: $0 \leq u_i(t) \leq u_{\max}$

- $A^{\text{out}} u(t) \preceq x(t)$ (can't ship out what's not on hand)

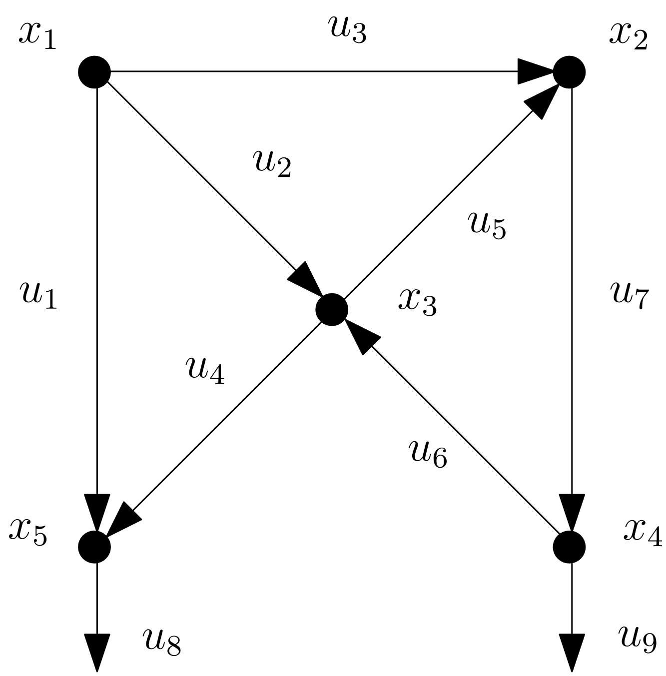
- shipping/transportation cost: $S(u(t))$
  (can also include sales revenue or manufacturing cost)

- warehousing/storage cost: $W(x(t))$

- objective: $\sum_{t=0}^{\infty} \left(S(u(t)) + W(x(t))\right)$

Example

- $n = 5$ nodes, $m = 9$ links (links 8, 9 are external links)



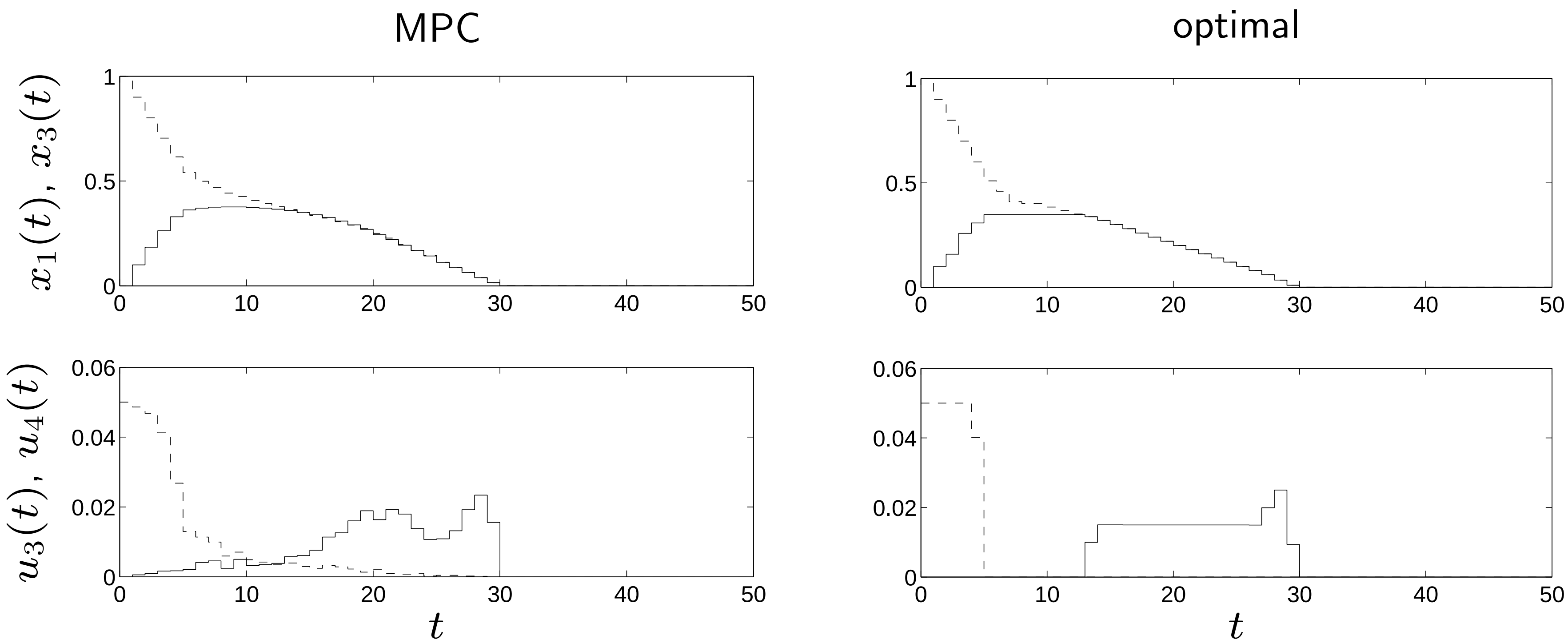
Example

- $x_{\max} = 1$, $u_{\max} = 0.05$
- storage cost: $W(x(t)) = \sum_{i=0}^{n}(x_i(t) + x_i(t)^2)$
- shipping cost:

$$S(u(t)) = \underbrace{u_1(t) + \cdots + u_7(t)}_{\text{transportation cost}} - \underbrace{(u_8(t) + u_9(t))}_{\text{revenue}}$$

- initial stock: $x(0) = (1, 0, 0, 1, 1)$
- we run MPC with $T = 5$, final cost $\hat{V}(x(t+T)) = 10(\mathbf{1}^T x(t+T))$
- optimal cost: $V(z) = 68.2$; MPC cost 69.5

MPC and optimal trajectories



solid: $x_3(t)$, $u_4(t)$; dashed: $x_1(t)$, $u_3(t)$

Variations on optimal control problem

- time varying costs, dynamics, constraints
  - discounted cost
  - convergence to nonzero desired state
  - tracking time-varying desired trajectory

- coupled state and input constraints, *e.g.*, $(x(t), u(t)) \in \mathcal{P}$ (as in supply chain management)

- slew rate constraints, *e.g.*, $\|u(t+1) - u(t)\|_\infty \leq \Delta u_{\max}$

- stochastic control: future costs, dynamics, disturbances not known (next lecture)