Πιθανοτική Ανάλυση Ενεργειακών Συστημάτων

Καθ. Νικόλαος Χατζηαργυρίου

Τεχνικές Βελτιστοποίησης

Περιεχόμενα

- Εισαγωγή – Το πρόβλημα της OPF
- Ορισμοί και βασικές αρχές
- Στοχαστικός προγραμματισμός
- Robust and chance-constrained optimisation
- (Stochastic) Model Predictive Control

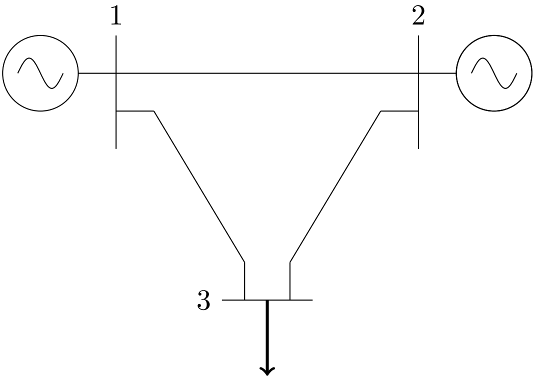
Πηγές

- Spyros Chatzivasileiadis, Associate Professor at DTU (*http://www.chatziva.com*)
  - Course 31765: Optimization in Modern Power Systems
- Stephen P. Boyd, Professor at Stanford University (*web.stanford.edu/~boyd*)
  - Course EE364a: Convex Optimization I
  - Course EE364b - Convex Optimization II
  - Book: Convex Optimization, Stephen Boyd and Lieven Vandenberghe

What is optimization?

source: Spyros Chatzivaseiliadis D

Economic Dispatch and Optimal Power Flow: Short Introduction on the Board



Economic Dispatch

$$\min \sum_i c_i P_{G_i}$$

subject to:

$$P_{G_i}^{min} \leq P_{G_i} \leq P_{G_i}^{max}$$

and

$$\sum_i P_{G_i} = P_D$$

Power Balance!

Economic Dispatch

$$\min \sum_i c_i P_{G_i}$$

subject to:

$$P_{G_i}^{min} \leq P_{G_i} \leq P_{G_i}^{max}$$

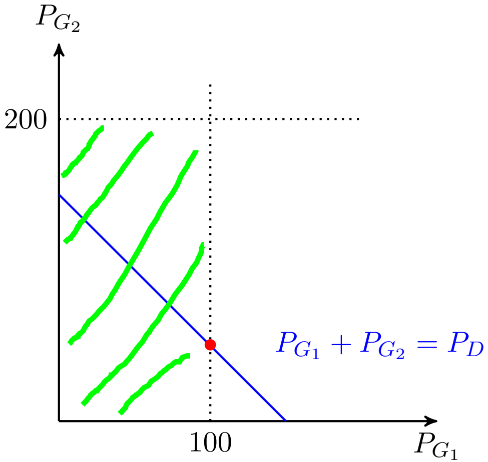
and

$$\sum_i P_{G_i} = P_D$$

How do you interpret these constraints for a 2-generator system on the cartesian plane?

Let's visualise things!!!

Graphical representation of the 2-generator Economic Dispatch



Assumptions

$c_1 < c_2$

$P_D = 150$ MW

$0 \leq P_{G_1} \leq 100$ MW

$0 \leq P_{G_2} \leq 200$ MW

DC-OPF

DC -> no reactive power, voltage magn. =1

$$\min \sum_i c_i P_{G_i}$$

subject to:

$$P_{G_i}^{min} \leq P_{G_i} \leq P_{G_i}^{max}$$

and

$$\mathbf{B} \cdot \delta = \mathbf{P_G} - \mathbf{P_D}$$

Power balance is repla

and

$$\frac{1}{x_{ij}}(\delta_i - \delta_j) \leq P_{ij,max}$$

Mathematical optimization

(mathematical) optimization problem

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq b_i, \quad i = 1, \ldots, m \end{array}$$

- $x = (x_1, \ldots, x_n)$: optimization variables
- $f_0 : \mathbf{R}^n \to \mathbf{R}$: objective function
- $f_i : \mathbf{R}^n \to \mathbf{R}$, $i = 1, \ldots, m$: constraint functions

solution or **optimal point** $x^\star$ has smallest value of f_0 among all vectors that satisfy the constraints

Examples

portfolio optimization

- variables: amounts invested in different assets
- constraints: budget, max./min. investment per asset, minimum return
- objective: overall risk or return variance

device sizing in electronic circuits

- variables: device widths and lengths
- constraints: manufacturing limits, timing requirements, maximum area
- objective: power consumption

data fitting

- variables: model parameters
- constraints: prior information, parameter limits
- objective: measure of misfit or prediction error, plus regularization term

Solving optimization problems

general optimization problem

- very difficult to solve
- methods involve some compromise, $e.g.$, very long computation time, or not always finding the solution (which may not matter in practice)

exceptions: certain problem classes can be solved efficiently and reliably

- least-squares problems
- linear programming problems
- convex optimization problems

Least-squares

$$\text{minimize} \quad \|Ax - b\|_2^2$$

You might have seen this as y=px and we are looking for the

solving least-squares problems

setting derivative equal to zero as we look for the minimum!

- analytical solution: $x^\star = (A^T A)^{-1} A^T b$

or p = (y^T x) / (x^T x)

- reliable and efficient algorithms and software
- computation time proportional to $n^2 k$ ($A \in \mathbf{R}^{k \times n}$); less if structured
- a mature technology

using least-squares

- least-squares problems are easy to recognize
- a few standard techniques increase flexibility ($e.g.$, including weights, adding regularization terms)

Linear programming

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & a_i^T x \leq b_i, \quad i = 1, \ldots, m \end{array}$$

solving linear programs

- no analytical formula for solution
- reliable and efficient algorithms and software

e.g Simplex

- computation time proportional to $n^2 m$ if $m \geq n$; less with structure
- a mature technology

using linear programming

- not as easy to recognize as least-squares problems
- a few standard tricks used to convert problems into linear programs ($e.g.$, problems involving ℓ_1- or ℓ_∞-norms, piecewise-linear functions)

Convex optimization problem

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq b_i, \quad i = 1, \ldots, m \end{array}$$

- objective and constraint functions are convex:

$$f_i(\alpha x + \beta y) \leq \alpha f_i(x) + \beta f_i(y)$$

  if $\alpha + \beta = 1$, $\alpha \geq 0$, $\beta \geq 0$

- includes least-squares problems and linear programs as special cases

solving convex optimization problems

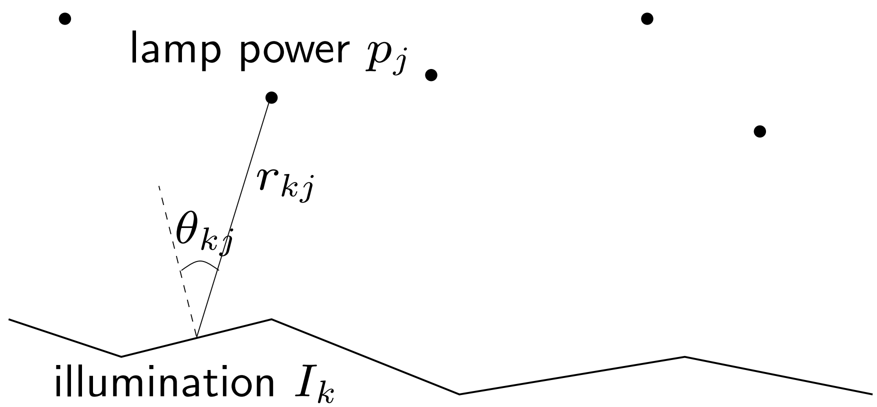
- no analytical solution
- reliable and efficient algorithms
- computation time (roughly) proportional to $\max\{n^3, n^2m, F\}$, where F is cost of evaluating f_i's and their first and second derivatives
- almost a technology

using convex optimization

- often difficult to recognize
- many tricks for transforming problems into convex form
- surprisingly many problems can be solved via convex optimization

Example

m lamps illuminating n (small, flat) patches



intensity I_k at patch k depends linearly on lamp powers p_j:

$$I_k = \sum_{j=1}^{m} a_{kj} p_j, \qquad a_{kj} = r_{kj}^{-2} \max\{\cos\theta_{kj}, 0\}$$

problem: achieve desired illumination I_{des} with bounded lamp powers

$$\begin{array}{ll} \text{minimize} & \max_{k=1,\ldots,n} |\log I_k - \log I_{\mathrm{des}}| \\ \text{subject to} & 0 \le p_j \le p_{\max}, \quad j = 1, \ldots, m \end{array}$$

how to solve?

1. use uniform power: $p_j = p$, vary p
2. use least-squares:

$$\text{minimize} \quad \sum_{k=1}^{n} (I_k - I_{\text{des}})^2$$

   round p_j if $p_j > p_{\text{max}}$ or $p_j < 0$

3. use weighted least-squares:

$$\text{minimize} \quad \sum_{k=1}^{n} (I_k - I_{\text{des}})^2 + \sum_{j=1}^{m} w_j (p_j - p_{\text{max}}/2)^2$$

   iteratively adjust weights w_j until $0 \le p_j \le p_{\text{max}}$

4. use linear programming:

$$\begin{array}{ll} \text{minimize} & \max_{k=1,\ldots,n} |I_k - I_{\text{des}}| \\ \text{subject to} & 0 \le p_j \le p_{\text{max}}, \quad j = 1, \ldots, m \end{array}$$

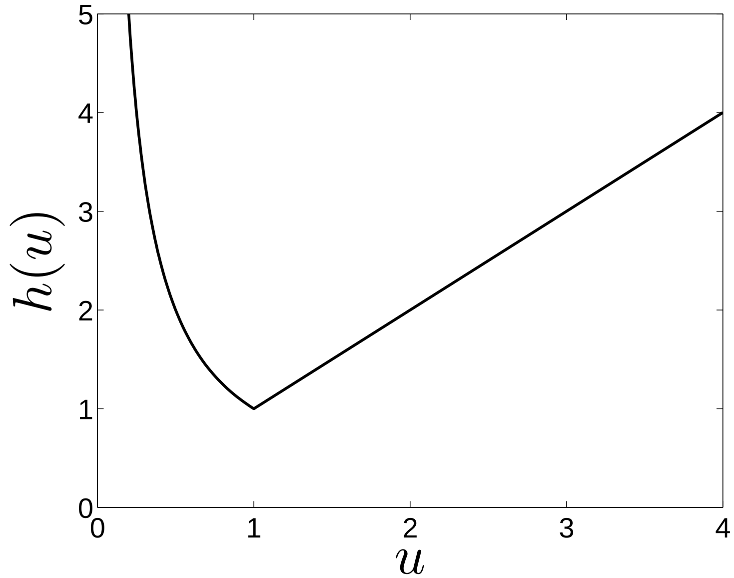
   which can be solved via linear programming

of course these are approximate (suboptimal) 'solutions'

5. use convex optimization: problem is equivalent to

$$\begin{array}{ll}\text{minimize} & f_0(p) = \max_{k=1,\ldots,n} h(I_k/I_{\mathsf{des}}) \\ \text{subject to} & 0 \leq p_j \leq p_{\mathsf{max}}, \quad j = 1,\ldots,m\end{array}$$

with $h(u) = \max\{u, 1/u\}$



f_0 is convex because maximum of convex functions is convex

exact solution obtained with effort $\approx$ modest factor $\times$ least-squares effort

additional constraints: does adding 1 or 2 below complicate the problem?

1. no more than half of total power is in any 10 lamps
2. no more than half of the lamps are on ($p_j > 0$)

p_i <= 0.5 sum p_i

u need binaries for that!

- answer: with (1), still easy to solve; with (2), extremely difficult
- moral: (untrained) intuition doesn't always work; without the proper background very easy problems can appear quite similar to very difficult problems

Stochastic programming

- stochastic programming
- 'certainty equivalent' problem
- violation/shortfall constraints and penalties
- Monte Carlo sampling methods
- validation

sources: Nemirovsky & Shapiro

Stochastic programming

- objective and constraint functions $f_i(x, \omega)$ depend on optimization variable x *and* a random variable ω

- ω models
  - parameter variation and uncertainty
  - random variation in implementation, manufacture, operation

- value of ω is not known, but its distribution is

- goal: choose x so that
  - constraints are satisfied on average, or with high probability
  - objective is small on average, or with high probability

Stochastic programming

- basic stochastic programming problem:

$$\begin{array}{ll} \text{minimize} & F_0(x) = \mathbf{E}\, f_0(x,\omega) \\ \text{subject to} & F_i(x) = \mathbf{E}\, f_i(x,\omega) \leq 0, \quad i = 1,\ldots,m \end{array}$$

  - variable is x
  - problem data are f_i, distribution of ω

- if $f_i(x,\omega)$ are convex in x for each ω
  - F_i are convex
  - hence stochastic programming problem is convex

- F_i have analytical expressions in only a few cases;
in other cases we will solve the problem approximately

It's not easy to go from the distributin of omega to the distribution of Fo, or Fi

Example with analytic form for F_i

- $f(x) = \|Ax - b\|_2^2$, with A, b random
- $F(x) = \mathbf{E}\, f(x) = x^T P x - 2q^T x + r$, where

$$P = \mathbf{E}(A^T A), \quad q = \mathbf{E}(A^T b), \quad r = \mathbf{E}(\|b\|_2^2)$$

- only need second moments of (A, b)
- stochastic constraint $\mathbf{E}\, f(x) \le 0$ can be expressed as standard quadratic inequality

remember? because it is a square...

'Certainty-equivalent' problem

- 'certainty-equivalent' (a.k.a. 'mean field') problem:

$$\begin{array}{ll} \text{minimize} & f_0(x, \mathbf{E}\,\omega) \\ \text{subject to} & f_i(x, \mathbf{E}\,\omega) \leq 0, \quad i = 1, \ldots, m \end{array}$$

- roughly speaking: ignore parameter variation

- if f_i convex in ω for each x, then
  - $f_i(x, \mathbf{E}\,\omega) \leq \mathbf{E}\, f_i(x, \omega)$
  - so optimal value of certainty-equivalent problem is lower bound on optimal value of stochastic problem

Solving stochastic programming problems

- analytical solution in special cases, *e.g.*, when expectations can be found analytically
  - ω enters quadratically in f_i
  - ω takes on finitely many values
- general case: approximate solution via (Monte Carlo) sampling

Run Monte Carlo -> Make scenarios -> run deterministic opt. for each scenario

Monte Carlo sampling method

- a general method for (approximately) solving stochastic programming problem

- generate N samples (realizations) $\omega_1, \dots, \omega_N$, with associated probabilities $\pi_1, \dots, \pi_N$ (usually $\pi_j = 1/N$)

- form sample average approximations

Here, is the case where we seek to minimize the average

$$\hat{F}_i(x) = \sum_{j=1}^{N} \pi_j f_i(x, \omega_j), \quad i = 0, \dots, m$$

- these are RVs (via $\omega_1, \dots, \omega_N$) with mean $\mathbf{E}\, f_i(x, \omega) = F_i(x)$

- now solve finite event problem

$$\begin{array}{ll} \text{minimize} & \hat{F}_0(x) \\ \text{subject to} & \hat{F}_i(x) \leq 0, \quad i = 1, \ldots, m \end{array}$$

- solution $x^\star_{\mathrm{mcs}}$ and optimal value $\hat{F}_0(x^\star_{\mathrm{mcs}})$ are random variables (hopefully close to $x^\star$ and $p^\star$, optimal value of original problem)

- theory says
  - (with some technical conditions) as $N \to \infty$, $x^\star_{\mathrm{mcs}} \to x^\star$
  - $\mathbf{E}\, \hat{F}_0(x^\star_{\mathrm{mcs}}) \leq p^\star$

Out-of-sample validation

- a practical method to check if N is 'large enough'
- use a second set of samples ('validation set') $\omega_1^{\text{val}}, \ldots, \omega_M^{\text{val}}$, with probabilities $\pi_1^{\text{val}}, \ldots, \pi_M^{\text{val}}$ (usually $M \gg N$)

  (original set of samples called 'training set')
- evaluate

use your solution form before to see if it works with real data

$$\hat{F}_i^{\text{val}}(x_{\text{mcs}}^\star) = \sum_{j=1}^{M} \pi_j^{\text{val}} f_i(x_{\text{mcs}}^\star, \omega_j^{\text{val}}), \quad i = 0, \ldots, m$$

- if $\hat{F}_i(x_{\text{mcs}}^\star) \approx \hat{F}_i^{\text{val}}(x_{\text{mcs}}^\star)$, our confidence that $x_{\text{mcs}}^\star \approx x^\star$ is enhanced
- if not, increase N and re-compute $x_{\text{mcs}}^\star$

Example

- we consider problem

$$\begin{array}{ll} \text{minimize} & F_0(x) = \mathbf{E} \max_i (Ax + b)_i \\ \text{subject to} & F_1(x) = \mathbf{E} \max_i (Cx + d)_i \leq 0 \end{array}$$

  with optimization variable $x \in \mathbf{R}^n$

  $A \in \mathbf{R}^{m \times n}$, $b \in \mathbf{R}^m$, $C \in \mathbf{R}^{k \times n}$, $d \in \mathbf{R}^k$ are random

- we consider instance with $n = 10$, $m = 20$, $k = 5$

- certainty-equivalent optimal value yields lower bound 19.1

- we use Monte Carlo sampling with $N = 10,\ 100,\ 1000$

- validation set uses $M = 10000$

	$N = 10$	$N = 100$	$N = 1000$
F_0 (training)	51.8	54.0	55.4
F_0 (validation)	56.0	54.8	55.2
F_1 (training)	0	0	0
F_1 (validation)	1.3	0.7	-0.03

we conclude:

- $N = 10$ is too few samples
- $N = 100$ is better, but not enough
- $N = 1000$ is probably fine

Robust Optimization

- definitions of robust optimization
- robust linear programs
- robust cone programs
- chance constraints

Robust optimization

convex objective $f_0 : \mathbf{R}^n \to \mathbf{R}$, *uncertainty set* $\mathcal{U}$, and $f_i : \mathbf{R}^n \times \mathcal{U} \to \mathbf{R}$,

$$x \mapsto f_i(x, u) \text{ convex for all } u \in \mathcal{U}$$

general form

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x, u) \leq 0 \text{ for all } u \in \mathcal{U}, i = 1, \ldots, m. \end{array}$$

equivalent to

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & \sup_{u \in \mathcal{U}} f_i(x, u) \leq 0, i = 1, \ldots, m. \end{array}$$

We want the max of the constraint realization to be within limits!

- Bertsimas, Ben-Tal, El-Ghaoui, Nemirovski (1990s–now)

Setting up robust problem

if we want the same for the objective function we do:

- can always replace objective f_0 with $\sup_{u \in \mathcal{U}} f_0(x,u)$, rewrite in epigraph form to

$$\begin{array}{ll} \text{minimize} & t \\ \text{subject to} & \sup_u f_0(x,u) \leq t, \sup_u f_i(x,u) \leq 0, \;\; i = 1,\ldots,m \end{array}$$

- equality constraints make no sense: a robust equality $a^T(x+u) = b$ for all $u \in \mathcal{U}$?

three questions:

- is robust formulation useful?
- is robust formulation computable?
- how should we choose $\mathcal{U}$?

Chance constrained optimization

- chance constraints and percentile optimization
- chance constraints for log-concave distributions
- convex approximation of chance constraints

sources: Rockafellar & Uryasev, Nemirovsky & Shapiro

Chance constraints and percentile optimization

- 'chance constraints' (η is 'confidence level'):

$$\mathbf{Prob}(f_i(x, \omega) \leq 0) \geq \eta$$

  - convex in some cases (later)
  - generally interested in $\eta = 0.9,\ 0.95,\ 0.99$
  - $\eta = 0.999$ meaningless (unless you're sure about the distribution tails)

- percentile optimization (γ is 'η-percentile'):

How to do the same for the objective function!

$$\begin{array}{ll} \text{minimize} & \gamma \\ \text{subject to} & \mathbf{Prob}(f_0(x, \omega) \leq \gamma) \geq \eta \end{array}$$

  - convex or quasi-convex in some cases (later)

Value-at-risk and conditional value-at-risk

- value-at-risk of random variable z, at level η:

$$\mathbf{VaR}(z;\eta) = \inf\{\gamma \mid \mathbf{Prob}(z \leq \gamma) \geq \eta\}$$

  - chance constraint $\mathbf{Prob}(f_i(x,\omega) \leq 0) \geq \eta$ same as $\mathbf{VaR}(f_i(x,\omega);\eta) \leq 0$

Gamma: i timi tis sunartiseis f_i gia tin opoia to h% twn allwn timwn einai mikroteres tis

In other words, VaR is the worst-case within the 95% percentile

Sto stoch. optimization mas noiazei giati mporoume na fixaroume to gamma kai na paiksoume

- conditional value-at-risk:

or Expected Shortfall -> what happens to f_i beyond gamma (at the tails!) -> it is sort of the weighted average of what happens there!

$$\mathbf{CVaR}(z;\eta) = \inf_{\beta}\left(\beta + 1/(1-\eta)\,\mathbf{E}(z-\beta)_+\right)$$

  - $\mathbf{CVaR}(z;\eta) \geq \mathbf{VaR}(z;\eta)$ (more on this later)

Model Predictive Control

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

Model predictive control (MPC)

Provlima elegxou pou exei mesa ena provlima veltistopoiisis

- at each time t solve the (planning) problem

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

x: state variableu: decision variable

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

T: horizon

- we take $u(t) = \tilde{u}(t)$
- this gives a complicated state feedback control $u(t) = \phi_{\text{mpc}}(x(t))$

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

Stochastic Model Predictive Control

- stochastic finite horizon control
- stochastic dynamic programming
- certainty equivalent model predictive control

Certainty equivalent model predictive control

- at every time t we solve the certainty equivalent problem

$$\begin{array}{ll} \text{minimize} & \sum_{\tau=t}^{T-1} \ell_t(x_\tau, u_\tau) + \ell_T(x_T) \\ \text{subject to} & u_\tau \in \mathcal{U}_\tau, \quad \tau = t, \ldots, T-1 \\ & x_{\tau+1} = Ax_\tau + Bu_\tau + \hat{w}_{\tau|t}, \quad \tau = t, \ldots, T-1 \end{array}$$

  with variables $x_{t+1}, \ldots, x_T$, $u_t, \ldots, u_{T-1}$ and data x_t, $\hat{w}_{t|t}, \ldots, \hat{w}_{T-1|t}$

- $\hat{w}_{t|t}, \ldots, \hat{w}_{T-1|t}$ are predicted values of $w_t, \ldots, w_{T-1}$ based on X_t ($e.g.$, conditional expectations)

- call solution $\tilde{x}_{t+1}, \ldots, \tilde{x}_T$, $\tilde{u}_t, \ldots, \tilde{u}_{T-1}$

- we take $\phi^{\text{mpc}}(X_t) = \tilde{u}_t$

  – ϕ^{mpc} is a function of X_t since $\hat{w}_{t|t}, \ldots, \hat{w}_{T-1|t}$ are functions of X_t

Certainty equivalent model predictive control

- widely used, *e.g.*, in 'revenue management'
- based on (bad) approximations:
  - future values of disturbance are exactly as predicted; there is no future uncertainty
  - in future, no recourse is available
- yet, often works very well