EE365: Model Predictive Control

Certainty-equivalent control

Constrained linear-quadratic regulator

Infinite horizon model predictive control

MPC with disturbance prediction

Certainty-equivalent control

Stochastic control

- dynamics $x_{t+1} = f_t(x_t, u_t, w_t)$, $t = 0, \ldots, T-1$
- $x_t \in \mathcal{X}$, $u_t \in \mathcal{U}$, $w_t \in \mathcal{W}$
- $x_0, w_0, \ldots, w_{T-1}$ independent
- stage cost $g_t(x_t, u_t)$; terminal cost $g_T(x_T)$
- state feedback policy $u_t = \mu_t(x_t)$, $t = 0, \ldots, T-1$
- stochastic control problem: choose policy to minimize

$$J = \mathbf{E}\left(\sum_{t=0}^{T-1} g_t(x_t, u_t) + g_T(x_T)\right)$$

Stochastic control

- can solve stochastic control problem in some cases
  - $\mathcal{X}$, $\mathcal{U}$, $\mathcal{W}$ finite (and as a practical matter, not too big)
  - $\mathcal{X}$, $\mathcal{U}$, $\mathcal{W}$ finite dimensional vector spaces, f_t affine, g_t convex quadratic
  - and a few other special cases
- in other situations, must resort to heuristics, suboptimal policies

Certainty-equivalent control

- a simple (usually) suboptimal policy
- replace each w_t with some predicted, likely, or typical value $\hat{w}_t$
- stochastic control problem reduces to deterministic control problem, called **certainty-equivalent problem**
- **certainty-equivalent policy** is optimal policy for certainty-equivalent problem
- useful when we can't solve stochastic problem, but we can solve deterministic problem
- sounds unsophisticated, but can work very well in some cases
- also called **model predictive control** (MPC) (for reasons we'll see later)

Where $\hat{w}_t$ comes from

- most likely value: choose $\hat{w}_t$ as value of w_t with maximum probability
- a random sample of w_t (yes, really)
- a nominal value
- a prediction of w_t (more on this later)
- when w_t is a number or vector: $\hat{w}_t = \mathbf{E}\, w_t$, rounded to be in $\mathcal{W}_t$

Optimal versus CE policy via dynamic programming

- optimal policy: $v_T^\star(x) = g_T(x)$; for $t = T-1, \ldots, 0$,

$$v_t^\star(x) = \min_u \left(g_t(x,u) + \mathbf{E}\, v_{t+1}^\star(f_t(x,u,w_t))\right)$$
$$\mu_t^\star(x) \in \operatorname*{argmin}_u \left(g_t(x,u) + \mathbf{E}\, v_{t+1}^\star(f_t(x,u,w_t))\right)$$

- CE policy: $v_T^{\rm ce}(x) = g_T(x)$; for $t = T-1, \ldots, 0$,

$$v_t^{\rm ce}(x) = \min_u \left(g_t(x,u) + v_{t+1}^{\rm ce}(f_t(x,u,\hat{w}_t))\right)$$
$$\mu_t^{\rm ce}(x) \in \operatorname*{argmin}_u \left(g_t(x,u) + v_{t+1}^{\rm ce}(f_t(x,u,\hat{w}_t))\right)$$

Computing CE policy via optimization

- CE policy μ^{ce} is typically not computed via DP (if you could do this, why not use DP to compute optimal policy?)
- instead we *evaluate* $\mu_t^{\mathrm{ce}}(x)$ by solving a deterministic control (optimization) problem

$$\begin{array}{ll} \text{minimize} & \sum_{\tau=t}^{T-1} g_\tau(x_\tau, u_\tau) + g_T(x_T) \\ \text{subject to} & x_{\tau+1} = f_\tau(x_\tau, u_\tau, \hat{w}_\tau), \quad \tau = t, \ldots, T-1 \\ & x_t = x \end{array}$$

  with variables $x_t, \ldots, x_T, u_t, \ldots, u_{T-1}$
  - find a solution $\bar{x}_t, \ldots, \bar{x}_T, \bar{u}_t, \ldots, \bar{u}_{T-1}$
  - then $\mu_t^{\mathrm{ce}}(x) = \bar{u}_t$ (and optimal value of problem above is $v_t^{\mathrm{ce}}(x)$)
- we don't have a formula for μ_t^{ce} (or v_t^{ce}) but we can compute $\mu_t^{\mathrm{ce}}(x)$ ($v_t^{\mathrm{ce}}(x)$) for any given x by solving an optimization problem

Certainty-equivalent control

- need to solve a (deterministic) optimal control problem in each step, with a given initial state
- these problems become shorter (smaller) as t increases toward T
- call solution of optimization problem at time t

$$\bar{x}_{t|t}, \ldots, \bar{x}_{T|t}, \quad \bar{u}_{t|t}, \ldots, \bar{u}_{T|t}$$

  - interpret as **plan of future action** at time t
    (based on assumption that disturbances take values $\hat{w}_t, \ldots, \hat{w}_{T-1}$)
  - solving problem above is **planning**
  - CE control executes first step in plan of action
  - once new state is determined, update plan

Example: Multi-queue serving

- N queues with capacity C: state is $q_t \in \{0, \dots, C\}^N$
- observe random arrivals w_t from some known distribution
- can serve up to S queues in each time period:

$$u_t \in \{0,1\}^N, \quad u_t \leq q_t, \quad \mathbf{1}^T u_t \leq S$$

- dynamics $q_{t+1} = (q_t - u_t + w_t)_{[0,C]}$
- stage cost

$$g_t(q_t, u_t, w_t) = \underbrace{\alpha^T q_t + \beta^T q_t^2}_{\text{queue cost}} + \underbrace{\gamma^T (q_t - u_t + w_t - C)_+}_{\text{rejection cost}}$$

- terminal cost $g^T(q_T) = \lambda^T q_T$

Example: Multi-queue serving

consider example with

- $N = 5$ queues, $C = 3$ capacity, $S = 2$ servers, horizon $T = 10$
- $|\mathcal{X}| = 1024$, $|\mathcal{U}| = 16$, $|\mathcal{W}| = 32$
- $w_t^{(i)} \sim \mathrm{Bernoulli}(p_i)$
- (randomly chosen) parameters:

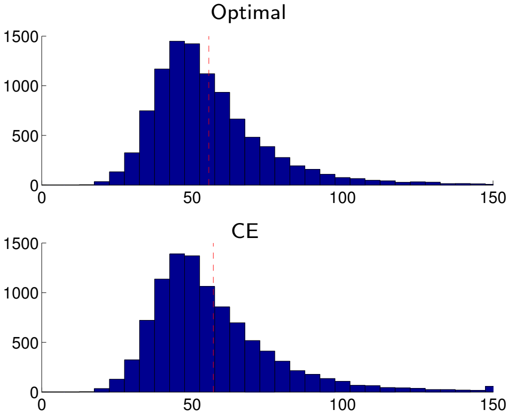
$$
\begin{aligned}
p &= \left(\; 0.47, \quad 0.17, \quad 0.25, \quad 0.21, \quad 0.60 \;\right) \\
\alpha &= \left(\; 1.32, \quad 0.11, \quad 0.63, \quad 1.41, \quad 1.83 \;\right) \\
\beta &= \left(\; 0.98, \quad 2.95, \quad 0.16, \quad 2.12, \quad 2.59 \;\right) \\
\gamma &= \left(\; 0.95, \quad 4.23, \quad 7.12, \quad 9.27, \quad 0.82 \;\right) \\
\lambda &= \left(\; 0.57, \quad 1.03, \quad 0.24, \quad 0.74, \quad 2.11 \;\right)
\end{aligned}
$$

Example: Multi-queue serving

- use deterministic values $\hat{w}_t = (1, 0, 0, 0, 1)$, $t = 0, \ldots, T-1$
- other choices lead to similar results (more later)
- problem is small enough that we can solve it exactly (for comparison)

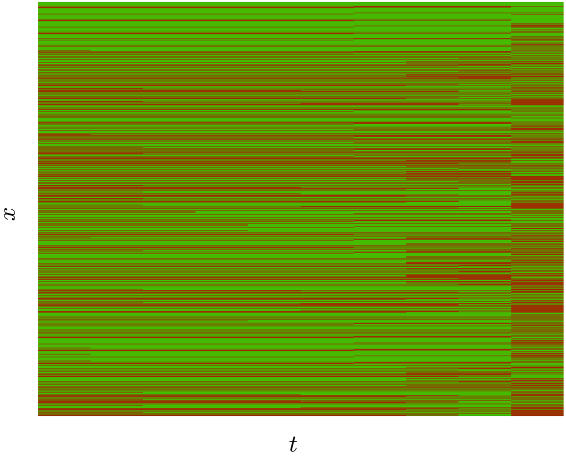
Example: Multi-queue serving

- 10000 Monte Carlo simulations with optimal and CE policies
- $J^\star = 55.55$, $J^{\rm ce} = 57.04$ (very nearly optimal!)



Example: Multi-queue serving

- red indicates $\mu^{\text{ce}}(x) \neq \mu^\star(x)$; policies differ in 37.91% of entries



Example: Multi-queue serving

- with (reasonable) different assumed values, such as $\hat{w}_t = (0, 0, 0, 0, 1)$, get different policies, also nearly optimal
- interpretation: CE policies work well because
  - there are many good (nearly optimal) policies
  - the CE policy takes into account the dynamics, stage costs
- there is no need to use CE policy when (as in this example) we can just as well compute the optimal policy

Constrained linear-quadratic regulator

Linear-quadratic regulator (LQR)

- $\mathcal{X} = \mathbb{R}^n$, $\mathcal{U} = \mathbb{R}^m$
- $x_{t+1} = Ax_t + Bu_t + w_t$
- $x_0, w_0, w_1, \ldots$ independent zero mean, $\mathbf{E}\, x_0 x_0^T = X_0$, $\mathbf{E}\, w_t w_t^T = W_t$
- cost (with $Q_t \geq 0$, $R_t > 0$)

$$J = (1/2) \sum_{t=0}^{T-1} \left(x_t^T Q_t x_t + u_t^T R_t u_t \right) + (1/2) x_T^T Q_T x_T$$

- can solve exactly, since $v_t^\star$ is quadratic, $\mu_t^\star$ is linear
- can compute $J^\star$ exactly

CE for LQR

- use $\hat{w}_t = \mathbf{E}\, w_t = 0$ (*i.e.*, neglect disturbance)
- **for LQR, CE policy is actually optimal**
  - in LQR lecture we saw that optimal policy doesn't depend on W
  - choice $W = 0$ corresponds to deterministic problems in CE
- another hint that CE isn't as dumb as it might first appear
- when $\mathbf{E}\, w_t \neq 0$, CE policy is not optimal

Constrained LQR

- same as LQR, but replace $\mathcal{U} = \mathbb{R}^m$ with $\mathcal{U} = [-1, 1]^m$
- *i.e.*, constrain control inputs to $[-1, 1]$ ('actuator limits')
- cannot practically compute (or even represent) $v_t^\star$, $\mu_t^\star$
- we don't know optimal value $J^\star$

CE for constrained linear-quadratic regulator

- CE policy usually called MPC for constrained LQR
- use $\hat{w}_t = \mathbf{E}\, w_t = 0$
- evaluate $\mu_t^{\mathrm{ce}}(x)$ by solving (convex) *quadratic program* (QP)

$$
\begin{array}{ll}
\text{minimize} & (1/2)\sum_{\tau=t}^{T-1}\left(x_\tau^T Q_\tau x_\tau + u_\tau^T R_\tau u_\tau\right) + (1/2)x_T^T Q_T x_T \\
\text{subject to} & x_{\tau+1} = Ax_\tau + Bu_\tau, \quad \tau = t,\ldots,T-1 \\
& x_\tau \in \mathbb{R}^n, \quad u_\tau \in [-1,1]^m \quad \tau = t,\ldots,T-1 \\
& x_t = x
\end{array}
$$

  with variables $x_t,\ldots,x_T, u_t,\ldots,u_{T-1}$

  - find solution $\bar{x}_t,\ldots,\bar{x}_T, \bar{u}_t,\ldots,\bar{u}_{T-1}$
  - execute first step in plan: $\mu_t^{\mathrm{mpc}}(x) = \bar{u}_t$
- these QPs can be solved **super fast** (*e.g.*, in microseconds)

Example

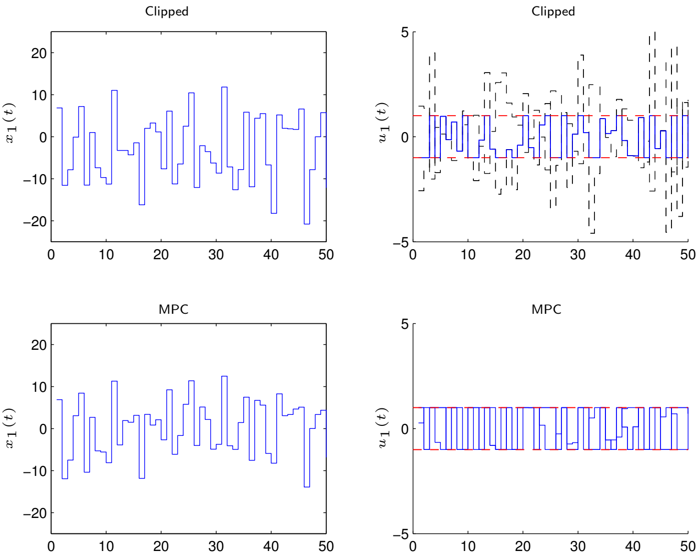
consider example with

- $n = 8$ states, $m = 2$ inputs, horizon $T = 50$
- A, B chosen randomly, A scaled so $\max_i |\lambda_i(A)| = 1$
- $X = 3I$, $W = 1.5I$
- $Q_t = I$, $R_t = I$
- associated (unconstrained) LQR problem has
    - $\|u\|_\infty > 1$ often
    - $J^{\text{lqr}} = 85$ (a lower bound on J^{lqr} for constrained LQR problem)

Example

- $\mu_t^{\text{clip}}(x) = (K_t^{\text{lqr}} x)_{[-1,1]}$ ('saturated LQR control')
  - yields performance $J^{\text{clip}} = 1641.8$
- MPC policy $\mu_t^{\text{mpc}}(x)$
  - yields performance $J^{\text{mpc}} = 1135.3$
- we don't know $J^\star$ (other than $J^\star > J^{\text{lqr}} = 85$)
- sophisticated lower bounding techniques can show J^{mpc} very near $J^\star$

Sample traces



Infinite horizon model predictive control

Infinite horizon MPC

- want approximate policy for infinite horizon average (or total) cost stochastic control problem
- replace w_t with some typical value $\hat{w}$ (usually constant)
- in most cases, cannot solve resulting infinite horizon deterministic control problem
- instead, solve the deterministic problem over a **rolling horizon** (or **planning horizon**) from current time t to $t + T$

Infinite horizon MPC

- to evaluate $\mu^{\text{mpc}}(x)$, solve optimization problem

$$\begin{array}{ll} \text{minimize} & \sum_{\tau=t}^{t+T-1} g(x_\tau, u_\tau) + g^{\text{eoh}}(x_{t+T}) \\ \text{subject to} & x_{\tau+1} = f(x_\tau, u_\tau, \hat{w}), \quad \tau = t, \ldots, t+T-1 \\ & x_t = x \end{array}$$

  with variables $x_t, \ldots, x_{t+T}, u_t, \ldots, u_{t+T-1}$

  - find a solution $\bar{x}_t, \ldots, \bar{x}_{t+T}, \bar{u}_t, \ldots, \bar{u}_{t+T-1}$
  - then $u_t^{\text{mpc}}(x_t) = \bar{u}_t$

- g^{eoh} is an end-of-horizon cost
- these optimization problems have the same size (*cf.* finite horizon MPC)

Infinite horizon MPC

- design parameters in MPC policy:
  - disturbance predictions $\hat{w}_t$ (typically constant)
  - horizon length T
  - end-of-horizon cost $g^{\rm eoh}$
- some common choices: $g^{\rm eoh}(x) = 0$, $g^{\rm eoh}(x) = \min_u g(x,u)$
- performance of MPC policy evaluated by Monte Carlo simulation
- for T large enough, particular value of T and choice of $g^{\rm eoh}$ shouldn't affect performance very much

Example: Supply chain management

- n nodes (warehouses/buffers)
- $x_t \in \mathbb{R}^n$ is amount of commodity at nodes at time t
- m unidirectional links between nodes, external world
- $u_t \in \mathbb{R}^m$ is amount of commodity transported along links at time t
- incoming and outgoing node incidence matrix:

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

  (include wholesale supply links and retail delivery links)
- dynamics: $x_{t+1} = x_t + A^{\text{in}} u_t - A^{\text{out}} u_t$

Example: Supply chain management

- buffer limits: $0 \le x_t \le x_{\max}$
- warehousing/storage cost: $W(x_t) = \alpha^T x_t + \beta^T x_t^2$, $\alpha,\ \beta \ge 0$
- link capacities: $0 \le u_t \le u_{\max}$
- $A^{\text{out}} u_t \le x_t$ (can't ship out what's not on hand)

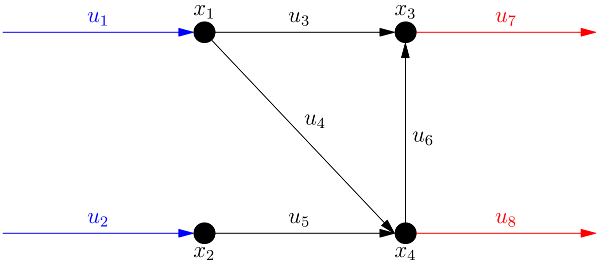
Example: Supply chain management

- shipping/transportation cost: $S(u_t) = S_1((u_t)_1) + \cdots + S_n((u_t)_m)$
- for internode link, $S_i((u_t)_i) = \gamma (u_t)_i$ is transportation cost
- for wholesale supply link, $S_i((u_t)_i) = (p_t^{\rm wh})_i (u_t)_i$ is purchase cost
- for retail delivery link, $S_i((u_t)_i) = -p^{\rm ret} \min\{(d_t)_i, (u_t)_i\}$ is the negative retail revenue, where $p^{\rm ret}$ is retail price and $(d_t)_i$ is the demand
- we assume wholesale prices $(p_t^{\rm wh})_i$ are IID, demands $(d_t)_i$ are IID
- link flows u_t chosen as function of x_t, $p_t^{\rm wh}$, d_t
- objective: minimize average stage cost

$$J = \lim_{T \to \infty} \frac{1}{T} \sum_{t=0}^{T} (S(u_t) + W(x_t))$$

Example

- $n = 4$ nodes, $m = 8$ links
- links 1,2 are wholesale supply; links 7,8 are retail delivery



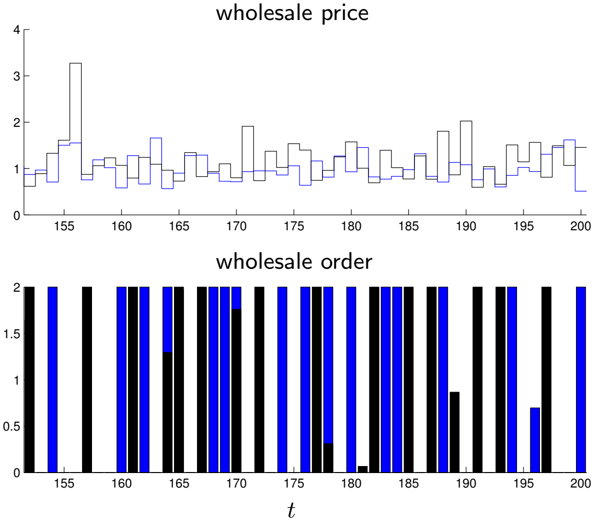
Example

- buffer capacities $x_{\max} = 3$
- link flow capacities $u_{\max} = 2$
- storage cost parameters $\alpha = \beta = 0.01$; $\gamma = 0.05$
- wholesale prices are log-normal with means 1, 1.2; variances 0.1, 0.2
- demands are log-normal with means 1, 0.8; variances , 0.4, 0.2
- retail price is $p^{\mathrm{ret}} = 2$

Example

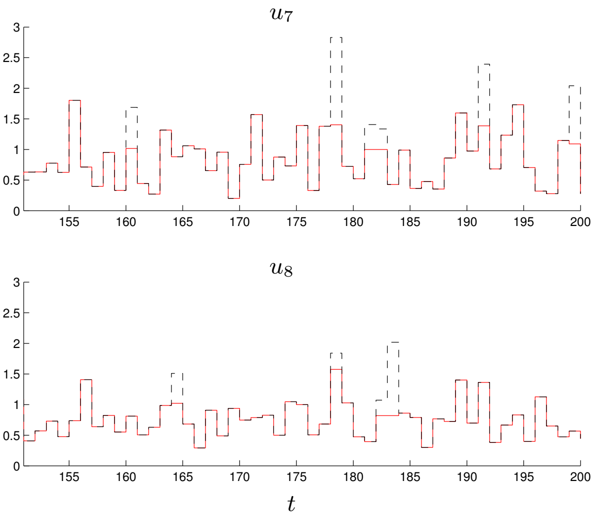
- MPC parameters:
  - future wholesale prices and retail demands assumed equal to their means (current wholesale prices and demands are known)
  - horizon $T = 30$
  - end-of-horizon cost $g^{\text{eoh}} = 0$
- MPC problem is QP (and readily solved)
- results: average cost $J = -1.69$
  - wholesale purchase cost 1.20
  - retail delivery income -3.16
- lower bounding techniques for similar problems suggests MPC is very nearly optimal

MPC sample trajectory: supply



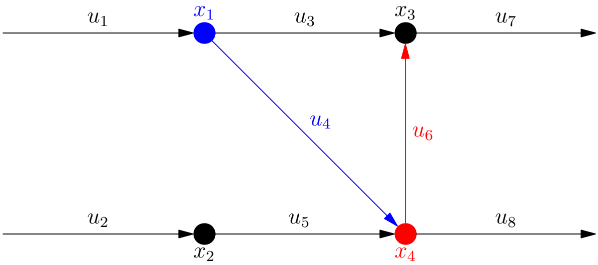
line: $(p_t^{\mathrm{wh}})_1$, $(p_t^{\mathrm{wh}})_2$; bar: u_1, u_2

MPC sample trajectory: delivery

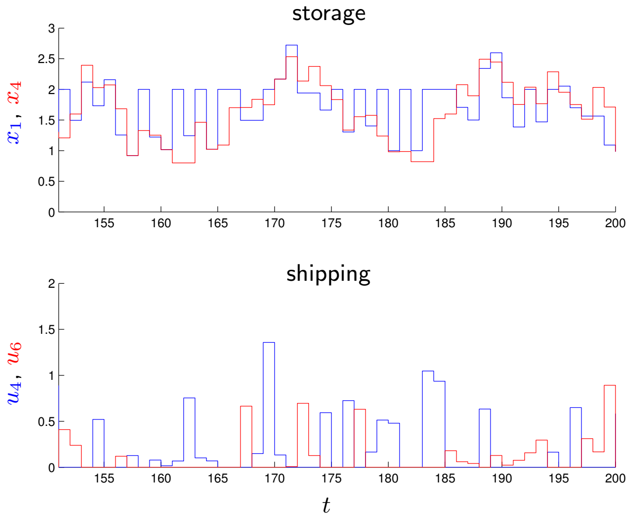


solid: delivery; dashed: demand

MPC sample trajectory



MPC sample trajectory



MPC with disturbance prediction

Rolling disturbance estimates

- in MPC, we interpret $\hat{w}_t$ as predictions of disturbance values
- no need to assume they are independent, or even random variables
- when w_t are not independent (or interpreted as random variables), additional information can improve predictions $\hat{w}_t$
- we let $\hat{w}_{t|s}$ denote the **updated estimate** of w_t made at time s using all information available at time s
- these are called **rolling estimates** of w_t
- $\hat{w}_{t|s}$ can come from a statistical model, experts' predictions, ...
- MPC with rolling disturbance prediction works very well in practice, is used in many applications (supply chain, finance)

MPC architecture

- MPC (rolling horizon, with updated predictions) splits into two components
    - the **predictor** uses all information available to make predictions of current and future values of w_t
    - the **planner** optimizes actions over a planning horizon that extends into the future, assuming the predictions are correct
- the MPC action is simply the current action in the current plan
- MPC is not optimal except in a few special cases
- but it often performs extremely well