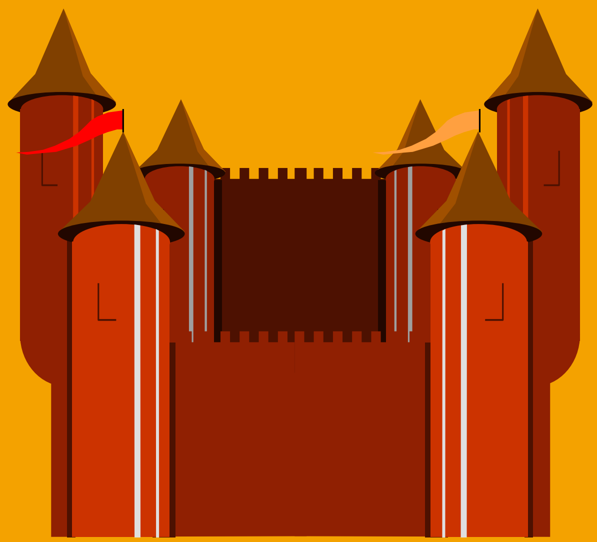
Energy and Uncertainty: Navigating the Jungle of Stochastic Optimization

CompSust 2012

July 4, 2012

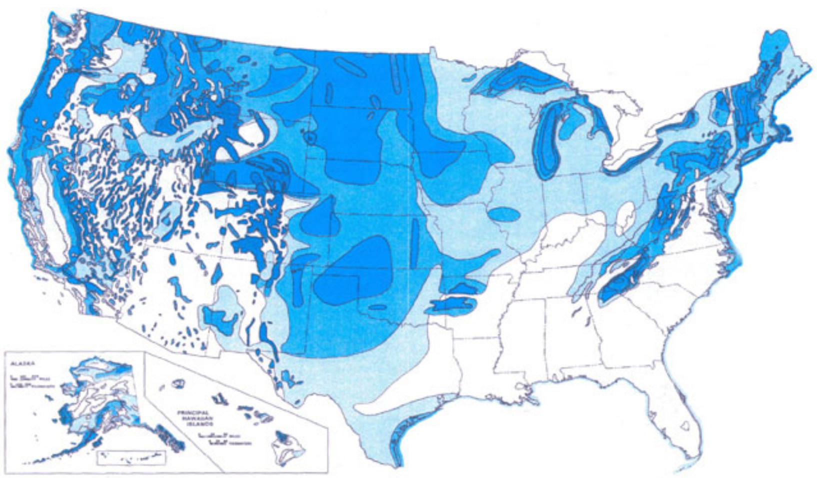


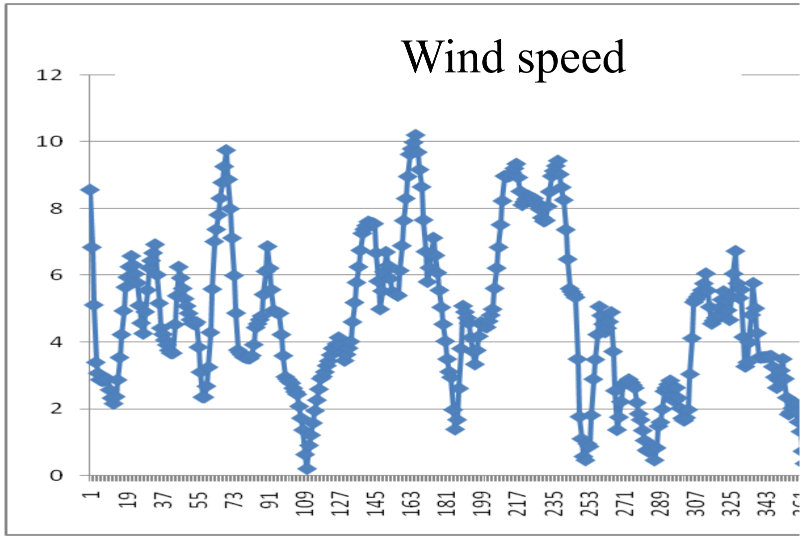
Warren B. Powell
PENSA Laboratory
Princeton University
http://www.castlelab.princeton.edu

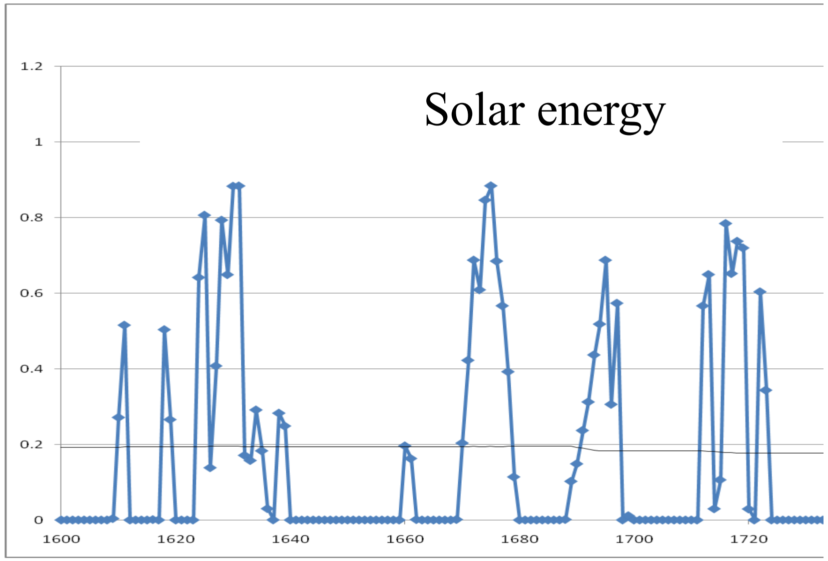
The PENSA team

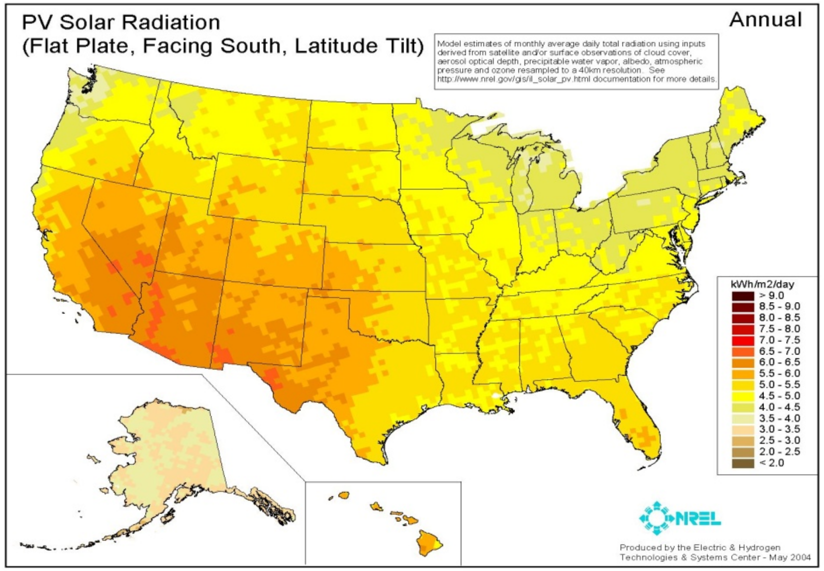
- Faculty
  - » Warren Powell (Director)
  - » Ronnie Sircar (ORFE)
  - » Craig Arnold (MAE)
  - » Rob Socolow (MAE)
  - » … (growing list)
- Staff/post-docs
  - » Hugo Simao (deputy director)
  - » Boris Defourny
  - » Arta Jamshidi
  - » Ricardo Collado
  - » Somayeh Moazeni
  - » Javad Khazaei
- Graduate students
  - » Warren Scott (ORFE)
  - » Ethan Fang (ORFE)
  - » DH Lee (COS)
  - » Daniel Salas (CBE)
  - » Jinzhen Jin (CEE)
  - » Dan Jiang (ORFE)
  - » Jae Ho Kim (EE)
- Undergraduate interns (2012)
  - » Tarun Sinha (MAE)
  - » Stephen Wang (ORFE)
  - » Henry Chai (ORFE)
  - » Ryan Peng (ORFE)
  - » Christine Feng (ORFE)
  - » Joe Yan (ORFE)
  - » Austin Wang (ORFE)

Intermittent energy sources



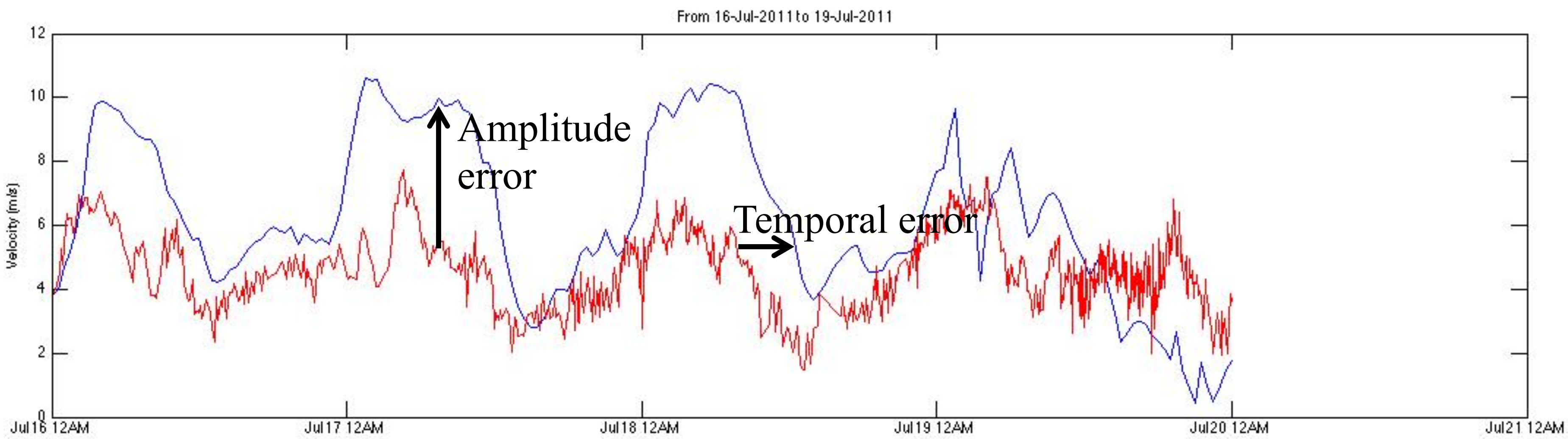






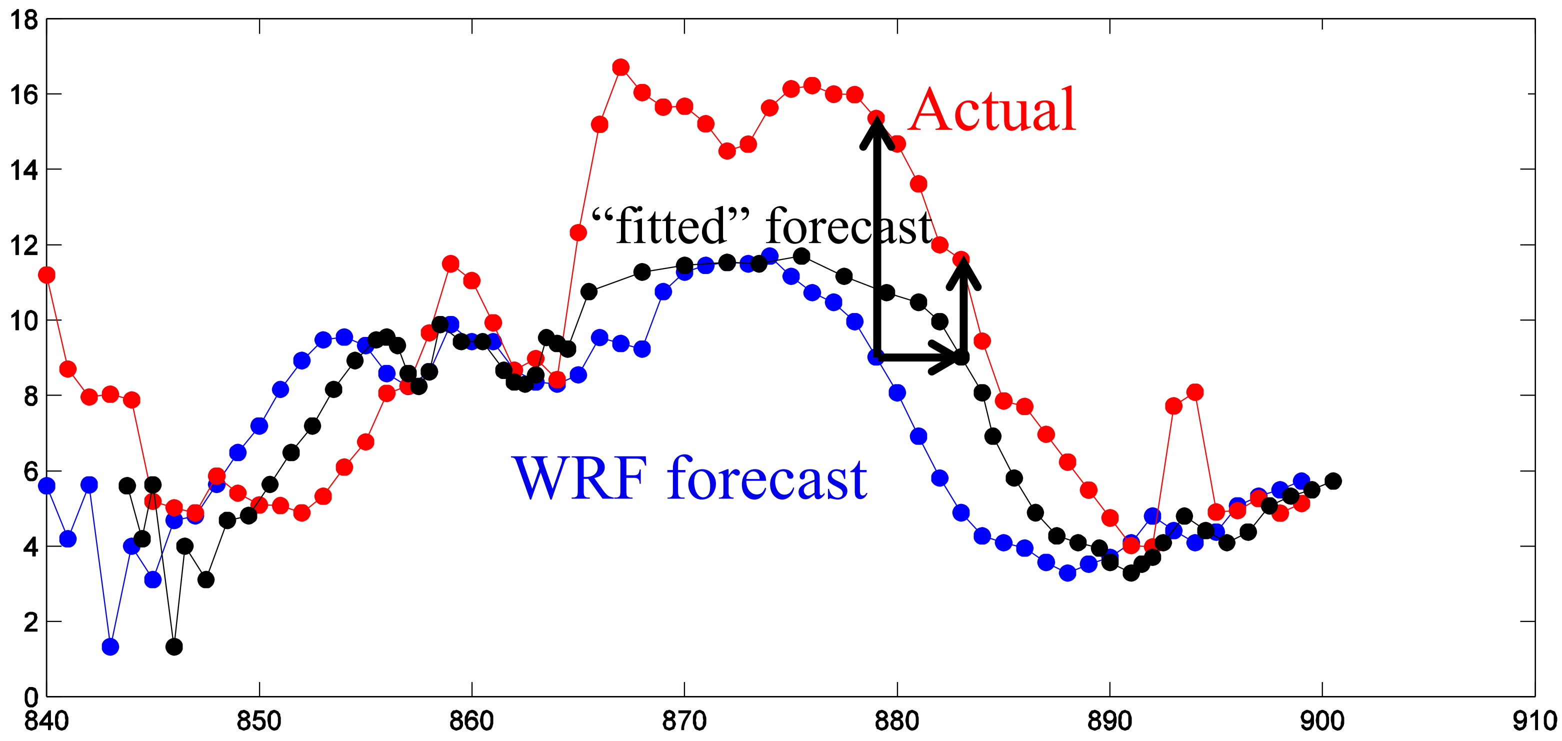
Modeling wind forecast errors

- We need a mathematical model of the stochastic process describing errors in wind forecast
  - » We are using the "WRF" model to predict wind. WRF is a sophisticated meteorological model that can predict shifts in weather patterns.
  - » We need to separate amplitude errors (how much wind at a point in time) from temporal errors (errors in the timing of a weather shift).



Modeling wind forecast errors

- We "fit" a forecast by optimizing temporal shifts
  - » Nonlinear cost function penalizes amplitude and penalty shifts
  - » Additional penalty for *changes* in shifts
  - » Optimized "fit" obtained by solving a dynamic program. State variable = (shift of previous point, change in two previous shifts)

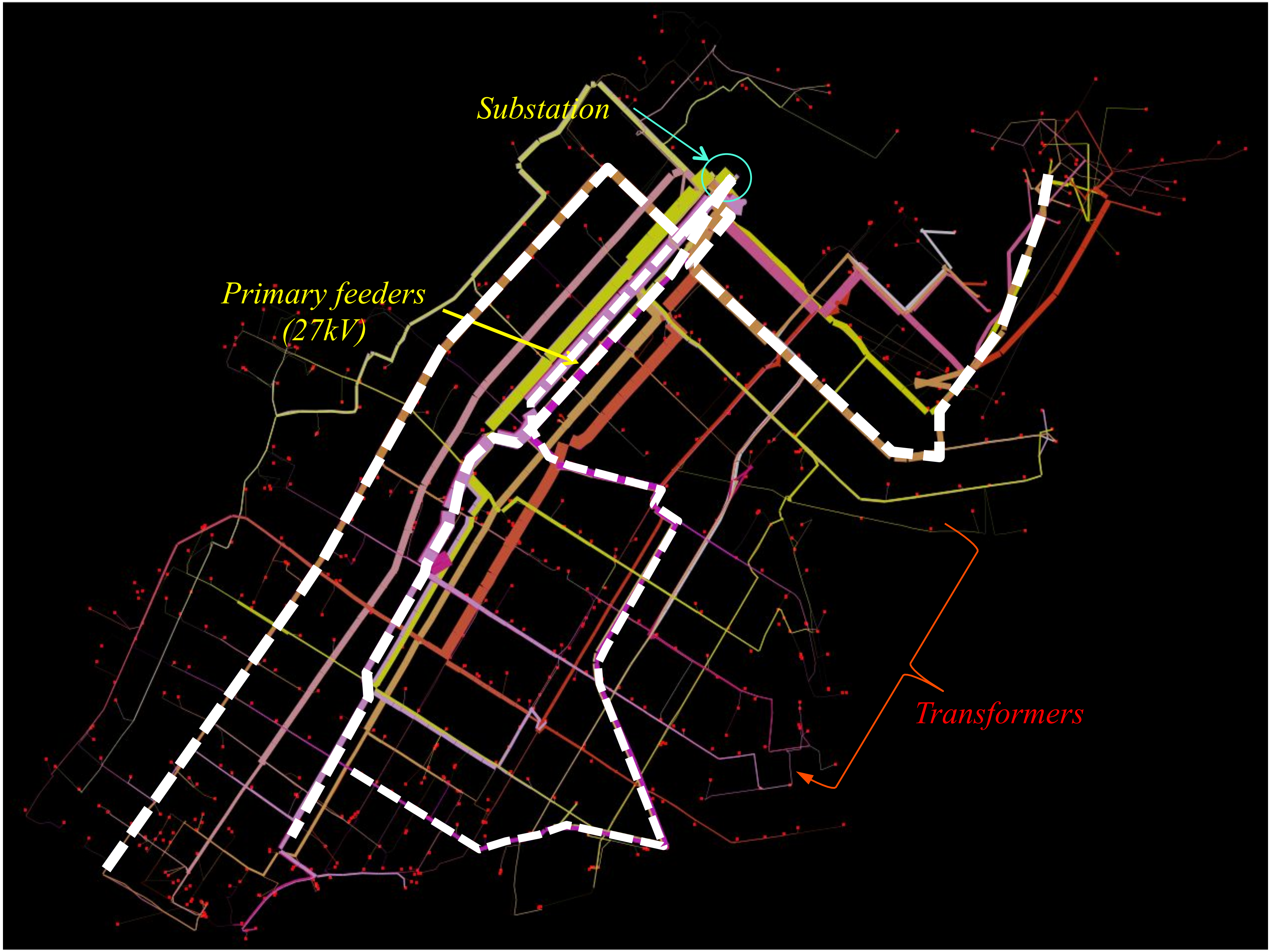


COLUMBIA CENTER FOR COMPUTATIONAL LEARNING SYSTEMS

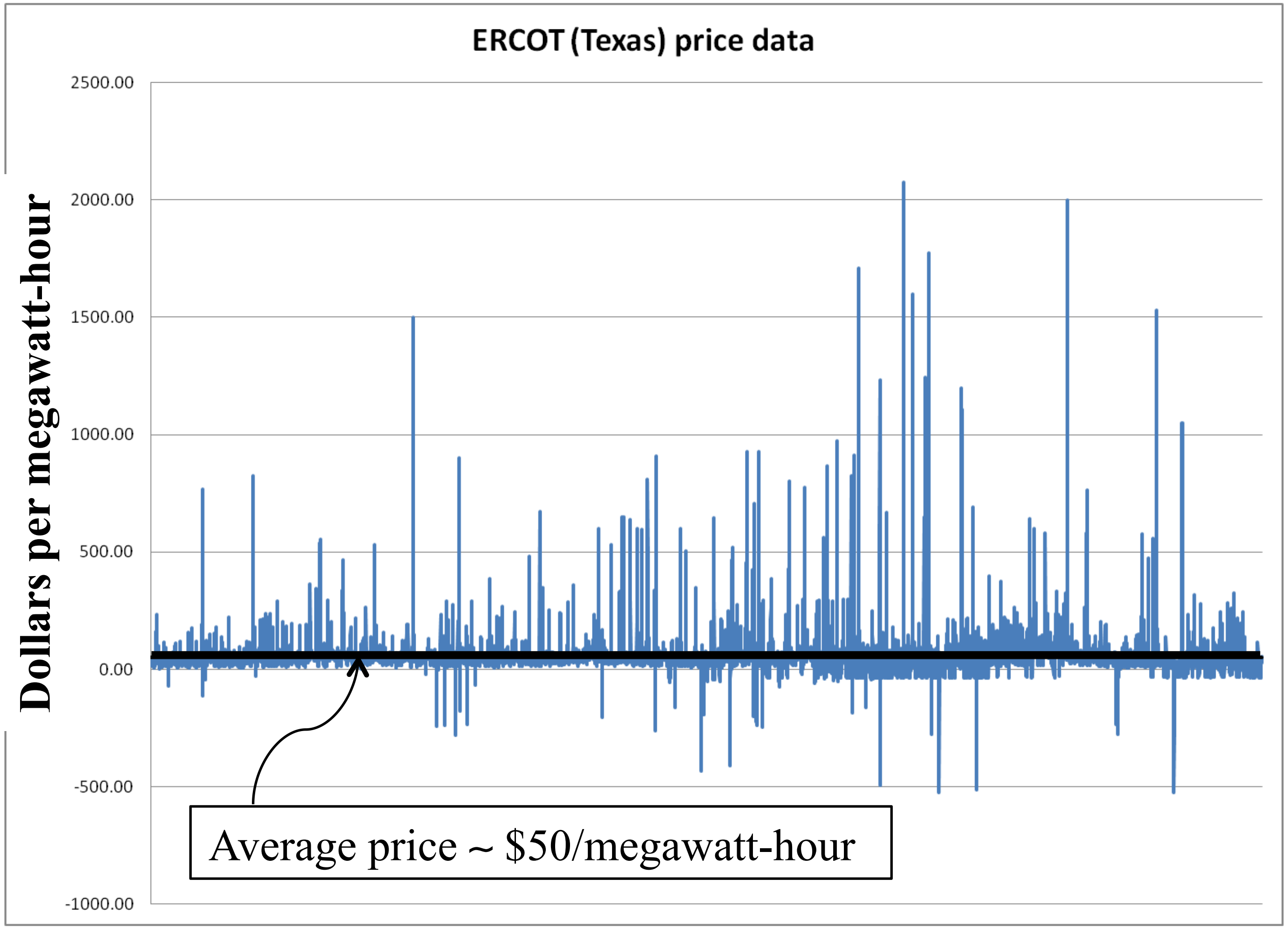
Princeton University

Operations Research and Financial Engineering


Secondary
mesh (120V)


Substation
Primary feeders
(27kV)
Transformers

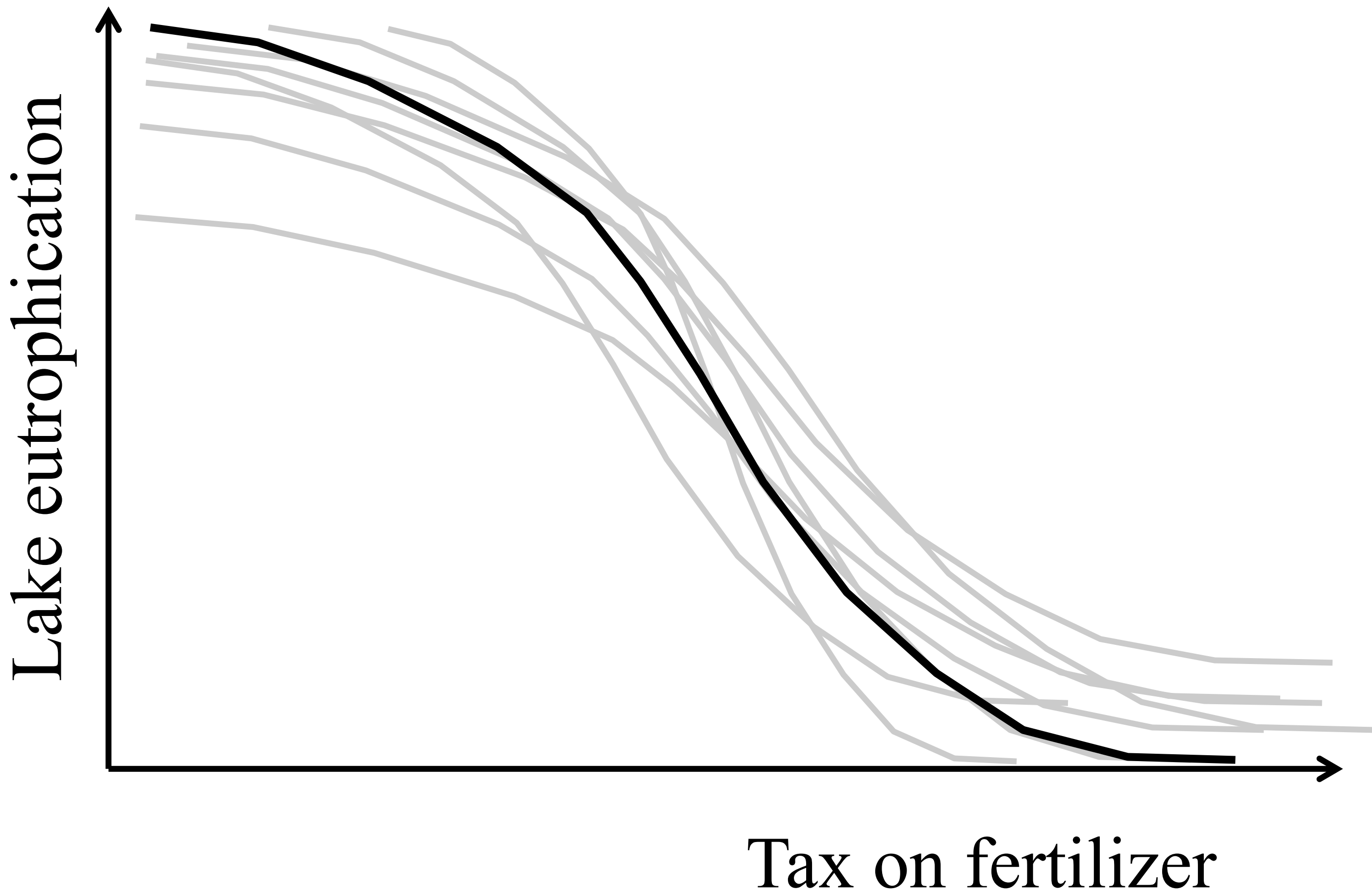
Electricity spot prices



Uncertainty in the model

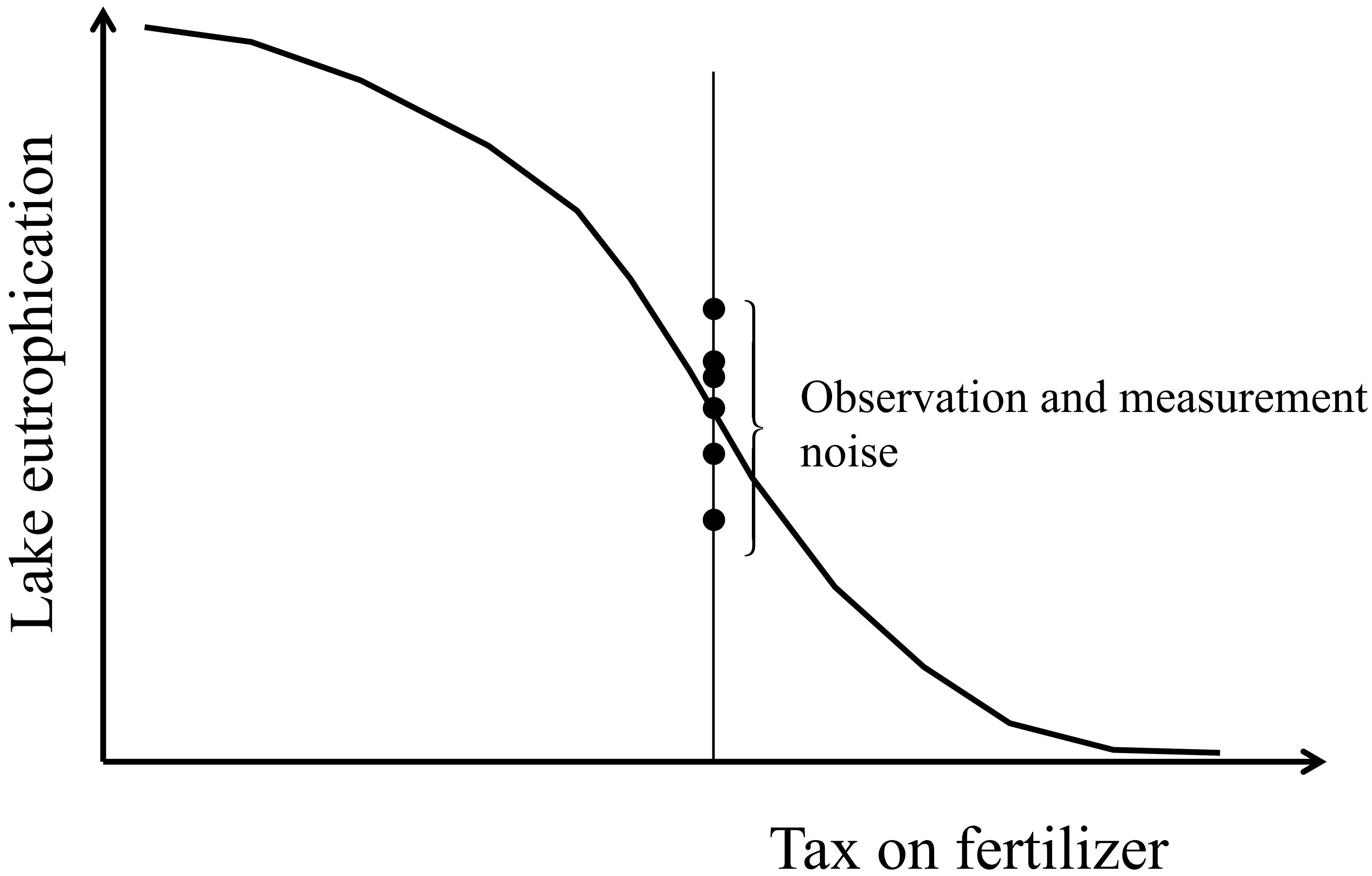
Uncertainty in the model

- Which curve is the right one?



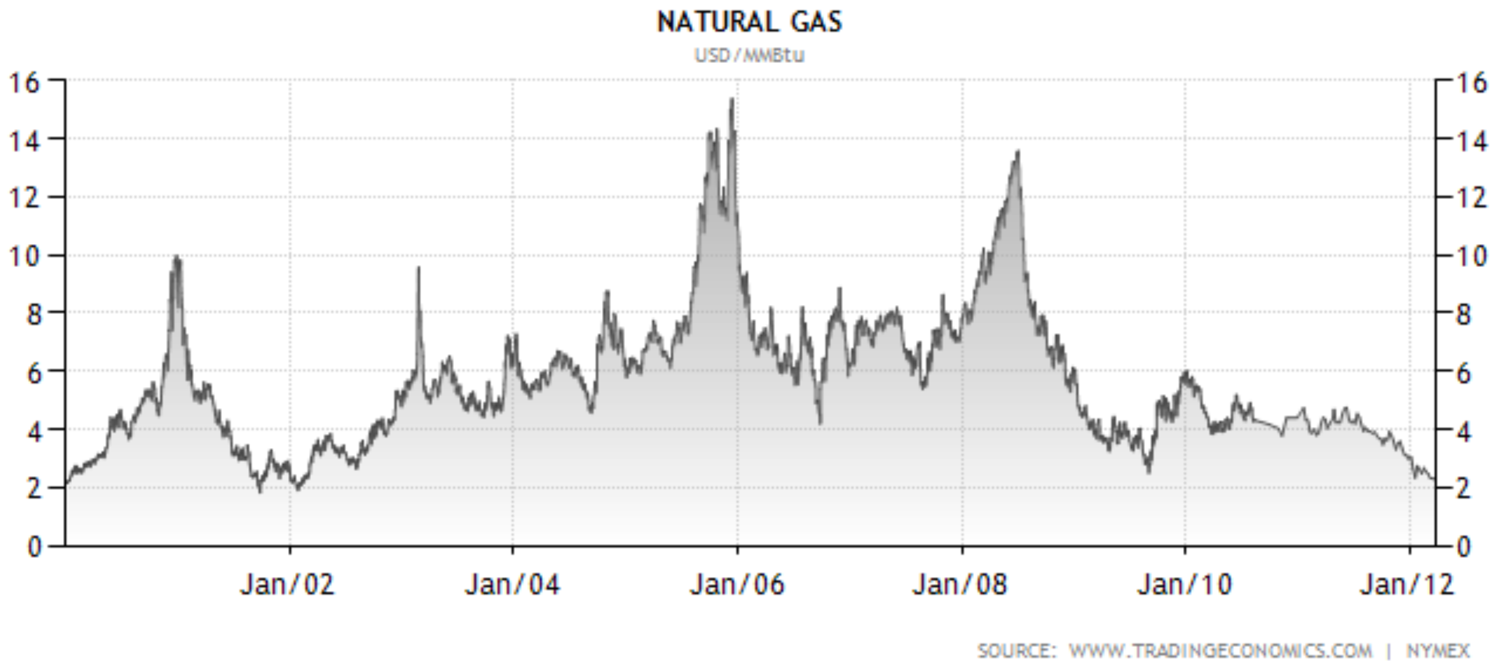
Uncertainty in the model

- Estimating the curve



Commodity prices

- The price of natural gas
  - » Reflects global and local economies, competing global commodities (primarily oil), policies (e.g. toward CO_2), and technology (e.g. fracking).



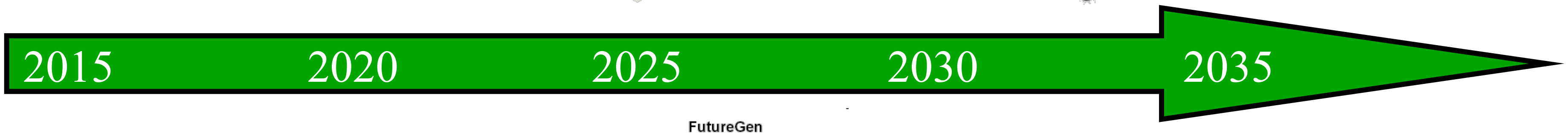
Energy resource modeling

- Need to plan long term energy investments…

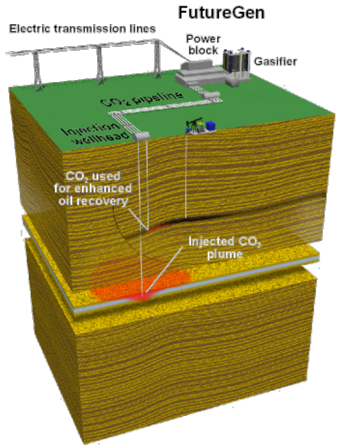
Tax policy

Price of oil

Batteries



Solar panels



Carbon capture and sequestration

Climate change

Deterministic optimization models

- We can solve deterministic models using linear programming:
  - » For static problems

$$\min cx$$
$$Ax = b$$
$$x \geq 0$$

  - » For time-staged problems

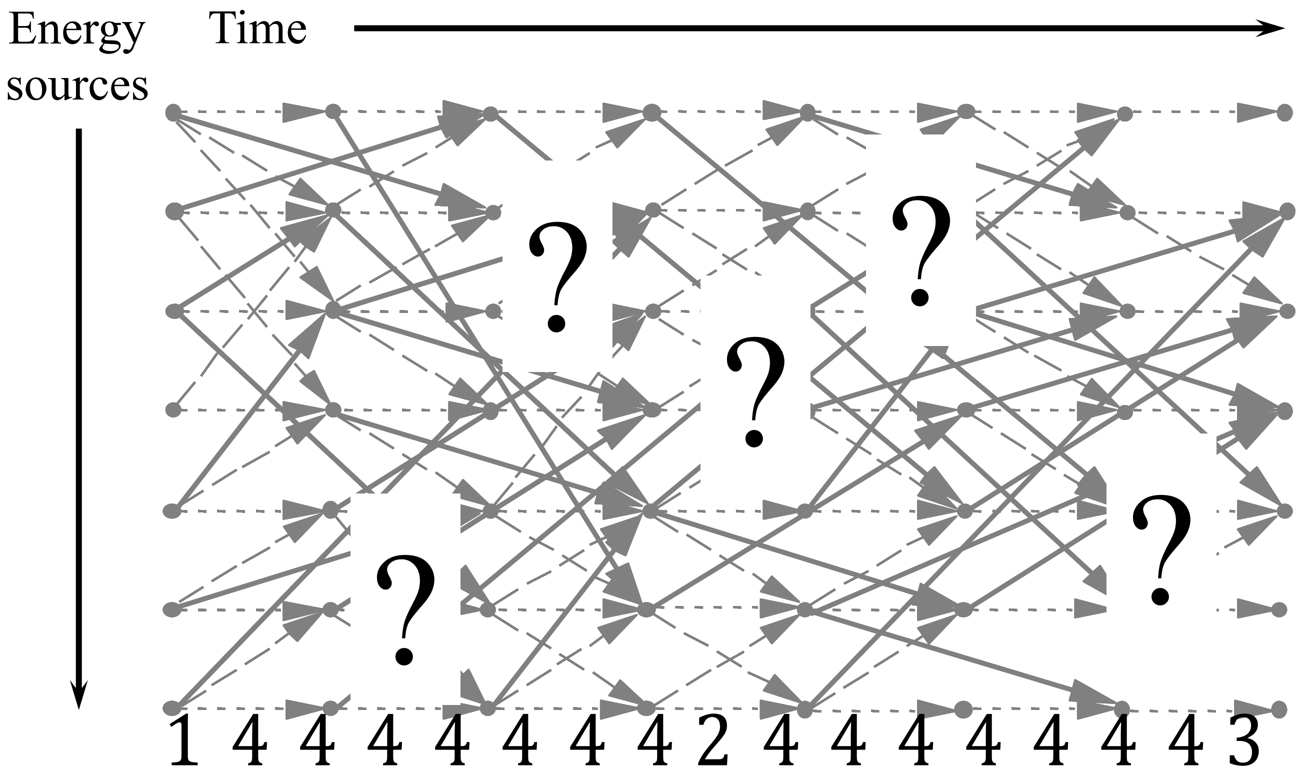
$$\min \sum_{t=0}^{T} c_t x_t$$
$$A_t x_t - B_{t-1} x_{t-1} = b_t$$
$$D_t x_t \leq u_t$$
$$x_t \geq 0$$

This is the mathematical foundation of energy policy models such as PIES, NEMS, Markal, META*Net, …

But how to handle uncertainty?

Linear programming with uncertainty

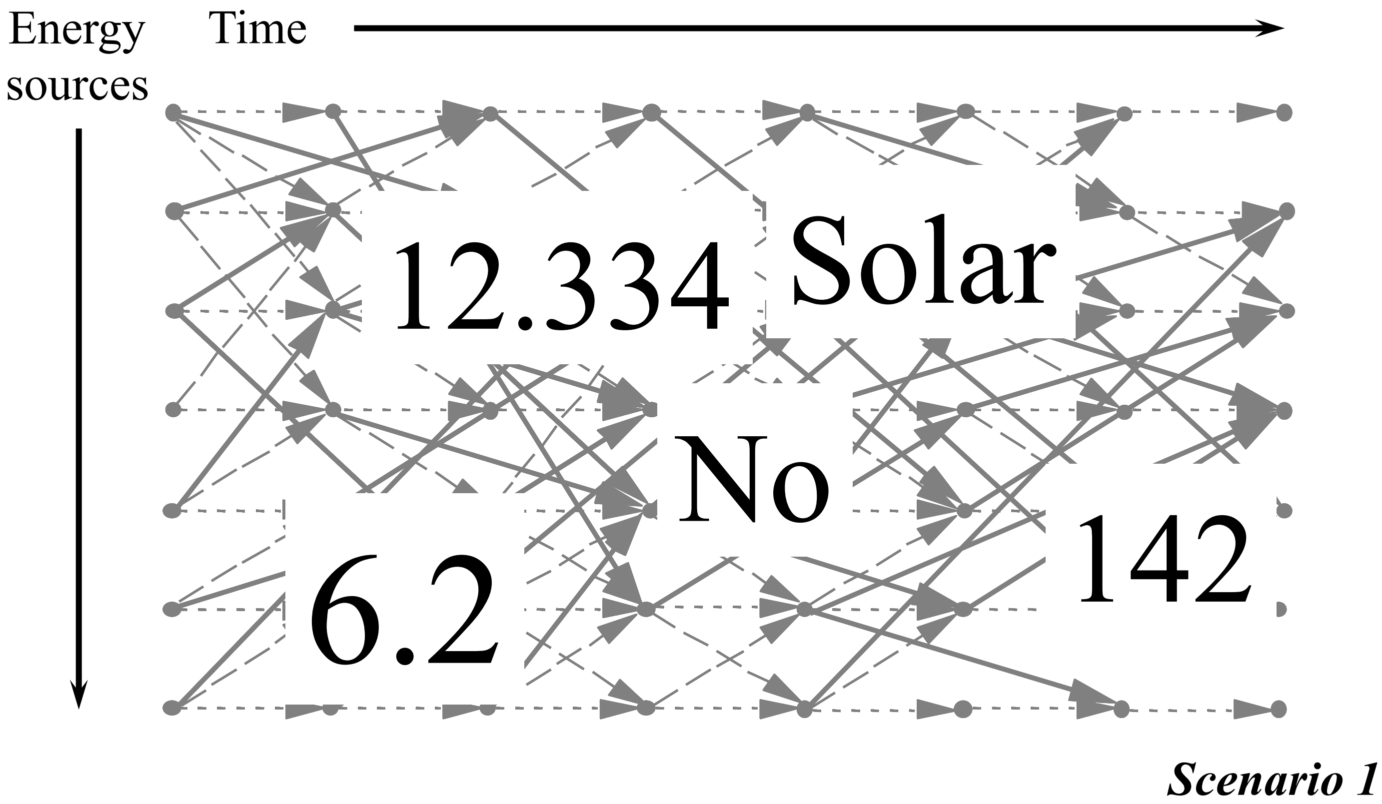
- Mixing optimization and uncertainty



"large optimization model (e.g. NEMS, MARKAL, …)"

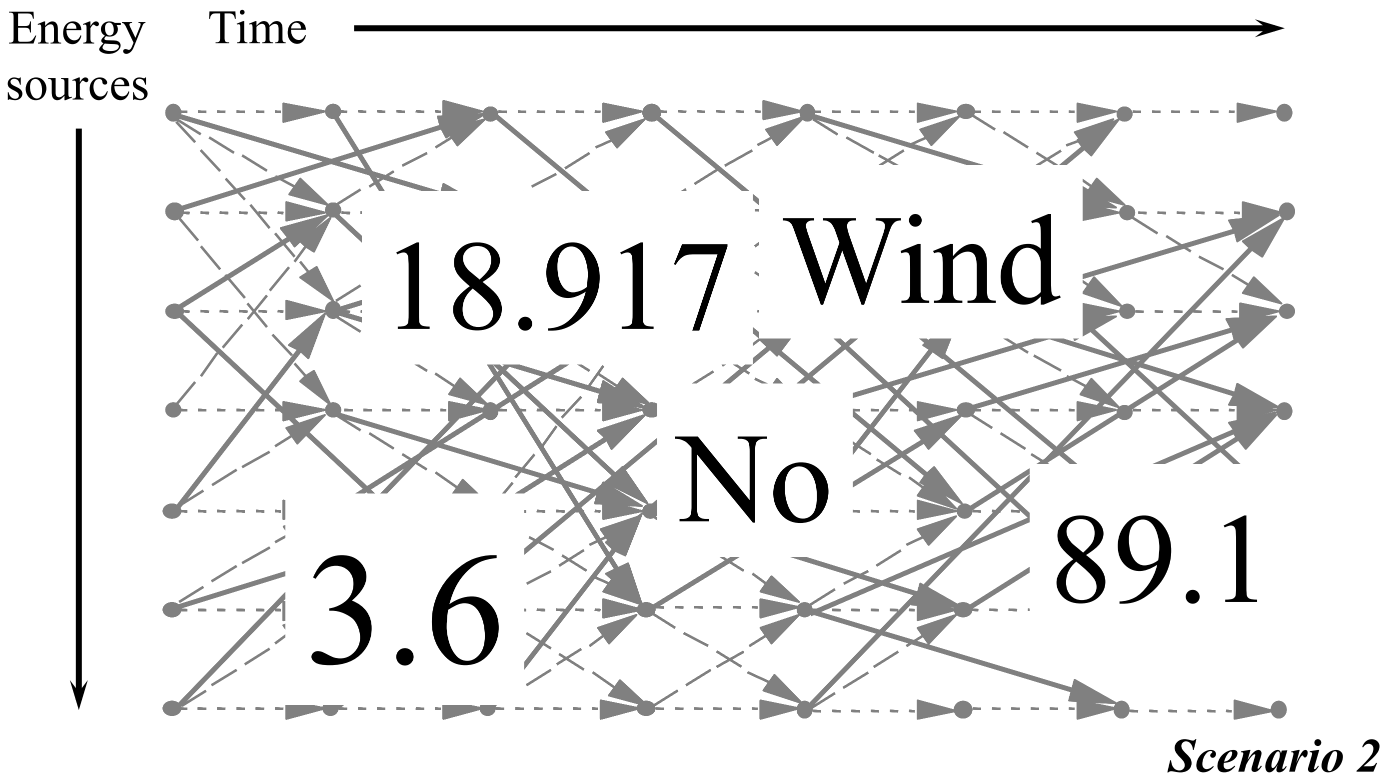
Linear programming with uncertainty

- Mixing optimization and uncertainty



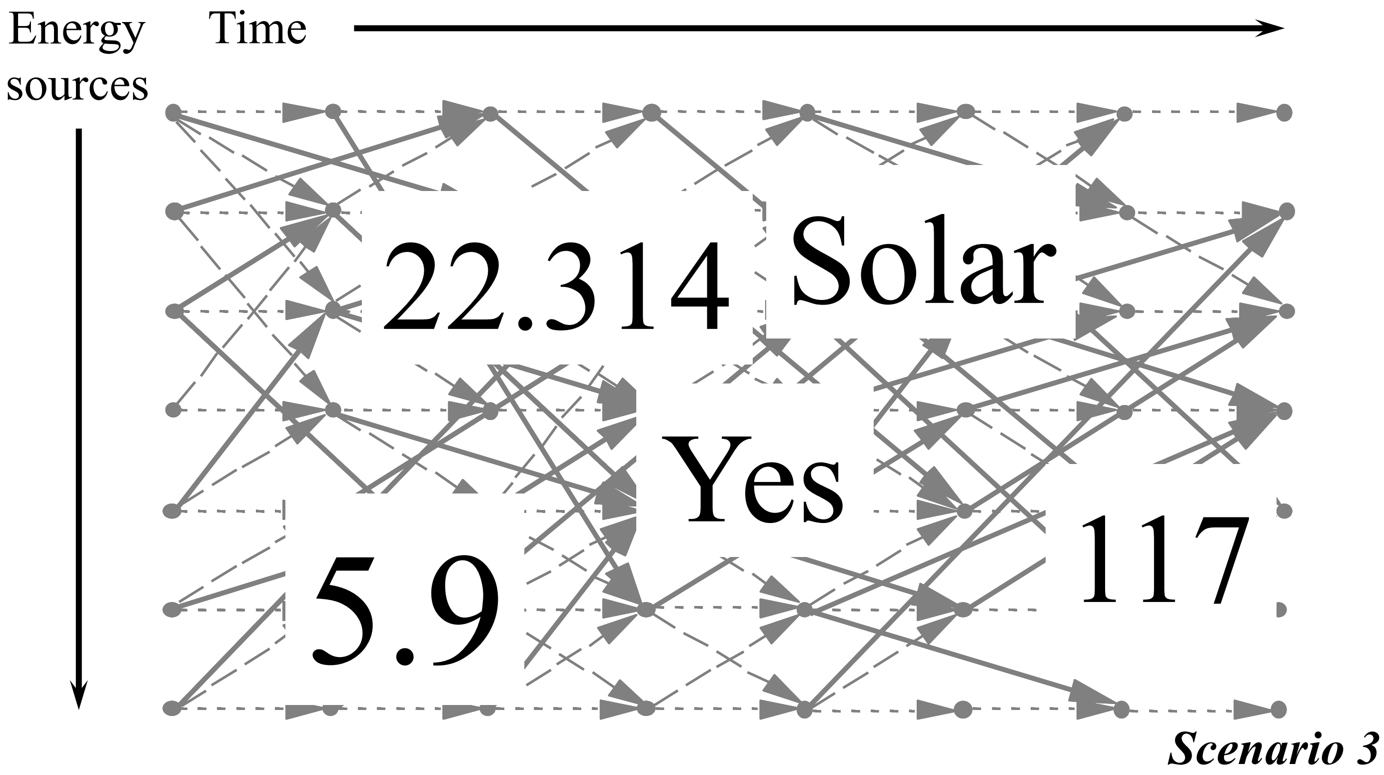
Linear programming with uncertainty

- Mixing optimization and uncertainty

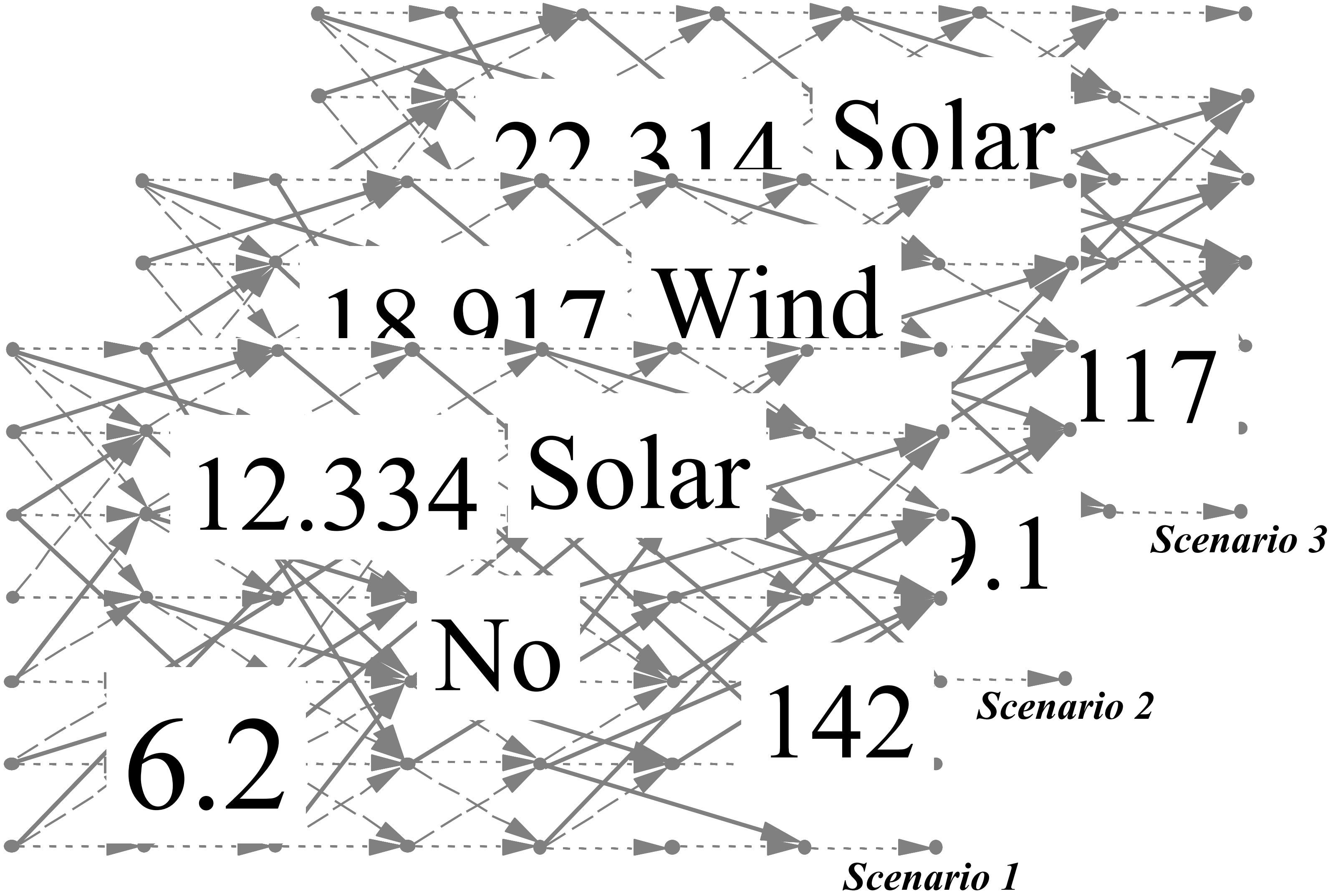


Linear programming with uncertainty

- Mixing optimization and uncertainty



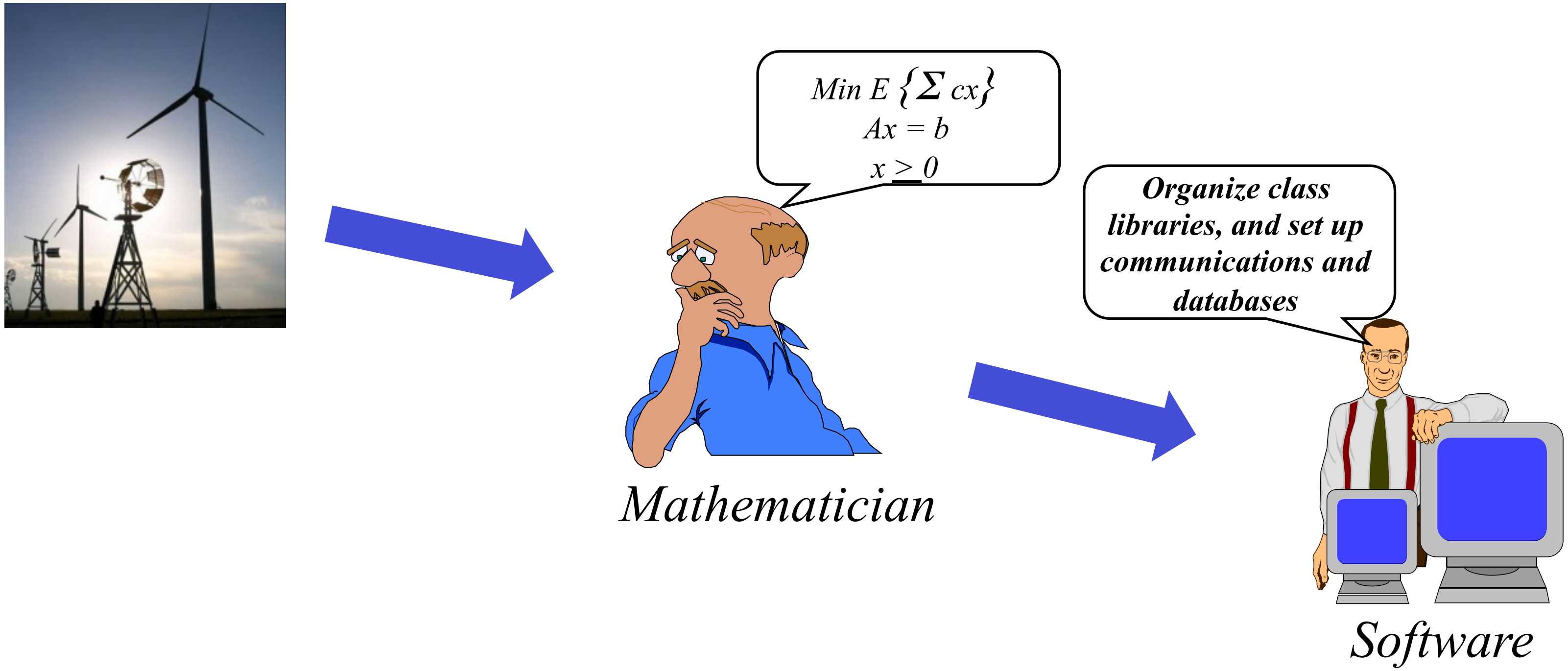
Linear programming with uncertainty



Now we have to combine the results of these three optimizations to make decisions.

Modeling dynamic problems

- Before we can *solve* these problems, we have to know how to *think* about them.



- The biggest challenge when making decisions under uncertainty is *modeling*.

Modeling dynamic problems

- The system state:




$S_t = (R_t, I_t, K_t)$ = System state, where:

R_t = Resource state (physical state)

Energy investments, energy storage, ...

Status of generators

I_t = Information state

State of the technology (costs, performance)

Climate, weather (temperature, rainfall, wind)

Market prices (oil, coal)

K_t = Knowledge state ("belief state")

Belief about the effect of CO2 on the environment

Belief about the effect of fertilizer on algal blooms

The state variable is the minimally dimensioned function of history that allows us to calculate the decision function, cost function and transition function.

Modeling dynamic problems

- Decisions:




Computer science

a_t = Discrete action

Control theory

u_t = Low-dimensional continuous vector

Operations research

x_t = Usually a discrete or continuous but high-dimensional vector of decisions.

$\pi(s)$ = Decision function (or "policy") mapping a state to an an action a, control u or decision x.

I prefer to write $A^{\pi}(s)$ as the function that returns an action a, where $\pi \in \Pi$ is the set of all policies (or functions). Use $X^{\pi}(s)$ if using decision x or $U^{\pi}(s)$ for control u.

Modeling dynamic problems

- Exogenous information:

W_t = New information = $\left(\hat{R}_t, \hat{D}_t, \hat{E}_t, \hat{p}_t\right)$

$\hat{R}_t$ = Exogenous changes in capacity, reserves

New gas/oil discoveries, breakthroughs in technology

$\hat{D}_t$ = New demands for energy from each source

Demand for energy

$\hat{E}_t$ = Changes in energy from wind and solar

$\hat{p}_t$ = Changes in prices of commodities, electricity, technology

Modeling dynamic problems

- The transition function




$$S_{t+1} = S^M(S_t, x_t, W_{t+1})$$

$$R_{t+1} = R_t + Ax_t + \hat{R}_{t+1}$$ Water in the reservoir

$$p_{t+1} = p_t + \hat{p}_{t+1}$$ Spot prices

$$e_{t+1}^{Wind} = e_t^{Wind} + \hat{e}_{t+1}^{Wind}$$ Energy from wind

Also known as the:

“System model”
“State transition model”
“Plant model”
“Model”

Stochastic optimization models

- The objective function

$$\min_{\pi} E^{\pi}\left\{\sum_t \gamma^t C\left(S_t, X^{\pi}(S_t)\right)\right\}$$

Expectation over all random outcomes

Cost function

State variable

Decision function (policy)

Finding the best policy

Given a *system model* (transition function)

$$S_{t+1} = S^M\left(S_t, x_t, W_{t+1}(\omega)\right)$$

» We have to find the best policy, which is a function that maps states to feasible actions, using only the information available when the decision is made.


Stochastic programming
Stochastic search
Model predictive control
Optimal control
Reinforcement learning
Q – learning
On-policy learning
Off-policy learning
Markov decision processes
Simulation optimization
Policy search

AI, OR and Control Theory: A Rosetta Stone for Stochastic Optimization

Warren B. Powell

July 4, 2012

Four classes of policies

1) Myopic policies

» Take the action that maximizes contribution (or minimizes cost) for just the current time period:

$$X^M(S_t) = \arg\max_{x_t} C(S_t, x_t)$$

» We can parameterize myopic policies with bonus and penalties to encourage good long-term behavior.

» We may use a ***myopic cost function approximation:***

$$X^M(S_t \mid \theta) = \arg\max_{x_t} \bar{C}^{\pi}(S_t, x_t \mid \theta)$$

The cost function approximation $\bar{C}^{\pi}(S_t, x_t \mid \theta)$ may be designed to produce better long-run behaviors.

Four classes of policies

2) Lookahead policies

Plan over the next T periods, but implement only the action it tells you to do now:

» Deterministic forecast (most common)

$$X^{LA}(S_t) = \arg\max_{x_t, x_{t+1}, \ldots, x_{t+T}} C(S_t, x_t) + \sum_{t'=t+1}^{T} C(S_{t'}, x_{t'})$$

» Probabilistic lookahead

» Rolling/receding horizon procedures

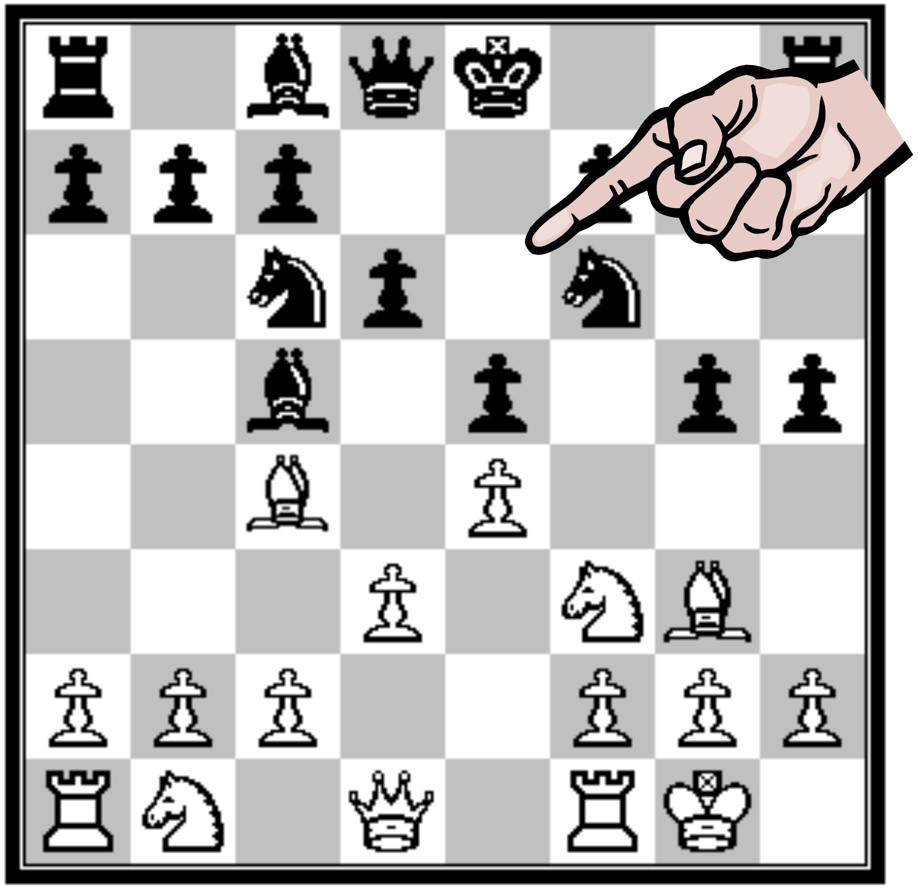
» Model predictive control

» Rollout heuristics

» Tree search (decision trees)

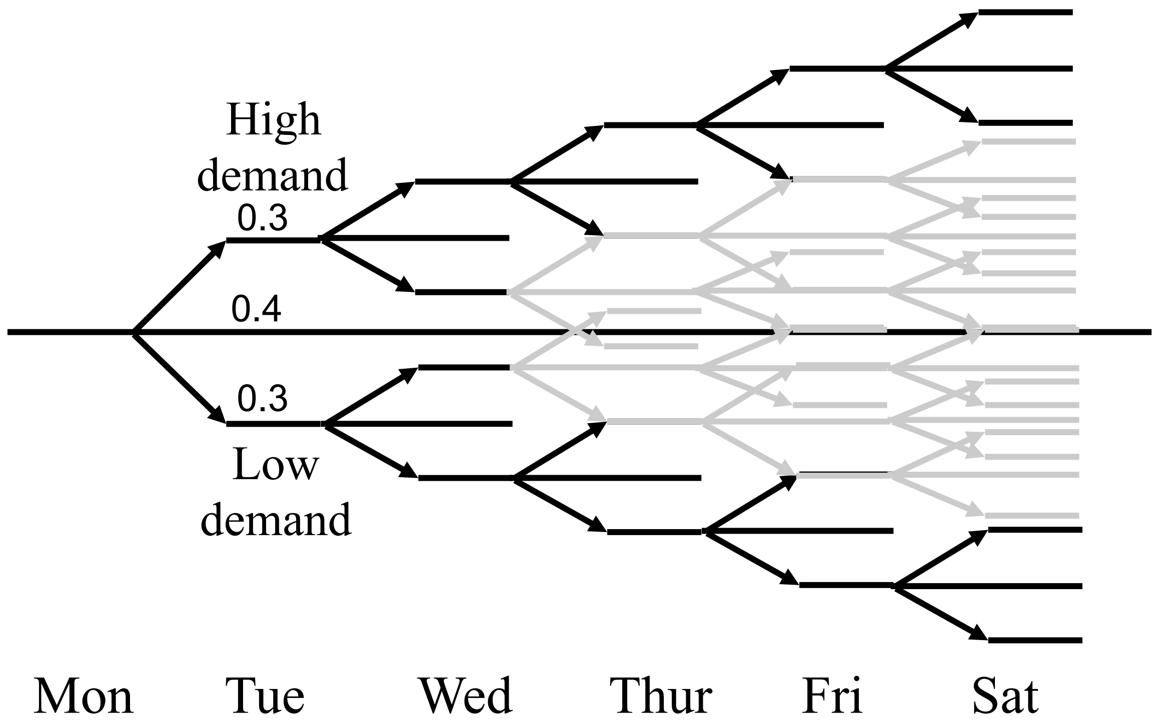
Four classes of policies

- A lookahead policy



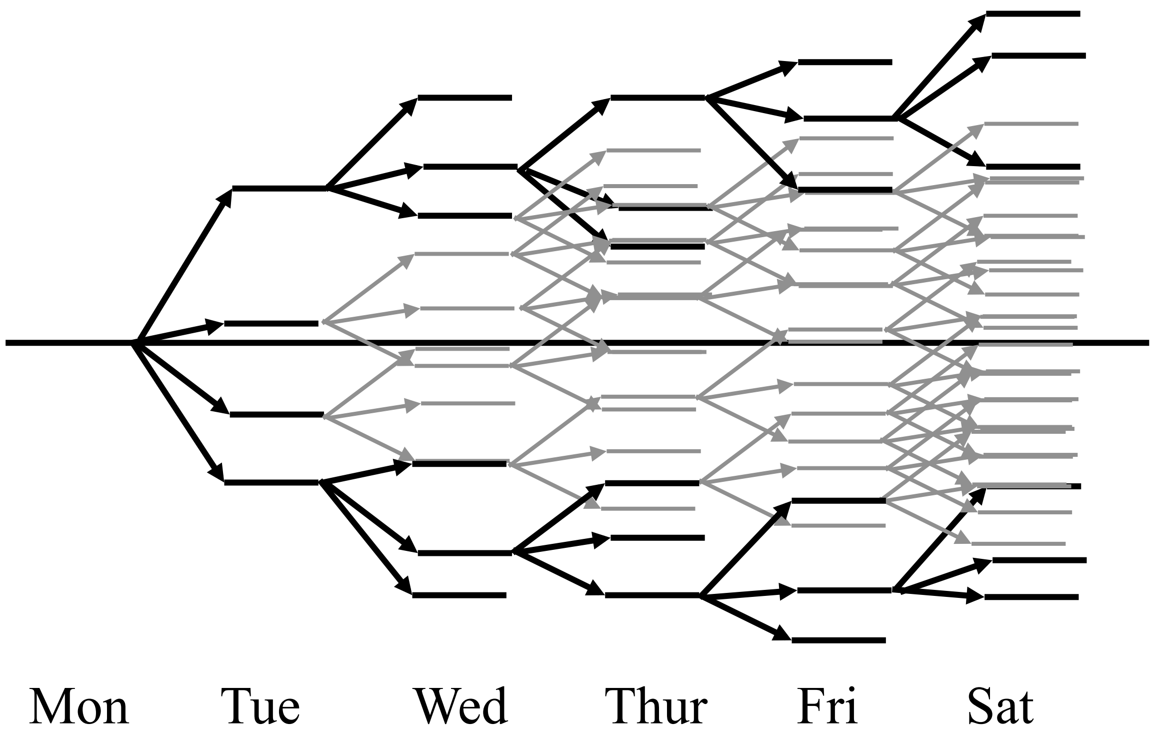
Four classes of policies

- Probabilistic lookahead
  - » This formulation is popular in water resource planning



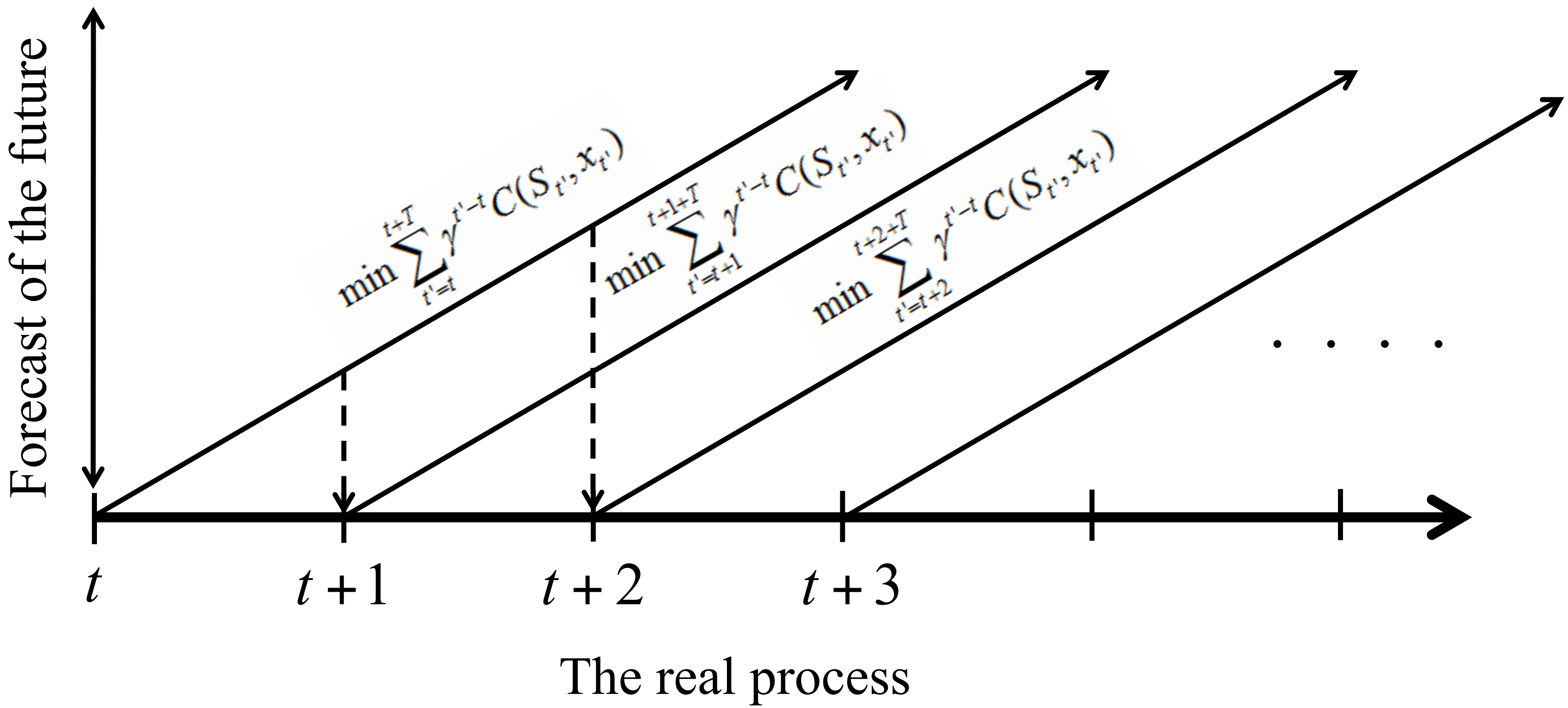
Four classes of policies

- Probabilistic lookahead
  - » This formulation is popular in water resource planning



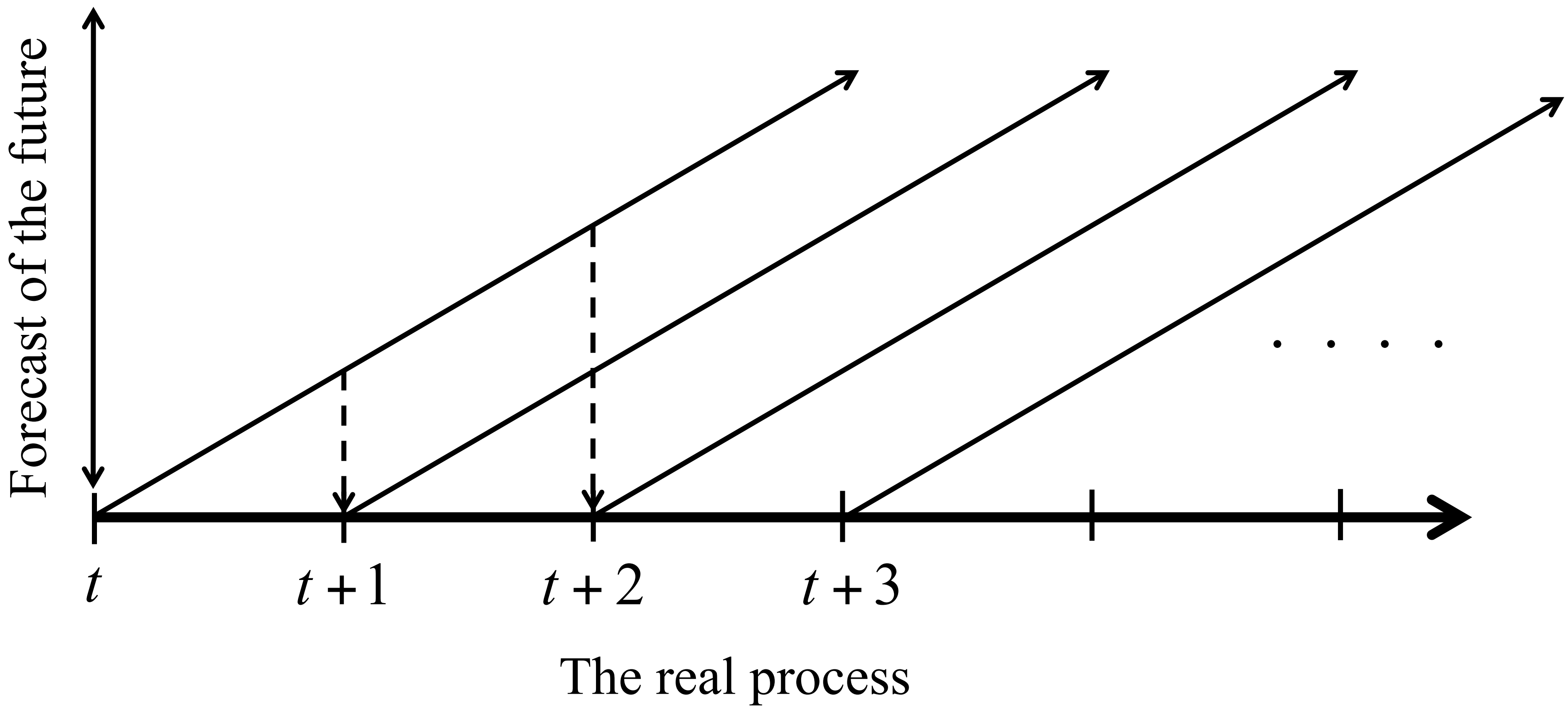
Lookahead policies

- Lookahead policies peek into the future
  - » Optimize over point forecast of the future



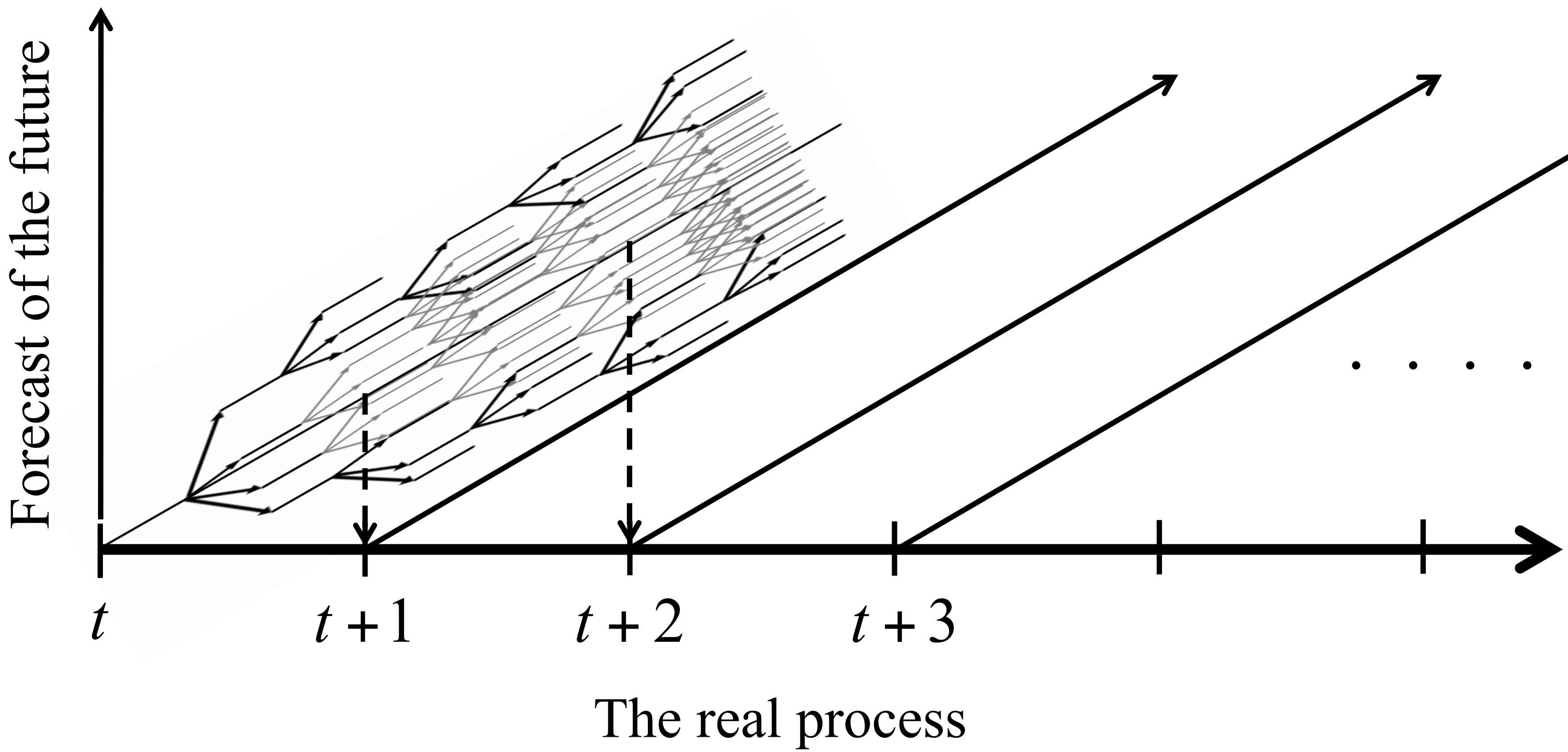
Four classes of policies

- Probabilistic lookahead



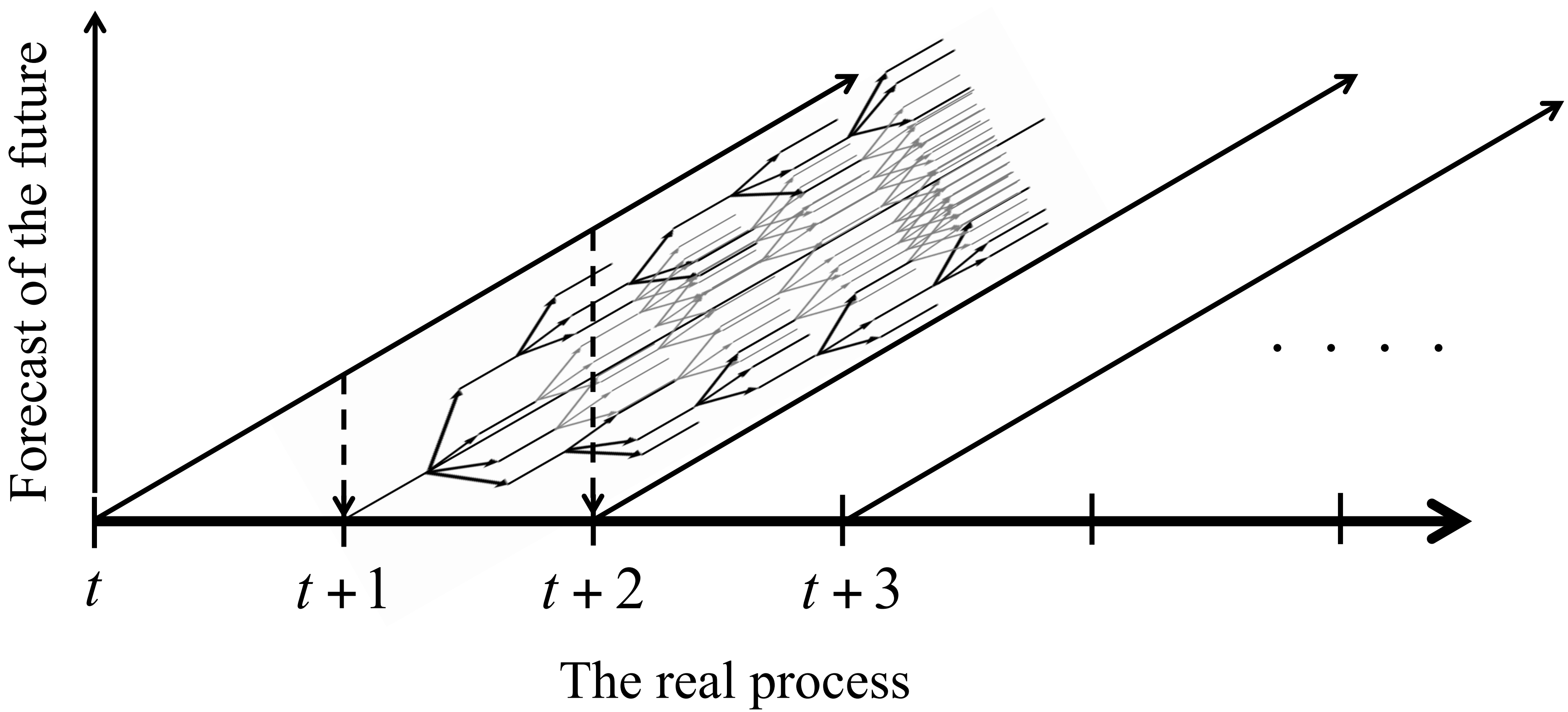
Four classes of policies

- Probabilistic lookahead
  - » Optimize over *stochastic* model of the future.



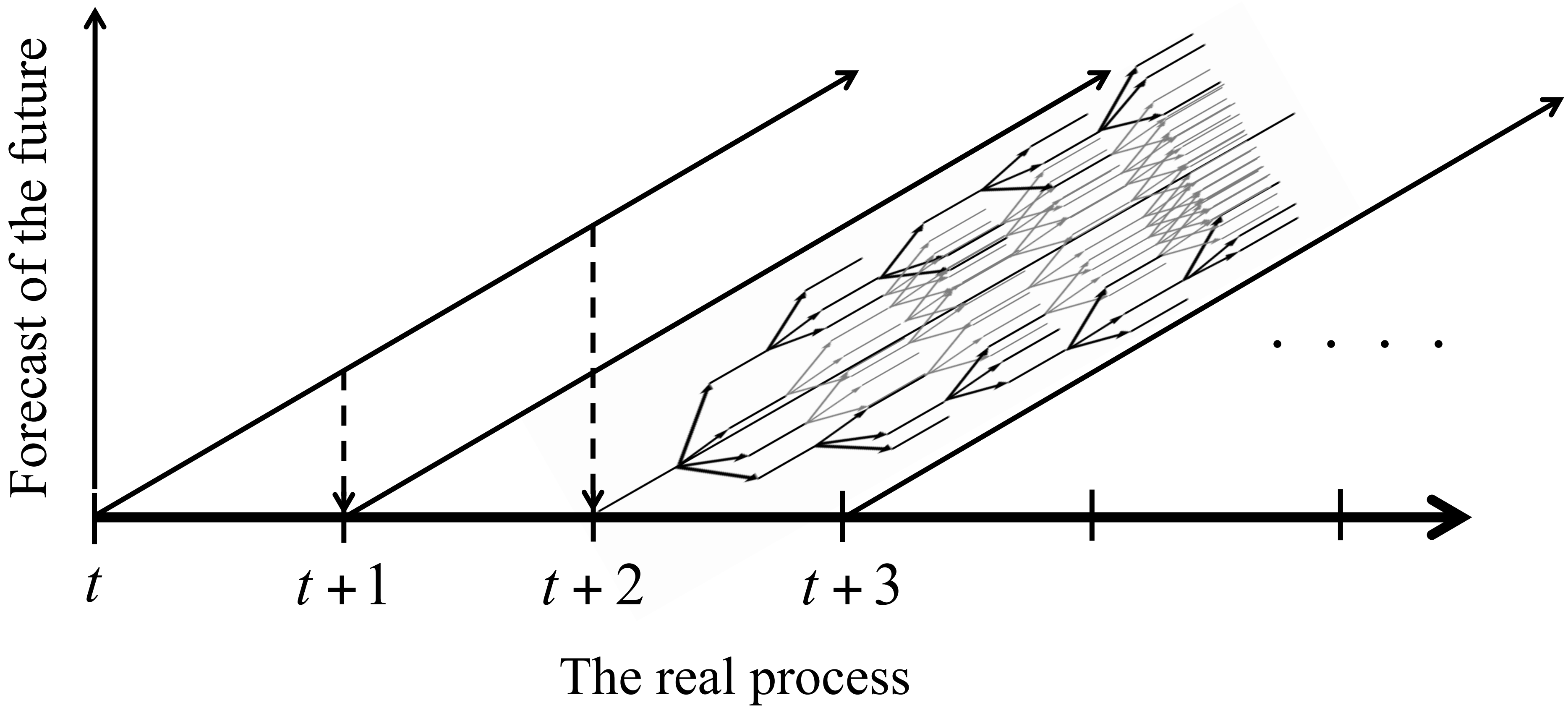
Four classes of policies

- Probabilistic lookahead
  - » Optimize over *stochastic* model of the future.



Four classes of policies

- Probabilistic lookahead
  - » Optimize over *stochastic* model of the future.



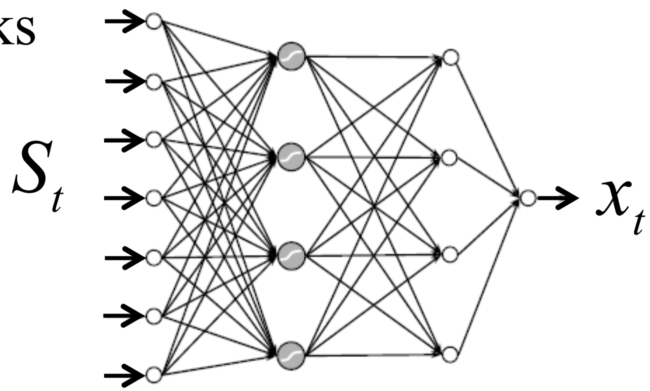
Four classes of policies

3) Policy function approximations

- » Lookup table
  - Recharge the battery between 2am and 6am each morning, and discharge as needed.
- » Parameterized functions
  - Recharge the battery when the price is below ρ^{charge} and discharge when the price is above $\rho^{\text{discharge}}$
- » Regression models

$$X^{PFA}(S_t \mid \theta) = \theta_0 + \theta_1 S_t + \theta_2 \left(S_t\right)^2$$

- » Neural networks



Four classes of policies

4) Policies based on value function approximations

» Using the pre-decision state

$$X^{VFA}(S_t) = \arg\max_{x_t} \left(C(S_t, x_t) + \gamma E \bar{V}_{t+1}(S_{t+1}) \right)$$

» Or the post-decision state:

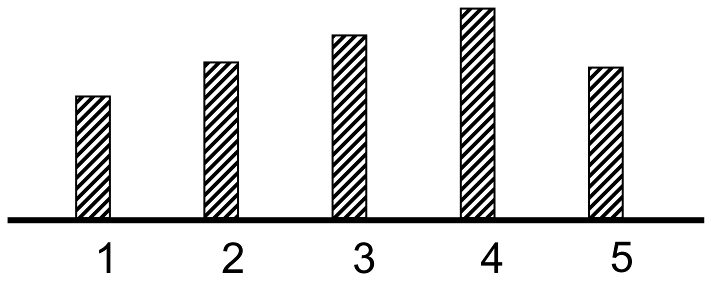
$$X^{VFA}(S_t) = \arg\max_{x_t} \left(C(S_t, x_t) + \gamma \bar{V}_{t+1}\left(S_t^x(S_t, x_t) \right) \right)$$

» This is what most people associate with "approximate dynamic programming"

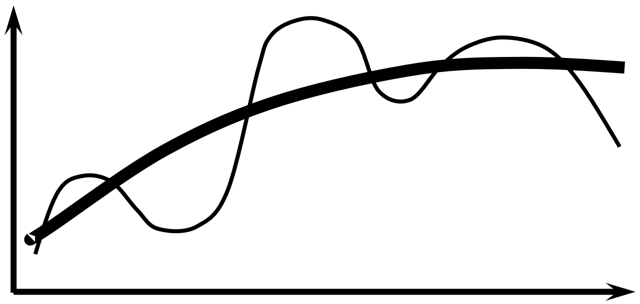
Four classes of policies

- There are three classes of approximation strategies

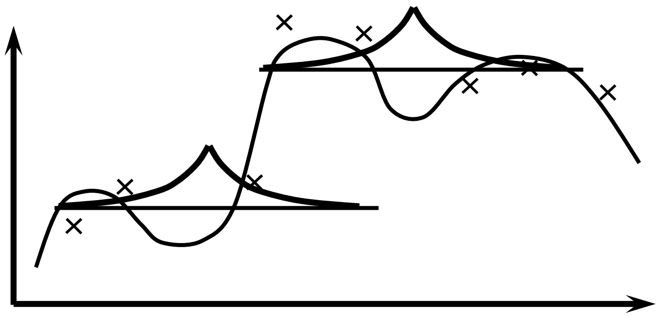
  » Lookup table
  - Given a discrete state, return a discrete action or value



  » Parametric models
  - Linear models (“basis functions”)
  - Nonlinear models
  - Neural networks



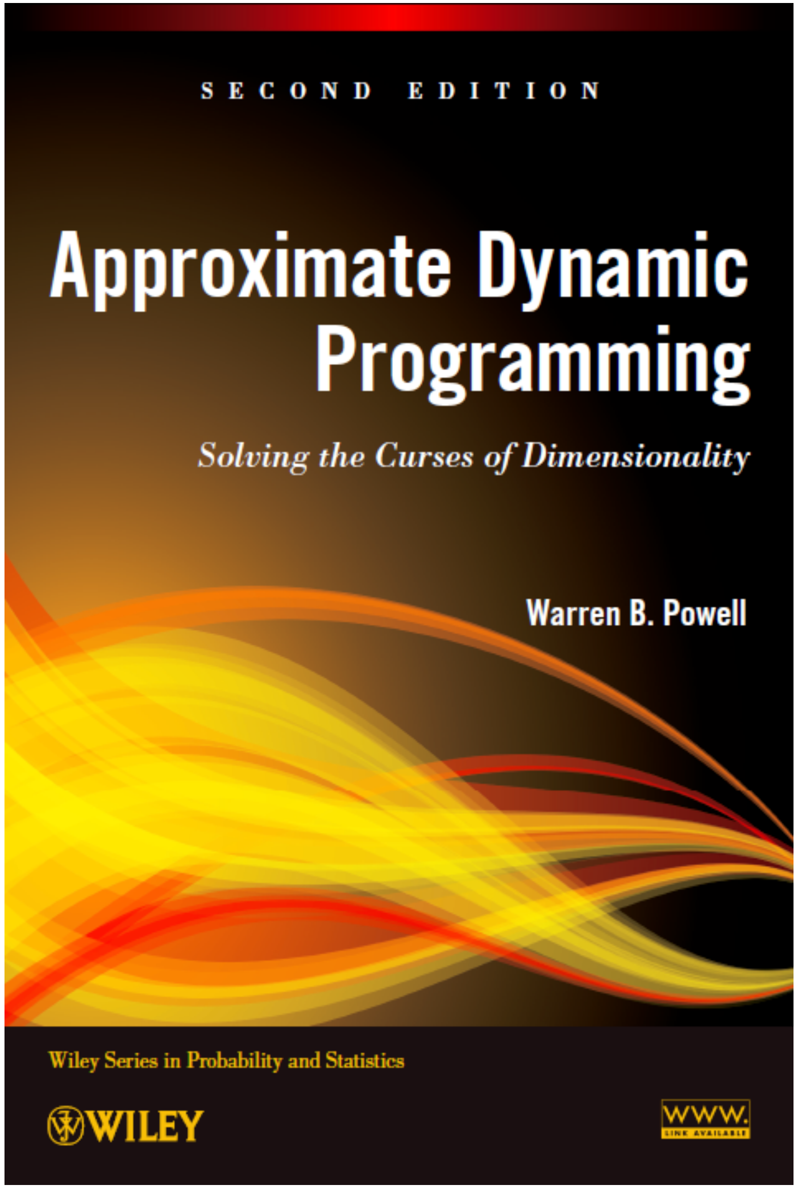
  » Nonparametric models
  - Kernel regression
  - Neural networks
  - Piecewise linear approximations
  - Splines, ….



Approximate dynamic programming

- Second edition
  - 300+ new pages
  - Four fundamental classes of policies
  - New chapter dedicated to policy search (uses optimal learning)
  - 3-chapter sequence for value function approximations.
  - Chapter 5 (on modeling) and chapter 6 (on policies) available at:

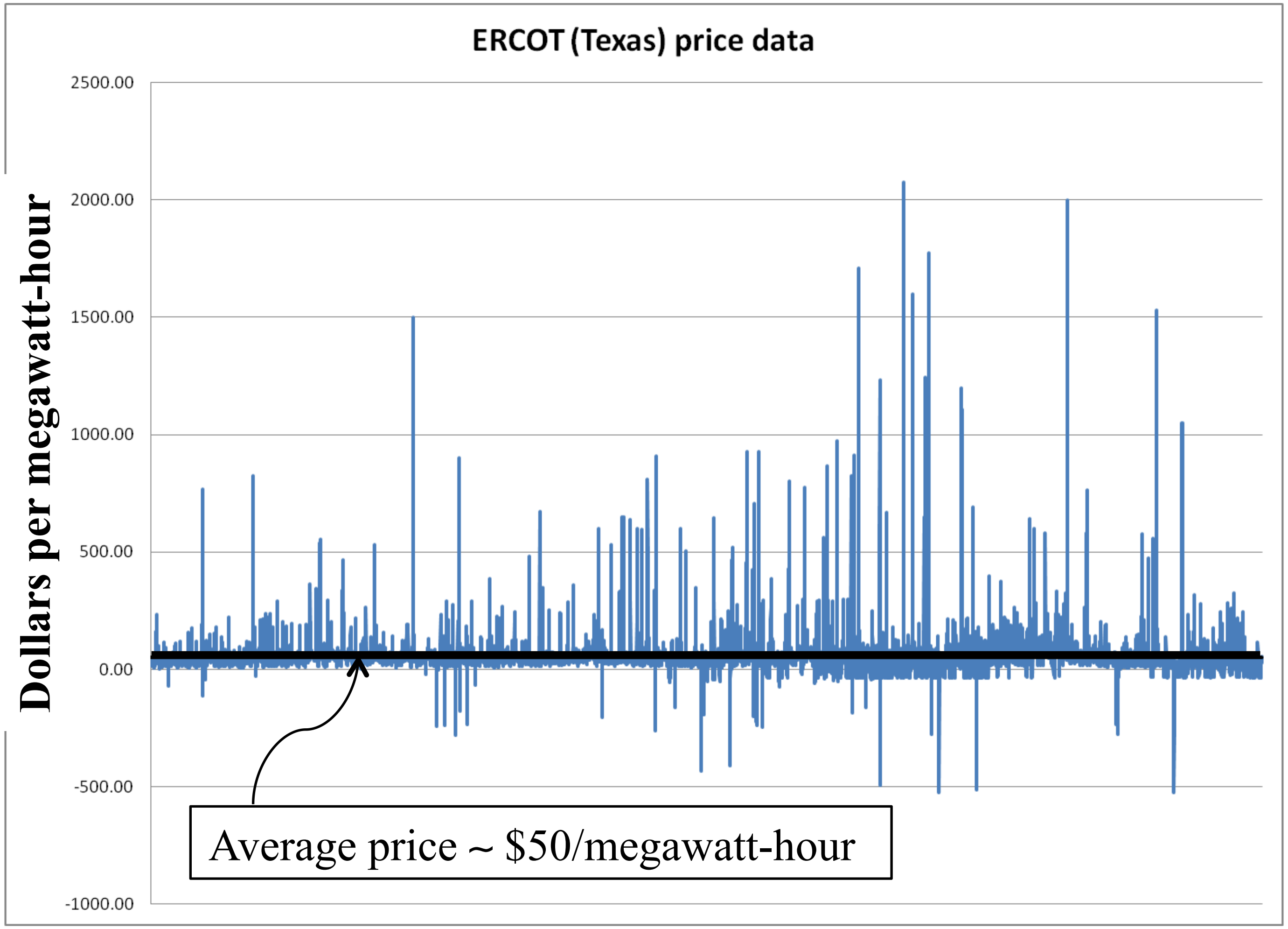
    http://adp.princeton.edu/



Lecture outline

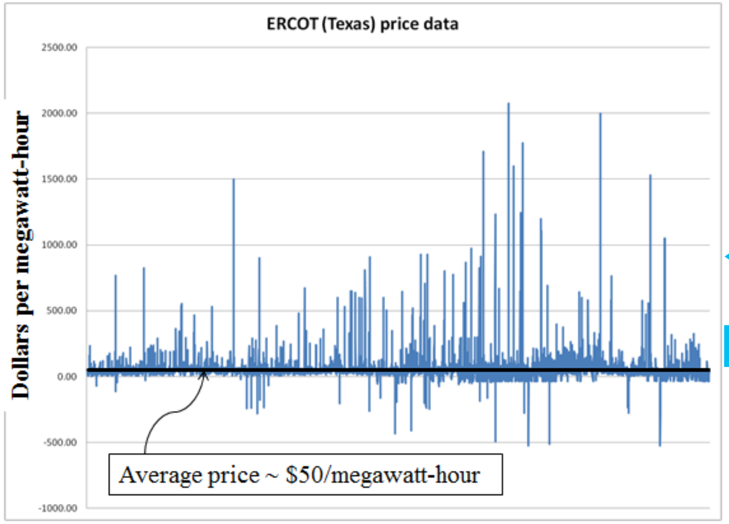
- Optimizing energy storage using a policy function approximation

Electricity spot prices



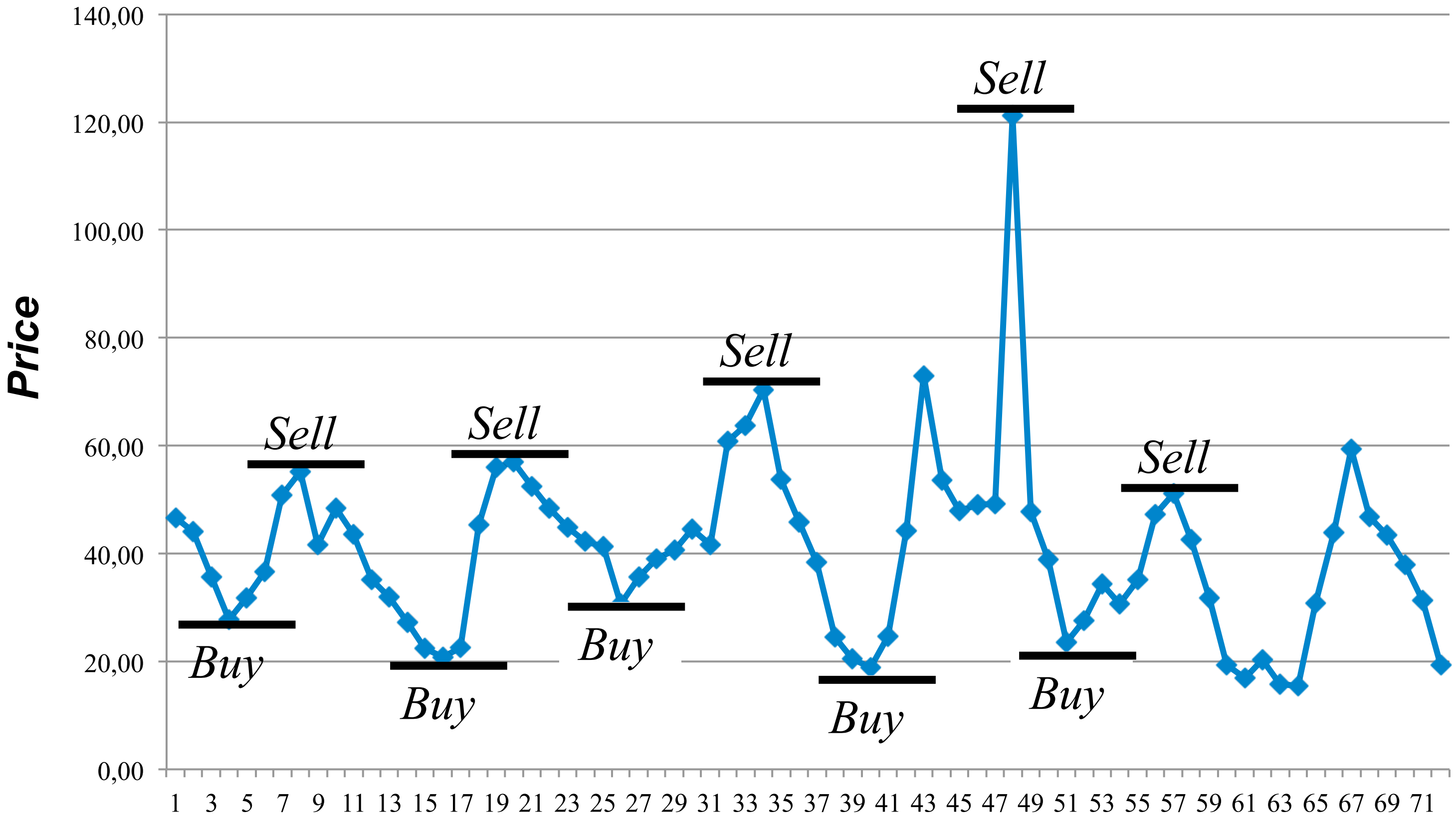
Policy optimization

- Optimizing a policy for battery arbitrage

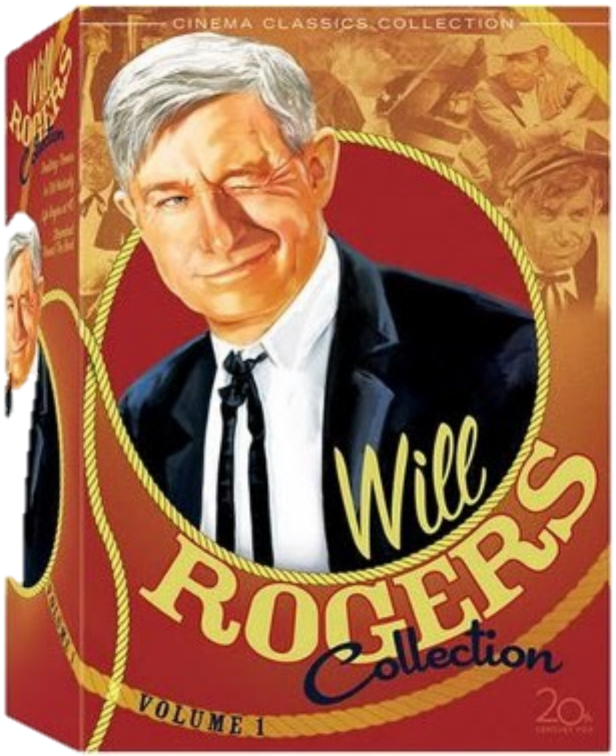


Optimizing storage

- Challenge: find a policy for charging and discharging the battery
  - » Strategy posed by the battery manufacturer: “Buy low, sell high”



Decision making under uncertainty



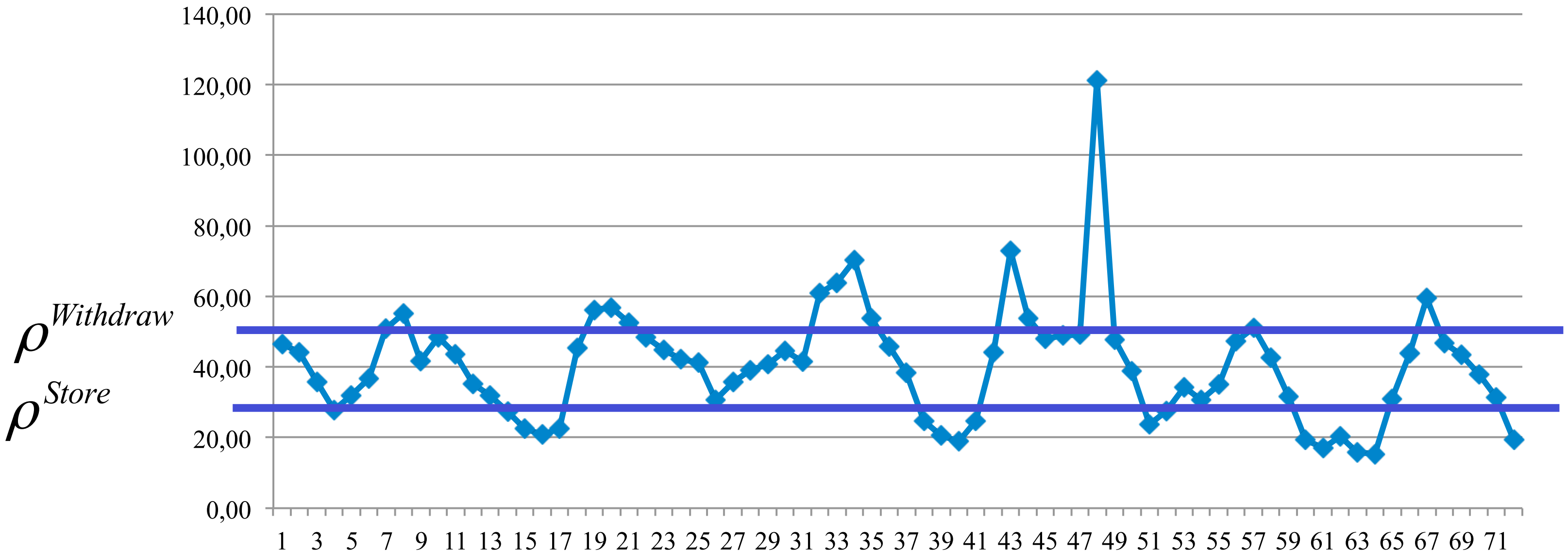
Don't gamble; take all your savings and buy some good stock and hold it till it goes up, then sell it. If it don't go up, don't buy it.

Will Rogers

It is not enough to model the *variability* of a process. You have to model the *uncertainty* – the flow of information.

Optimizing storage

- We had to design a *simple*, *implementable* policy that did not cheat!

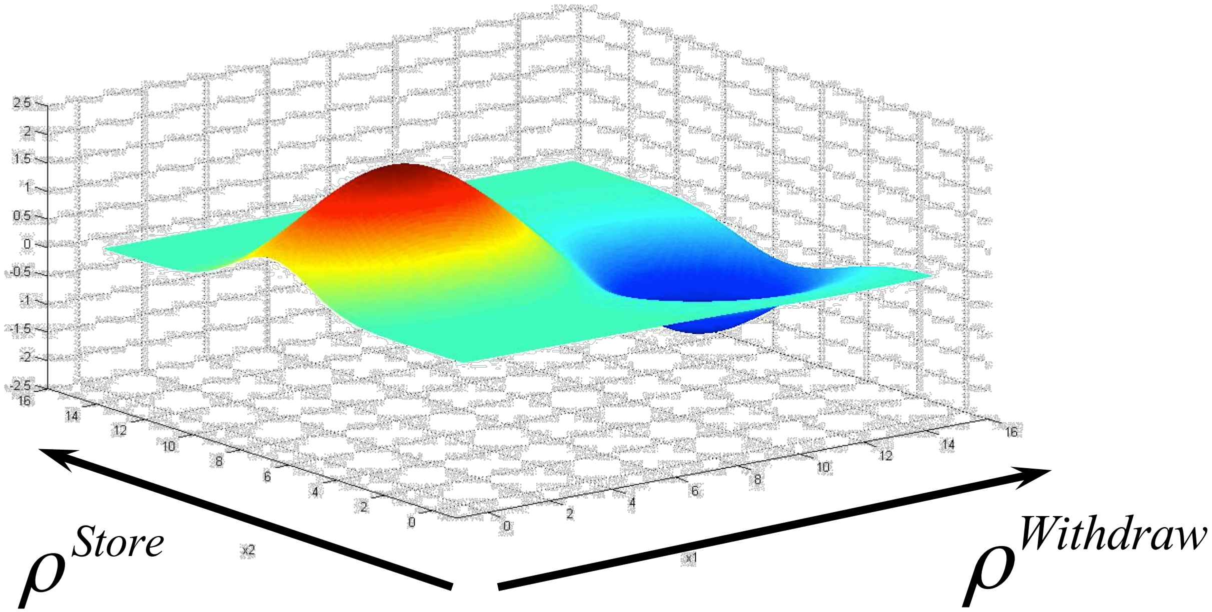


- We have developed a separate line of research in *optimal learning* to determine ρ^{Store} and $\rho^{Withdraw}$.

Optimizing storage

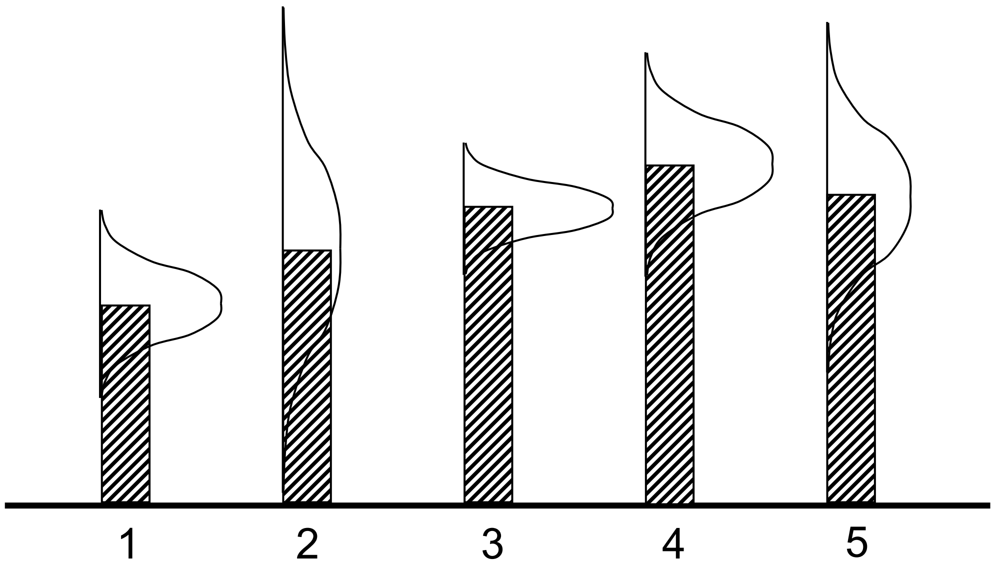
- Finding the best policy ("policy search")
  - » Let $X^{\pi}(S_t \mid \rho^{store}, \rho^{withdraw})$ be the "policy" that chooses the actions.
  - » We wish to maximize the function

$$\min_{\rho} \mathrm{E}F(\rho, W) = \mathrm{E}\sum_{t=0}^{T} \gamma^t C\left(S_t, X^{\pi}(S_t \mid \rho)\right)$$



Optimal learning

- Now assume we have five choices, with uncertainty in our belief about how well each one will perform.
- If you can make one measurement, which would you measure?



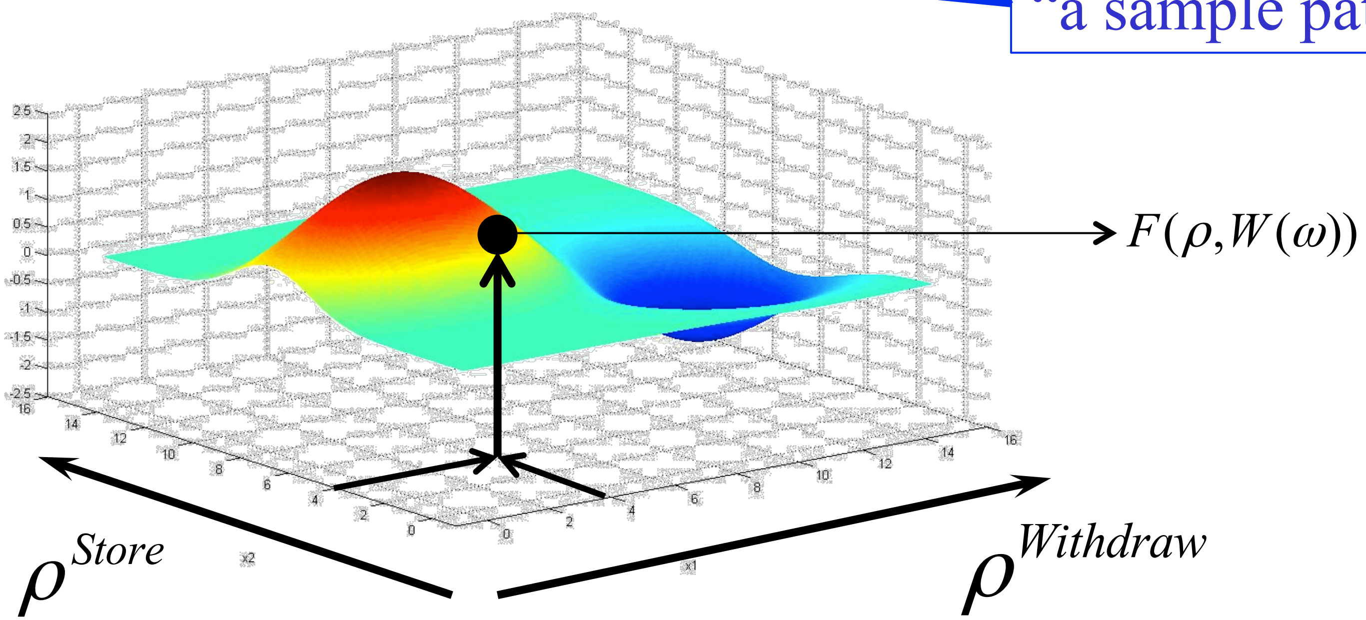
Possible values of $x = (\rho^{store}, \rho^{withdraw})$

Optimal learning

- Policy search process:
  - » Choose ρ^{store} and $\rho^{withdraw}$
  - » Simulate the policy to get a noisy estimate of its value:

$$F(\rho, W(\omega)) = \sum_{t=0}^{T} \gamma^t C\Big(S_t(\omega), X^\pi(S_t(\omega) \mid \rho)\Big)$$

"a sample path"



Optimal learning

- At first, we believe that

$$\mu_x \sim N\left(\theta_x^0, 1/\beta_x^0\right)$$

- But we measure alternative x and observe

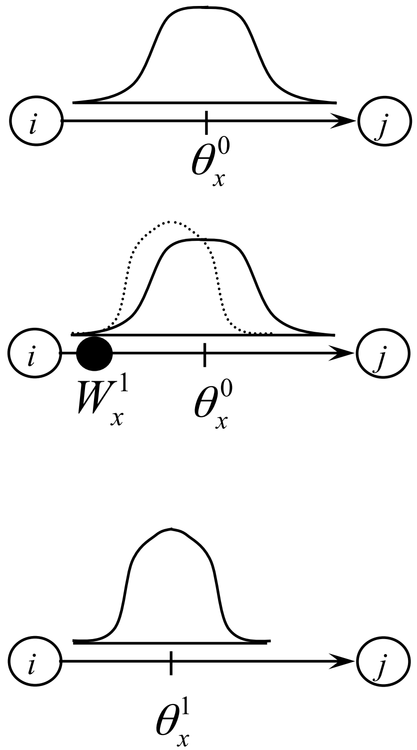
$$y_x^1 = F(\rho, W(\omega)) \sim N\left(\mu_x, 1/\beta^W\right)$$

- Our beliefs change:

$$\beta_x^1 = \beta_x^0 + \beta^W$$

$$\theta_x^1 = \frac{\beta_x^0 \theta_x^0 + \beta^W y_x^1}{\beta_x^0 + \beta^W}$$

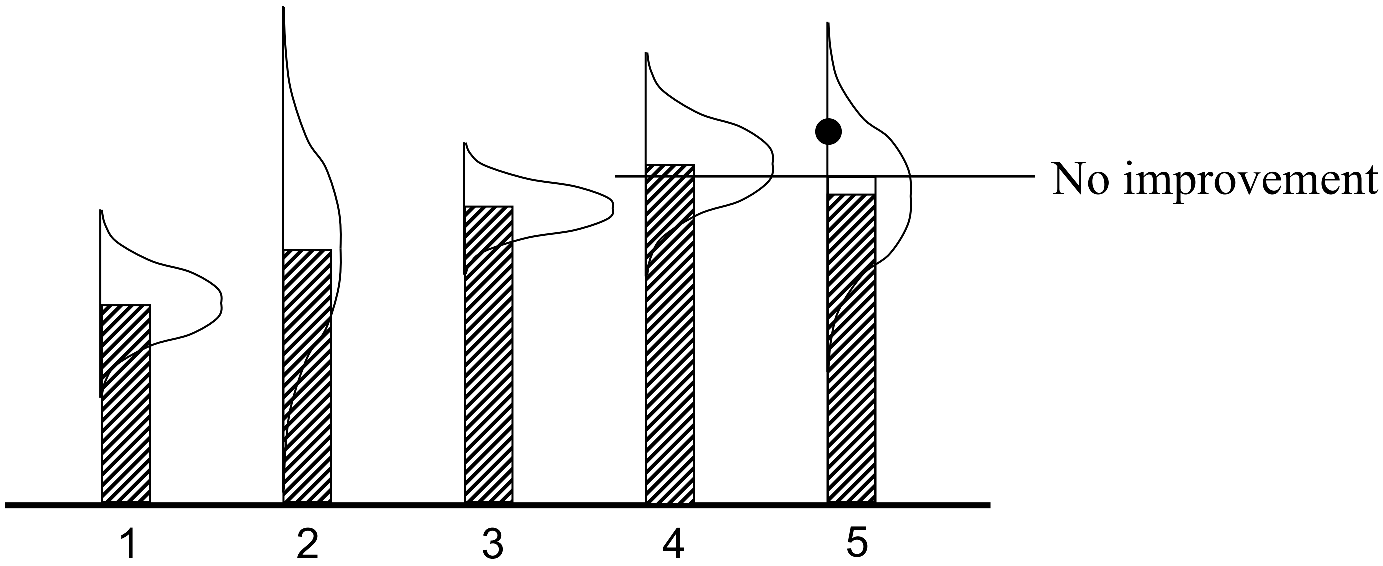
$$\mu_x \sim N\left(\theta_x^1, 1/\beta_x^1\right)$$



- Thus, our beliefs about the rewards are gradually improved over measurements

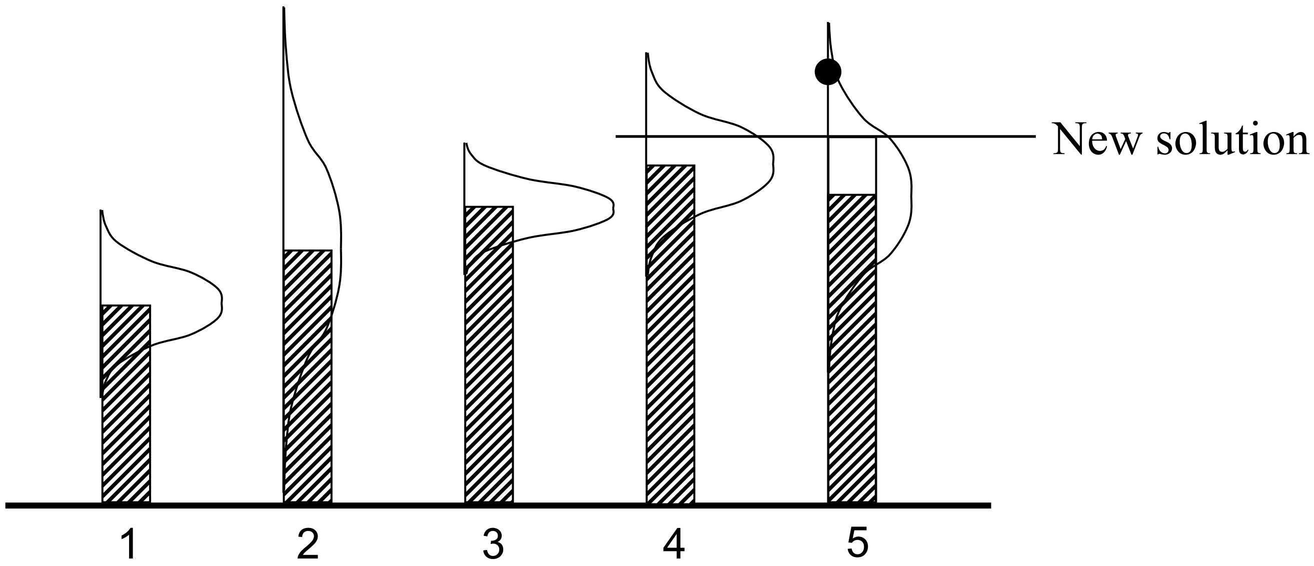
Optimal learning

- Now assume we have five choices, with uncertainty in our belief about how well each one will perform.
- If you can make one measurement, which would you measure?



Optimal learning

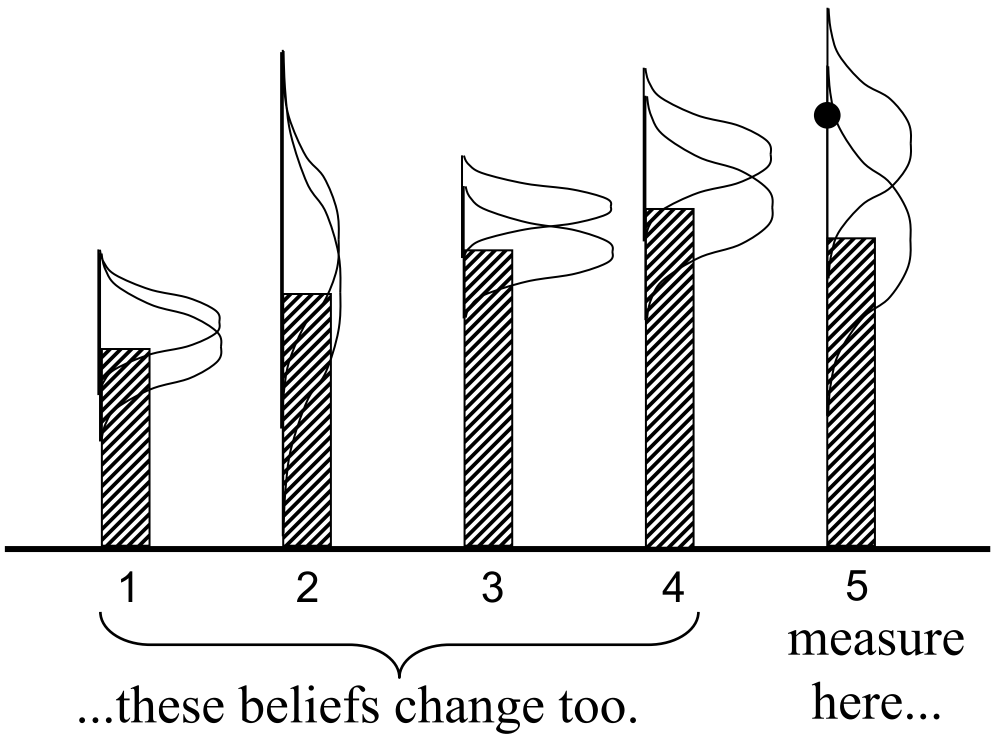
- Now assume we have five choices, with uncertainty in our belief about how well each one will perform.
- If you can make one measurement, which would you measure?



The value of learning is that it may change your decision.

Optimal learning

- An important problem class involves *correlated beliefs* – measuring one alternative tells us something other alternatives.



Optimal learning with a physical state

- The knowledge gradient
  - » The knowledge gradient is the expected value of a single measurement *x*, given by

$$\nu_x^{KG,n} = E^n\left\{\max_y F(y, K^{n+1}(x))\right\} - \max_y F(y, K^n)$$

Marginal value of measuring *x* (the knowledge gradient)

New optimization problem

Expectation over different measurement outcomes

Knowledge state

Implementation decision

Optimization problem given what we know

Updated knowledge state given measurement *x*

  - » Knowledge gradient policy chooses the measurement with the highest marginal value.

The knowledge gradient

- Computing the knowledge gradient for Gaussian beliefs
  - » The change in variance can be found to be

$$\tilde{\sigma}_x^{2,n} = Var\left[\theta_x^{n+1} - \theta_x^n \mid K^n\right]$$
$$= \sigma_x^{2,n} - \sigma_x^{2,n+1}$$

  - » Next compute the *normalized influence*:

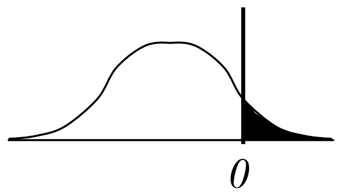
$$\zeta_x^n = -\left|\frac{\theta_x^n - \max_{x' \neq x} \theta_{x'}^n}{\tilde{\sigma}_x^n}\right|$$

Comparison to other alternatives

  - » Let $f(\zeta) = \zeta\Phi(\zeta) + \phi(\zeta)$

$\Phi(\zeta)$ = Cumulative standard normal distribution

$\phi(\zeta)$ = Standard normal density



  - » Knowledge gradient is computed using

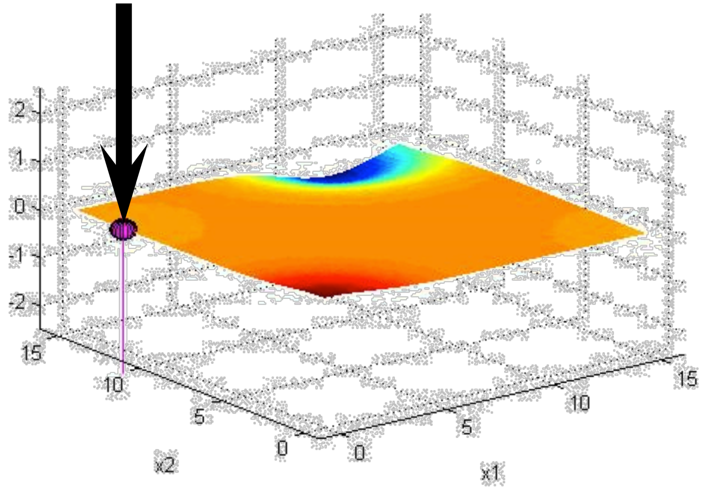
$$\nu_x^{KG} = \tilde{\sigma}_x^n f\left(\zeta_x^n\right)$$

Optimizing storage

- ❑ After four measurements:

Estimated value

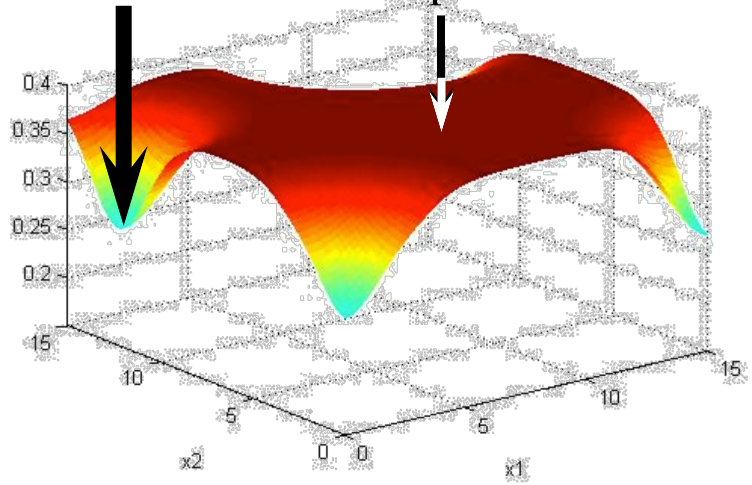
Measurement



Knowledge gradient

Value of another measurement
at same location.

New optimum

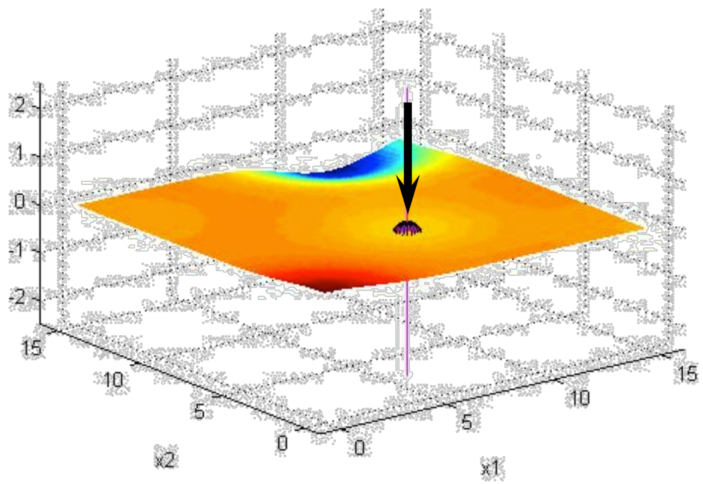


» Whenever we measure at a point, the value of another measurement at the same point goes down. The knowledge gradient guides us to measuring areas of high uncertainty.

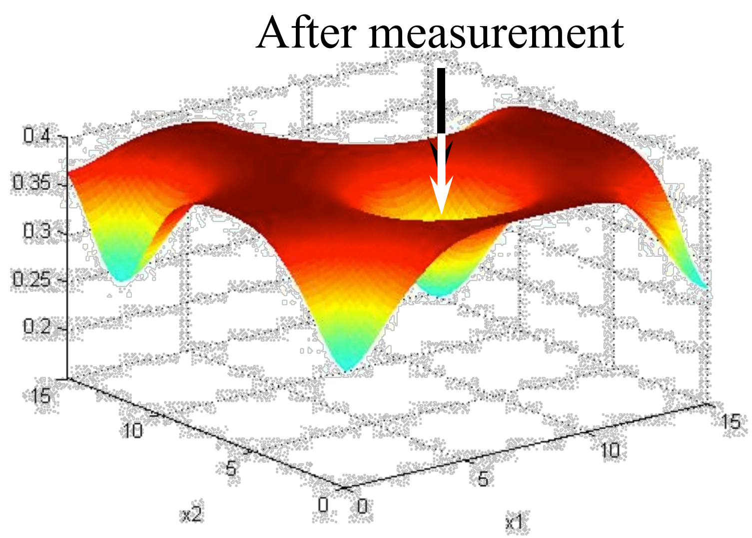
Optimizing storage

- After five measurements:

Estimated value



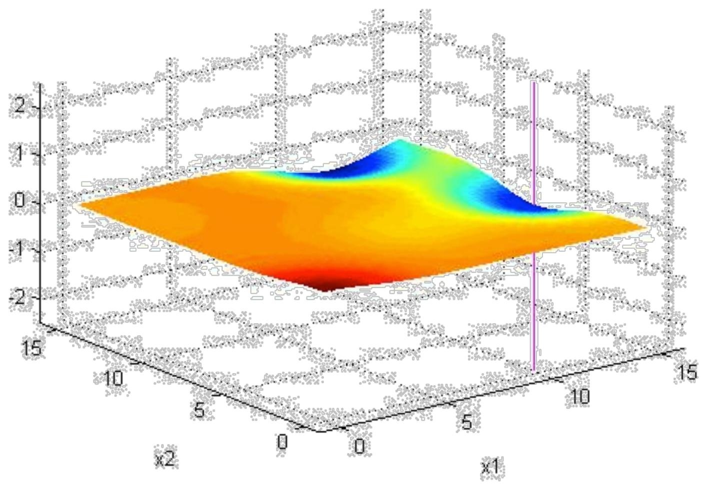
Knowledge gradient



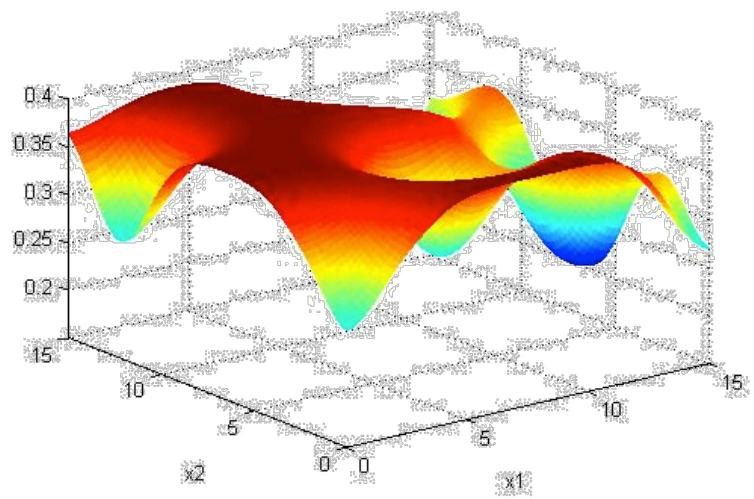
Optimizing storage

- After six samples

Estimated value



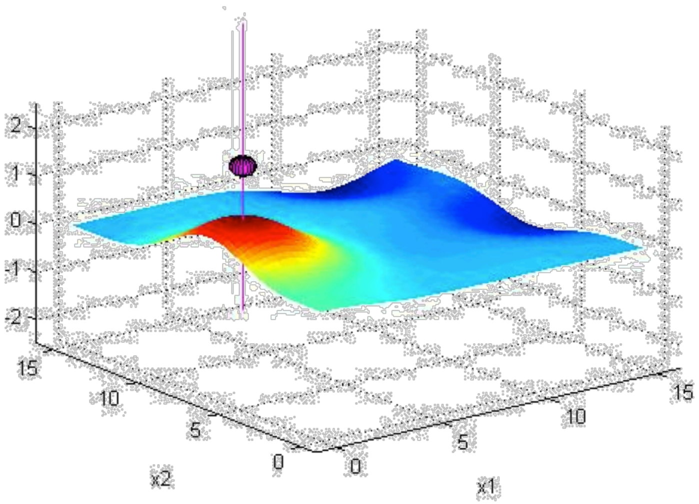
Knowledge gradient



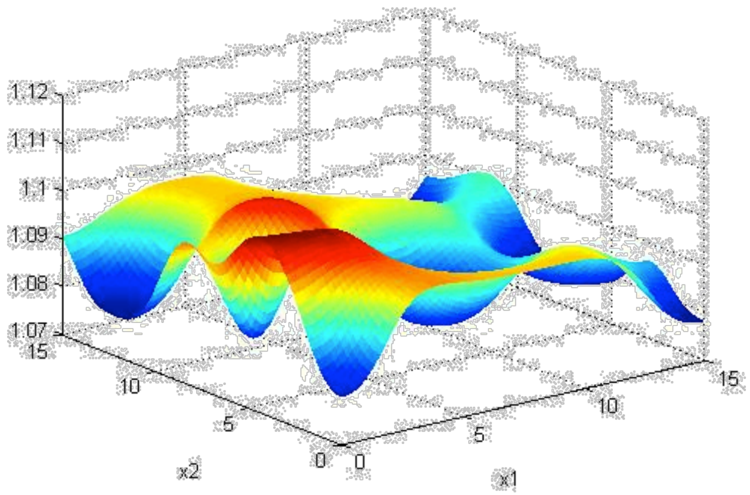
Optimizing storage

- After seven samples

Estimated value



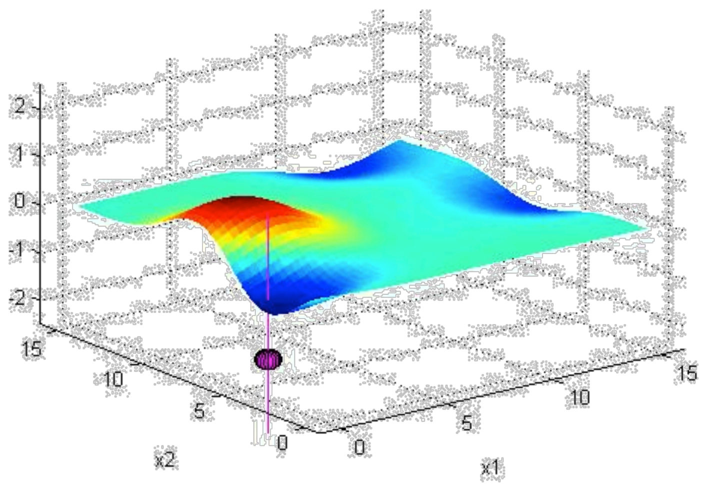
Knowledge gradient



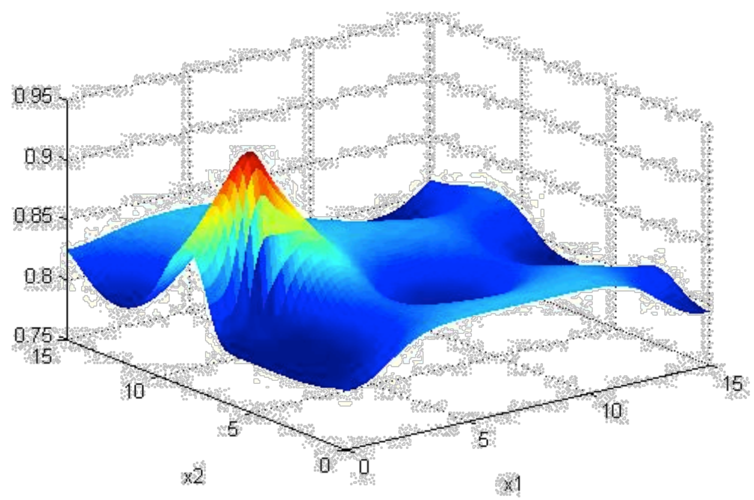
Optimizing storage

- After eight samples

Estimated value



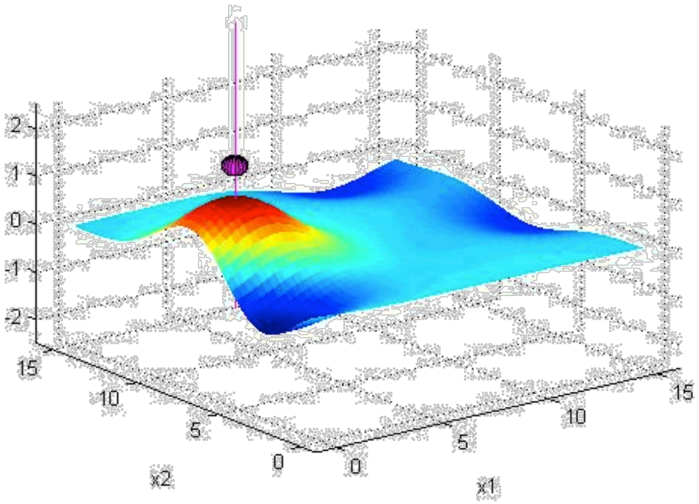
Knowledge gradient



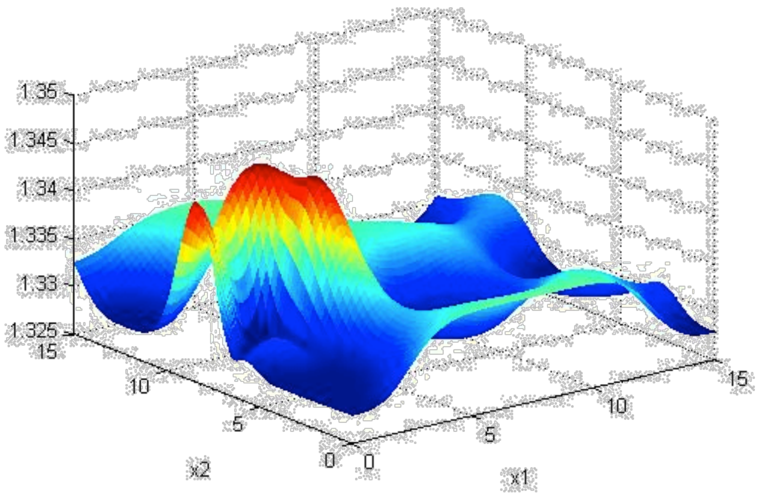
Optimizing storage

- After nine samples

Estimated value



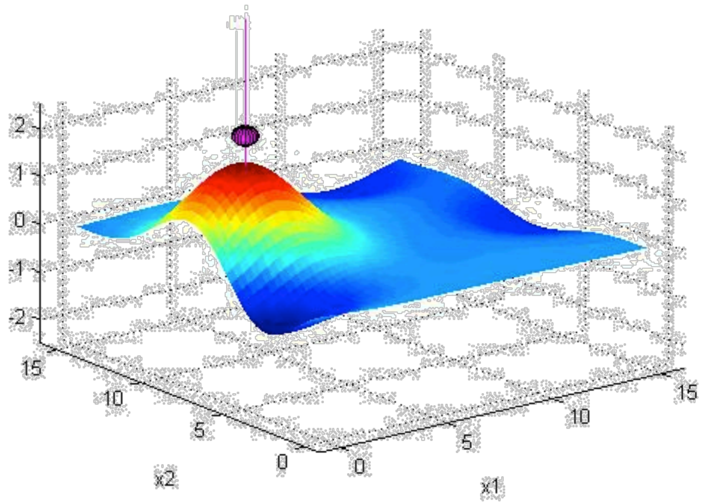
Knowledge gradient



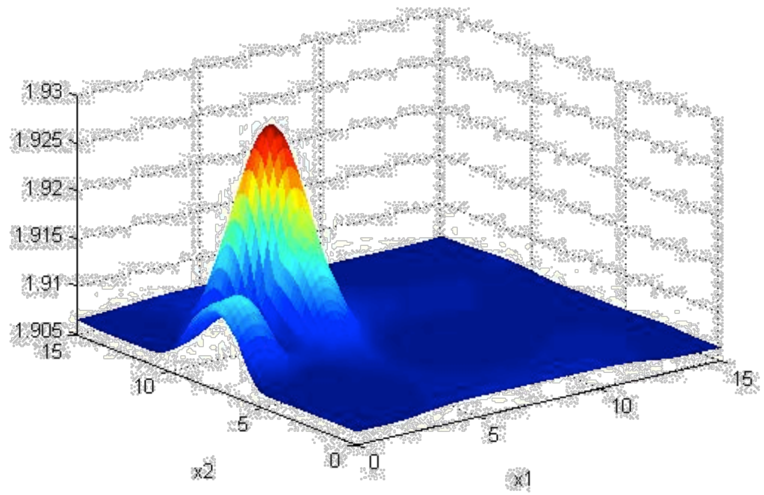
Optimizing storage

- After ten samples

Estimated value



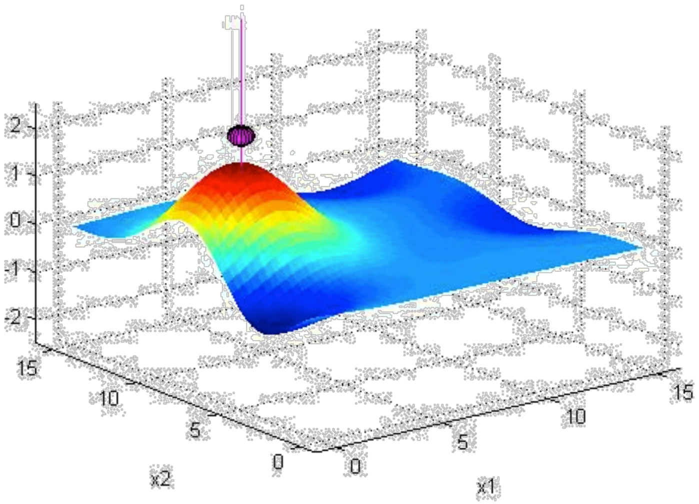
Knowledge gradient



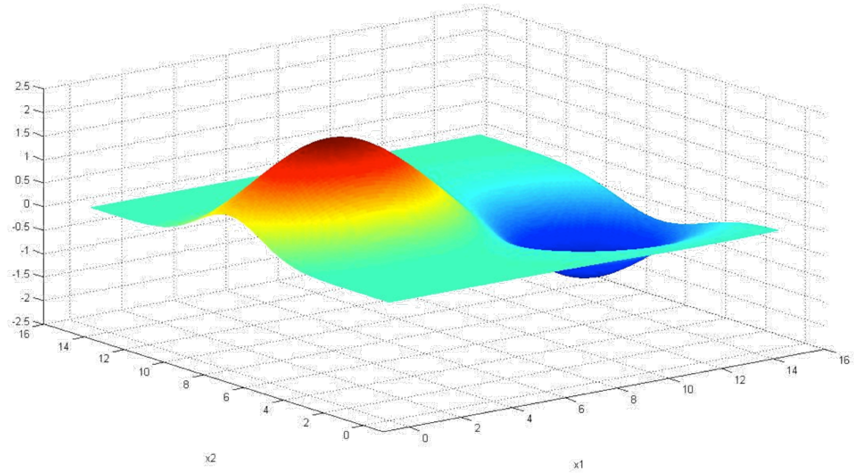
Optimizing storage

- After ten samples, our estimate of the surface:

Estimated value



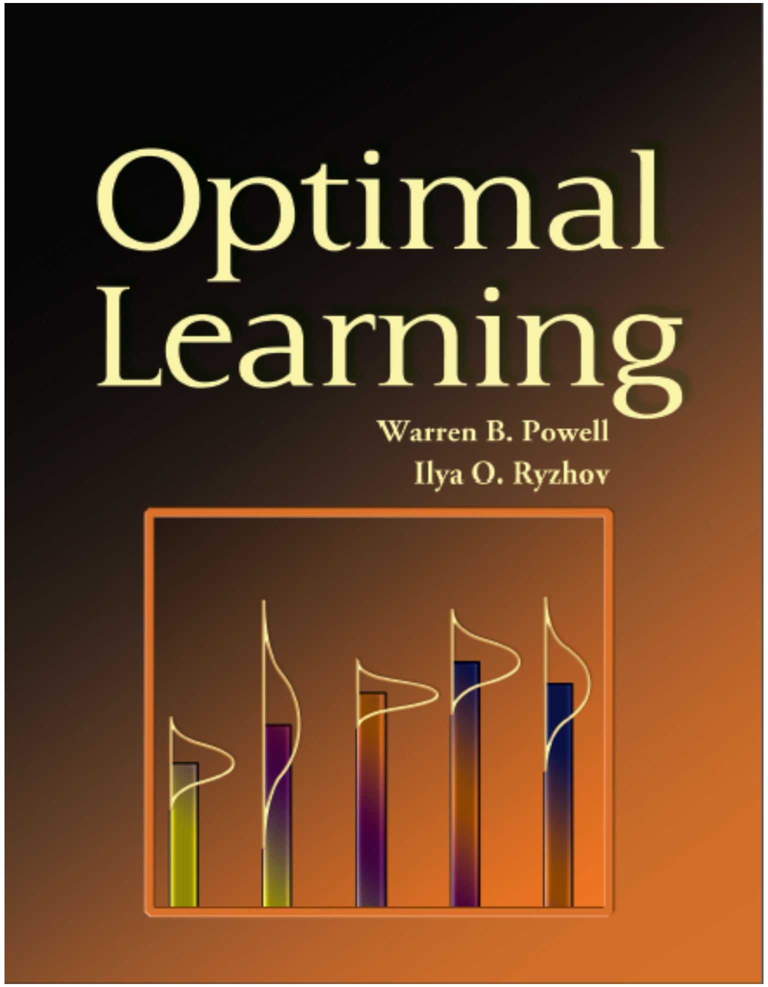
True value



New book!

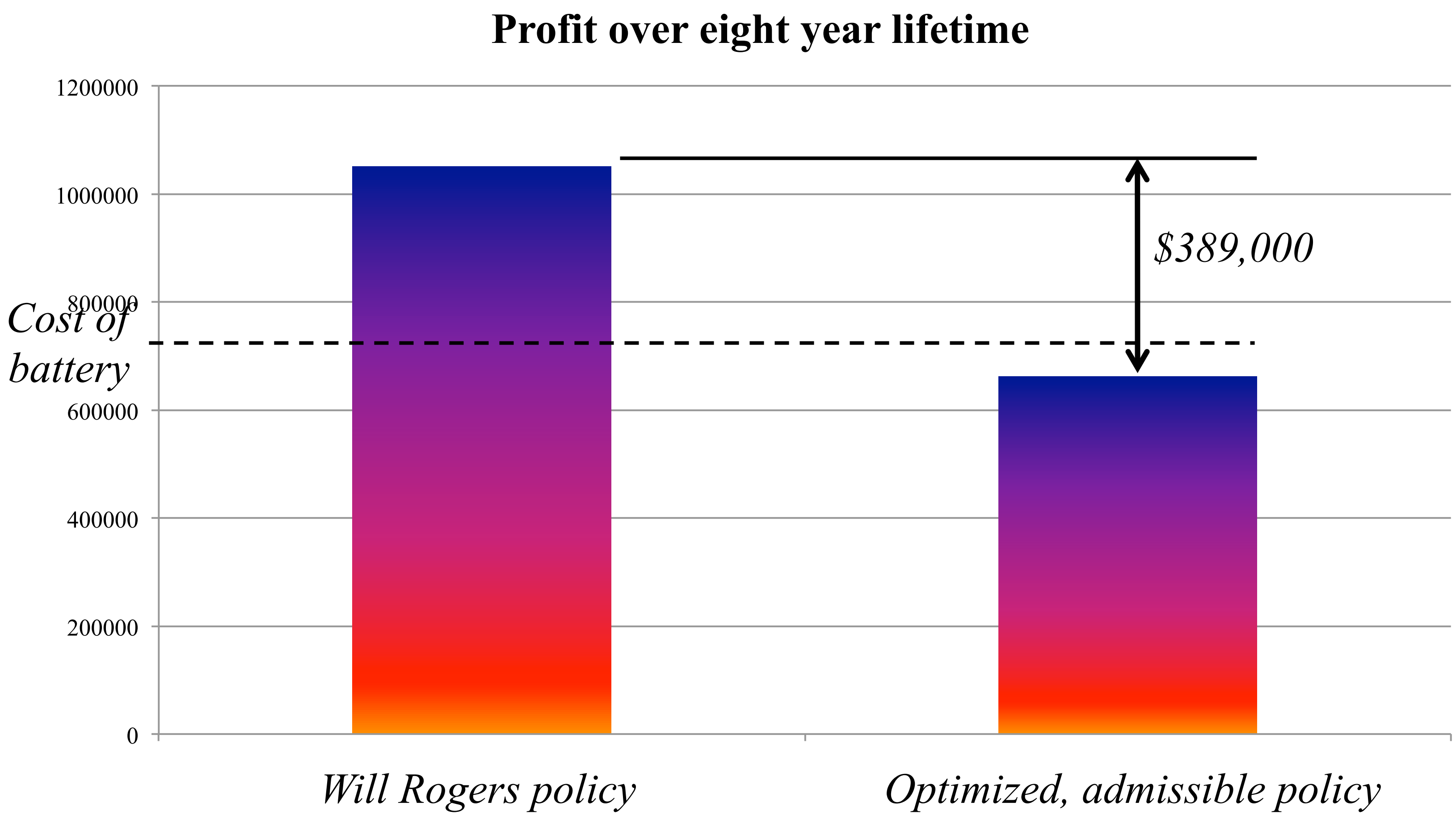
- New book on *Optimal Learning*
  - » Published by John Wiley
  - » First 12 chapters are at an advanced undergraduate level.
- Synthesizes communities:
  - » Ranking and selection
  - » Bandit (Gittins and UCB)
  - » Stochastic search
  - » Simulation optimization
  - » Global optimization
  - » Special focus on knowledge gradient

http://optimallearning.princeton.edu/



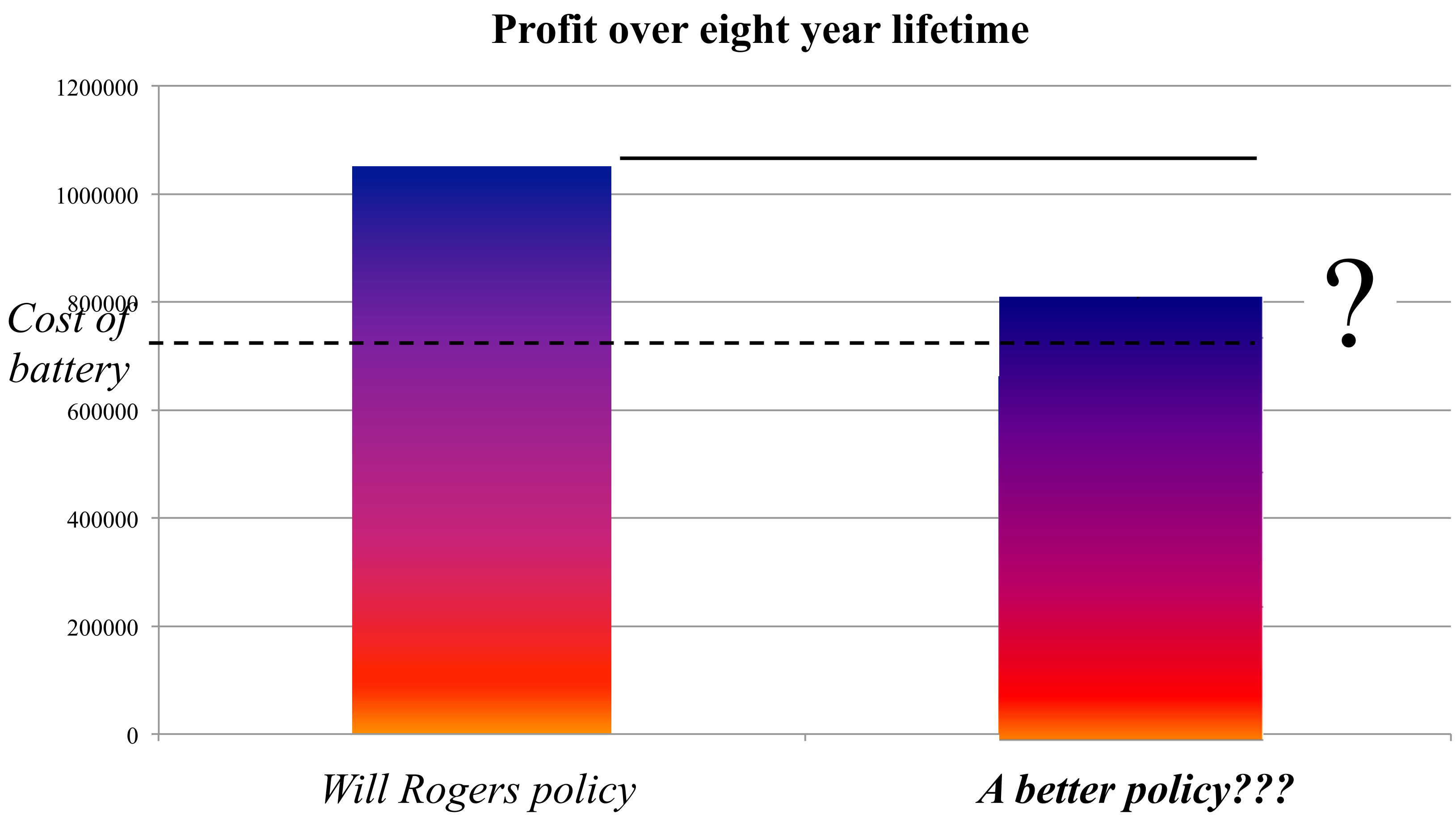
Optimizing storage

- The value of perfect information



Optimizing storage

- The value of perfect information



Lecture outline

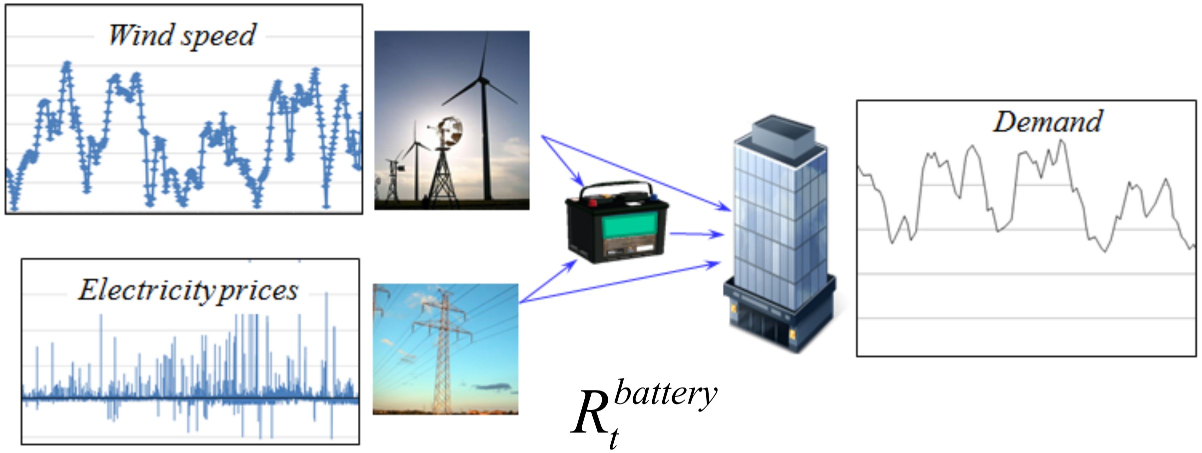
- Balancing energy from wind and the grid using a value function approximation

The test problem

- Energy storage with stochastic prices, supplies and demands.

E_t^{wind}

P_t^{grid}



$R_t^{battery}$

D_t

$$E_{t+1}^{wind} = E_t^{wind} + \hat{E}_{t+1}^{wind}$$

$$P_{t+1}^{grid} = P_t^{grid} + \hat{P}_{t+1}^{grid}$$

$$D_{t+1}^{load} = D_t^{load} + \hat{D}_{t+1}^{load}$$

$$R_{t+1}^{battery} = R_t^{battery} + Ax_t$$

W_{t+1} = Exogenous inputs

S_t = State variable

x_t = Controllable inputs

The test problem

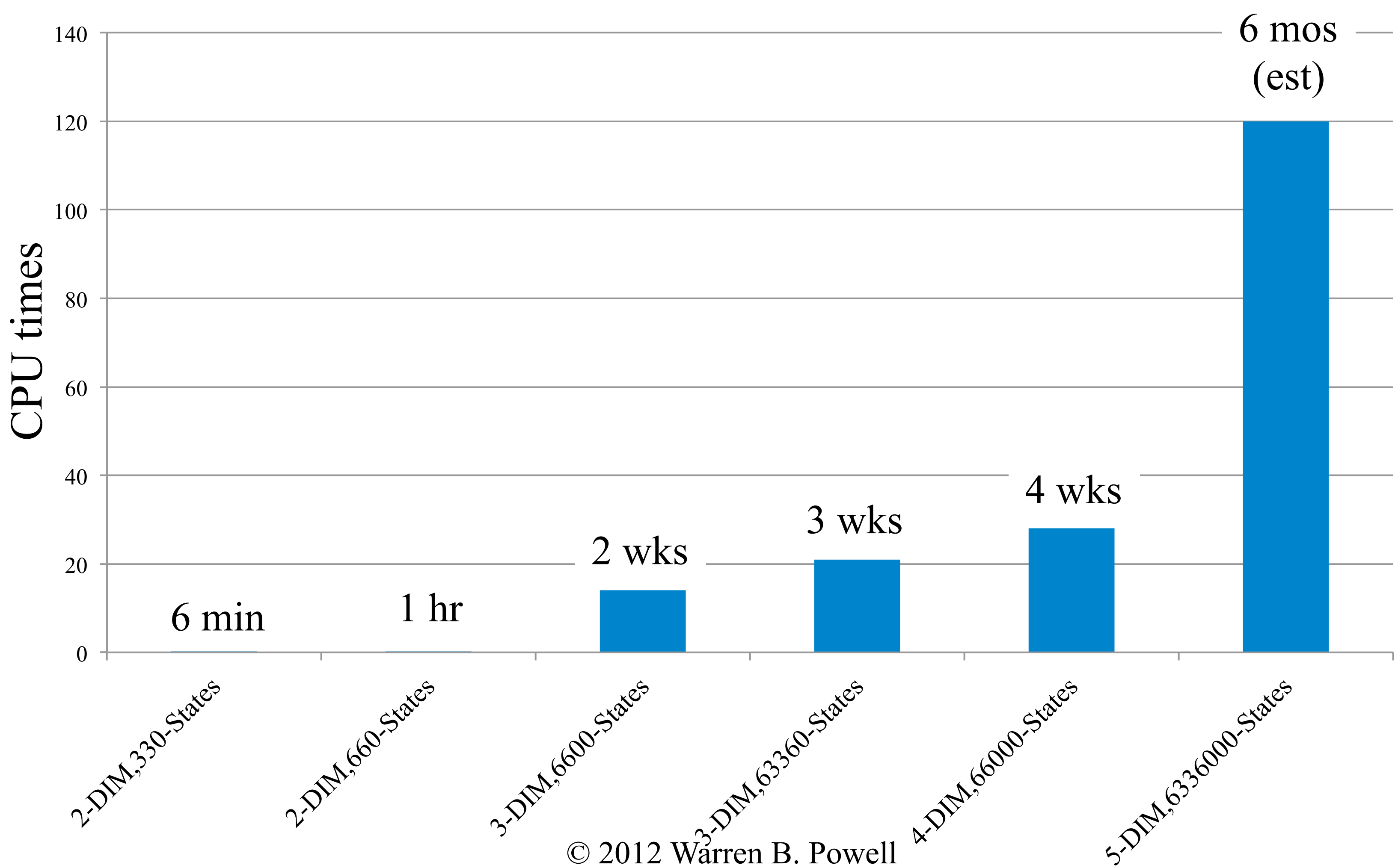
- Bellman's optimality equation

$$V(S_t) = \min_{x_t \in \mathcal{X}} \left(C(S_t, x_t) + \gamma \mathbb{E} V(S_{t+1}(S_t, x_t, W_{t+1})) \right)$$

$$S_t = \begin{bmatrix} E_t^{wind} \\ P_t^{grid} \\ D_t^{load} \\ R_t^{battery} \end{bmatrix} \qquad x_t = \begin{bmatrix} x_t^{wind-battery} \\ x_t^{wind-load} \\ x_t^{grid-battery} \\ x_t^{grid-load} \\ x_t^{battery-load} \end{bmatrix} \qquad W_{t+1} = \begin{bmatrix} \hat{E}_{t+1}^{wind} \\ \hat{P}_{t+1}^{grid} \\ \hat{D}_{t+1}^{load} \end{bmatrix}$$

The curse of dimensionality

- Finding an optimal solution using exact methods:



Approximate value iteration

Step 1: Start with a pre-decision state S_t^n

Step 2: Solve the deterministic optimization using an approximate value function:

$$\hat{v}_t^n = \min_x \left(C_t(S_t^n, x_t) + \bar{V}_t^{n-1}(S^{M,x}(S_t^n, x_t)) \right)$$

to obtain x_t^n.

Deterministic optimization

Step 3: Update the value function approximation

$$\bar{V}_{t-1}^n(S_{t-1}^{x,n}) = (1-\alpha_{n-1})\bar{V}_{t-1}^{n-1}(S_{t-1}^{x,n}) + \alpha_{n-1}\hat{v}_t^n$$

Recursive statistics

Step 4: Obtain Monte Carlo sample of $W_t(\omega^n)$ and compute the next pre-decision state:

$$S_{t+1}^n = S^M(S_t^n, x_t^n, W_{t+1}(\omega^n))$$

Simulation

Step 5: Return to step 1.

Approximate policy iteration

Step 1: Start with a pre-decision state S_t^n

Step 2: Inner loop: Do for m=1,…,M:

Step 2a: Solve the deterministic optimization using an approximate value function:

$$\hat{v}^m = \min_x \left(C(S^m, x) + \bar{V}^{n-1}(S^{M,x}(S^m, x)) \right)$$

to obtain x^m.

Step 2b: Update the value function approximation

$$\bar{V}^{n-1,m}(S^{x,m}) = (1-\alpha_{m-1})\bar{V}^{n-1,m-1}(S^{x,m}) + \alpha_{m-1}\hat{v}^m$$

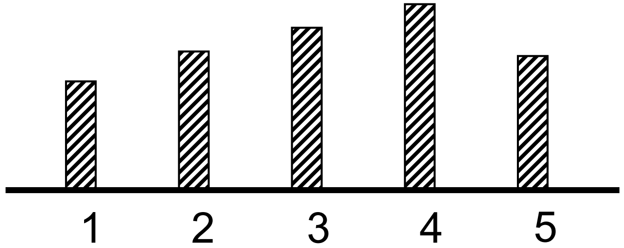
Step 2c: Obtain Monte Carlo sample of $W(\omega^m)$ and compute the next pre-decision state:

$$S^{m+1} = S^M(S^m, x^m, W(\omega^m))$$

Step 3: Update $\bar{V}^n(S)$ using $\bar{V}^{n-1,M}(S)$ and return to step 1.

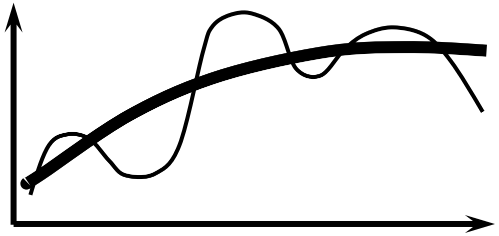
Approximating functions

- Approximation architectures
  - » Lookup tables - one value for each input variable.

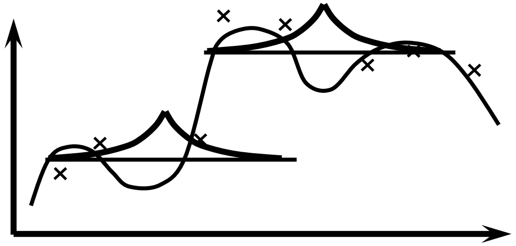


  - » Parametric models

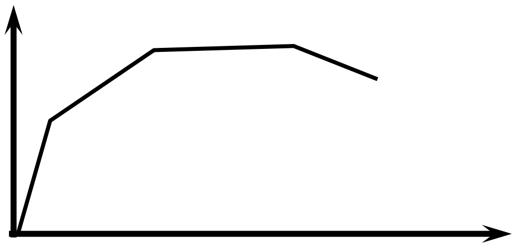
$$\bar{V}(s) = \theta_0 + \theta_1\phi_1(s) + \theta_2\phi_2(s) + \ldots$$



  - » Nonparametric models



  - » Exploiting structure (convexity)



Approximating the value function

- Notes on computational experience
  - » Most convergence proofs assume lookup tables
    - Does not scale!!!
  - » Parametric models are the most attractive

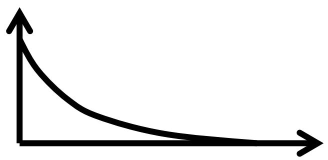
$$V(\bar{S}) = \sum_{f \in \mathcal{F}} \theta_f \phi_f(S)$$

→ "features" = "basis functions"

- Simple, popular, but dangerous.
- Challenge is designing features (the "art" of ADP)
- Can work, but can work very poorly. Use at your own risk!!

  - » Nonparametric models

$$\bar{V}^n(s) = \sum_{i=1}^{n} v^i \frac{k(s,s^i)}{\sum_{j=1}^{n} k(s,s^j)} \qquad k(s,s^i) =$$



- Flexible, but clumsy inside algorithms, and limited to a few dimensions.

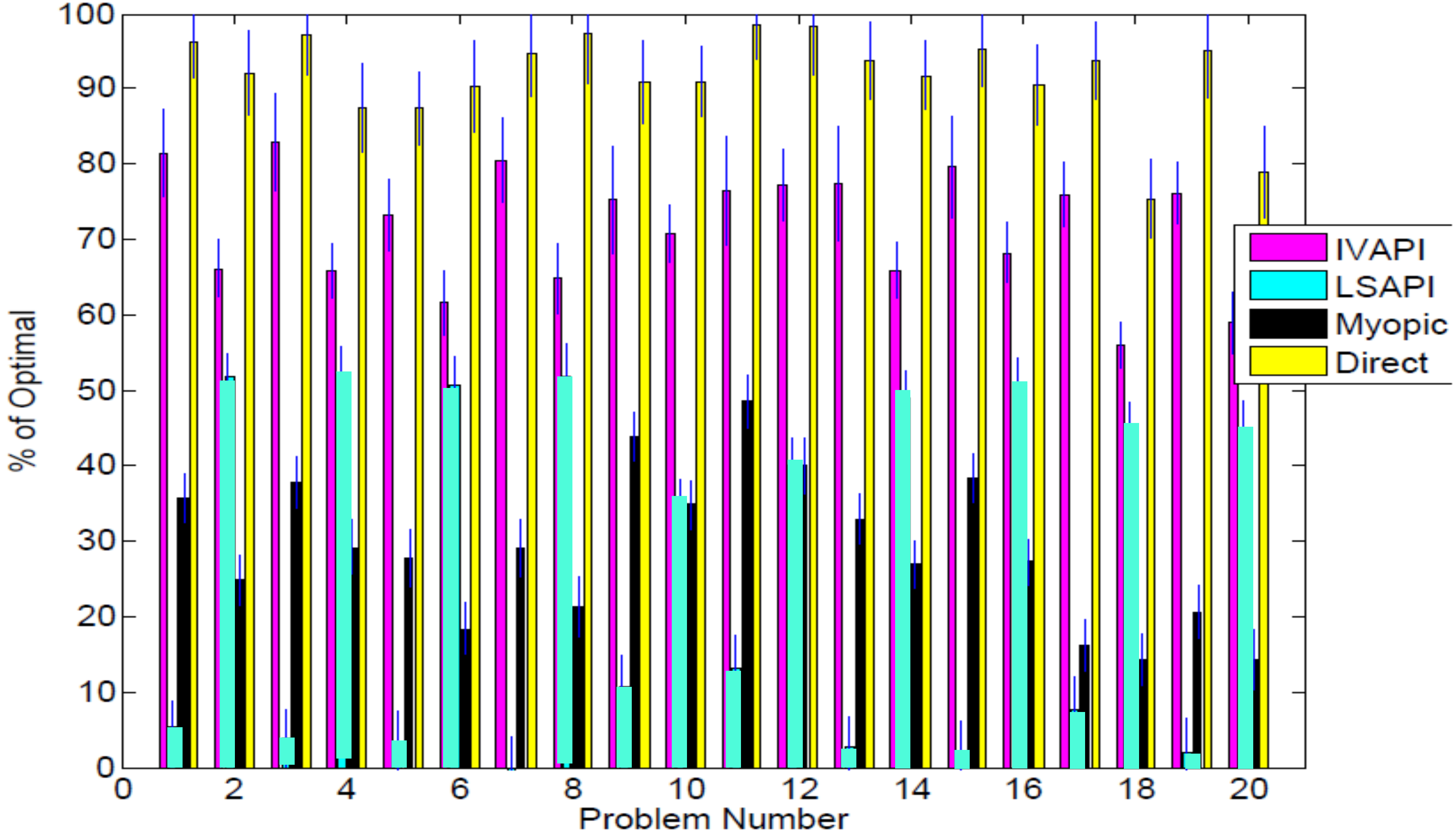
Experiments with energy storage

- Algorithmic strategies
  - 1) Discretized benchmark:

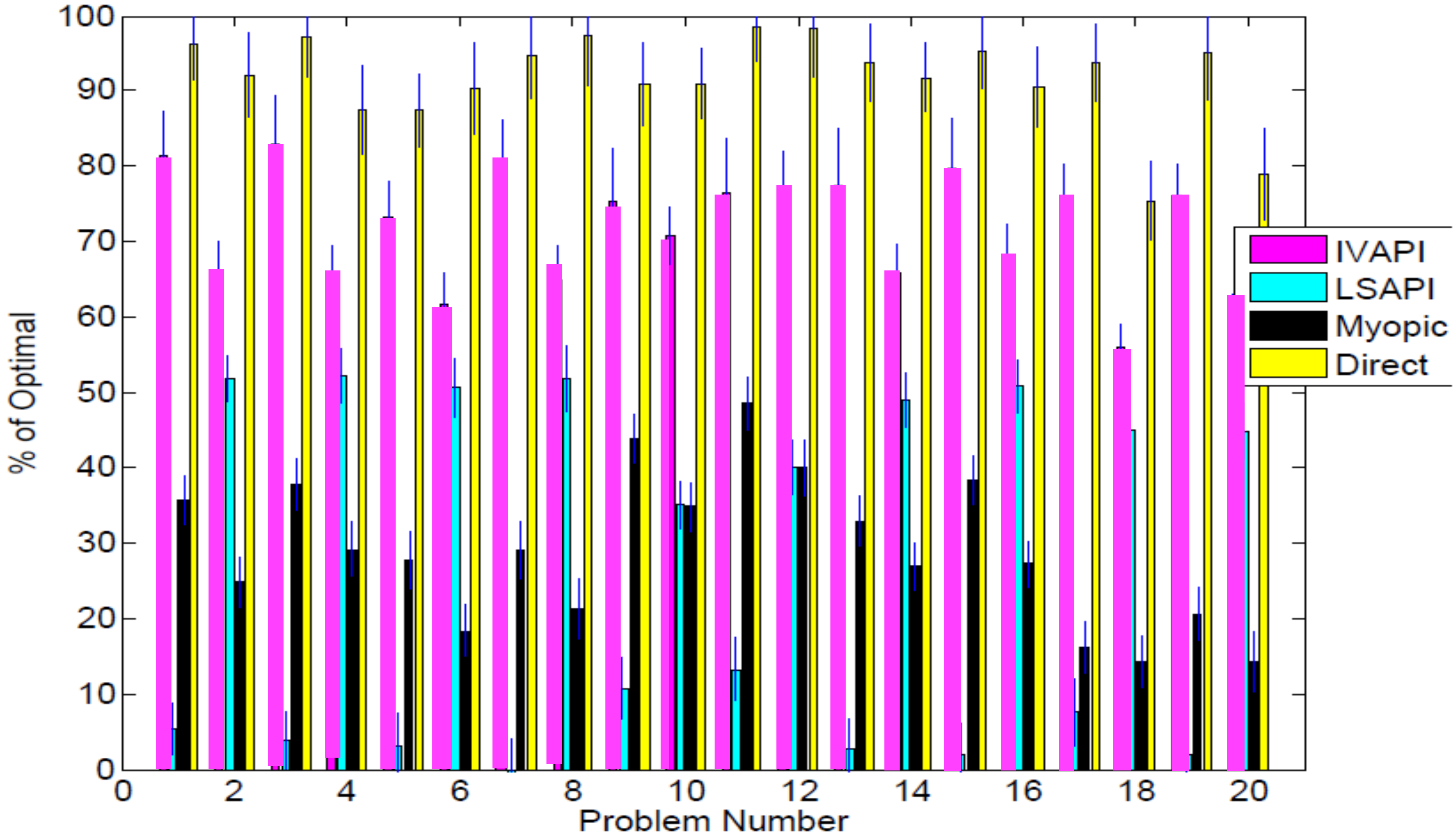
$$V(S_t) = \min_{x_t \in \mathcal{X}} \left(C(S_t, x_t) + \gamma \mathrm{E} V(S_{t+1}(S_t, x_t, W_{t+1})) \right)$$

  - 2) Vanilla ADP using least squares policy iteration (Lagoudakis and Parr)
  - 3) LSPI using instrumental variables
  - 4) Direct policy search (described below)

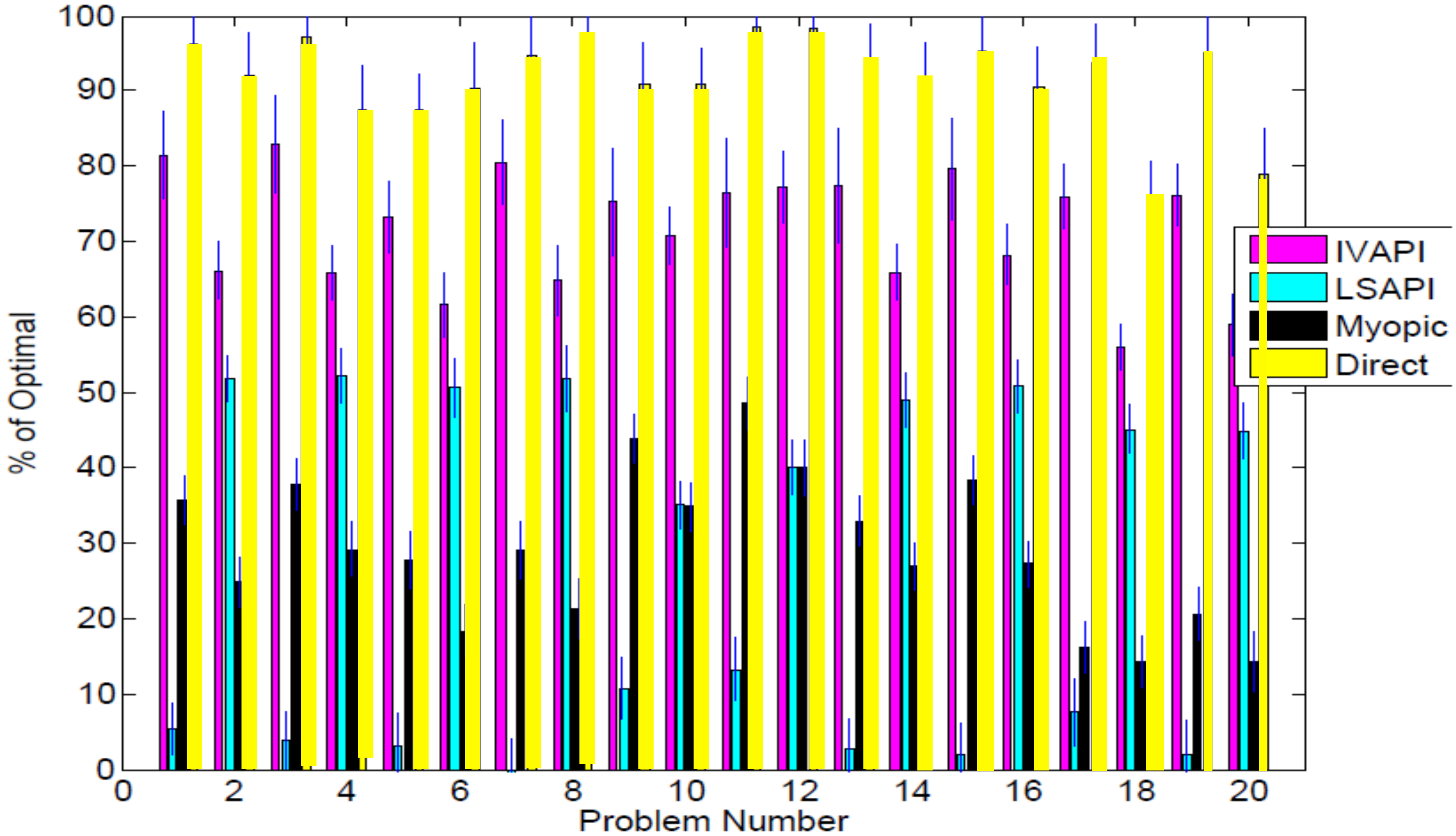
An energy storage application



An energy storage application

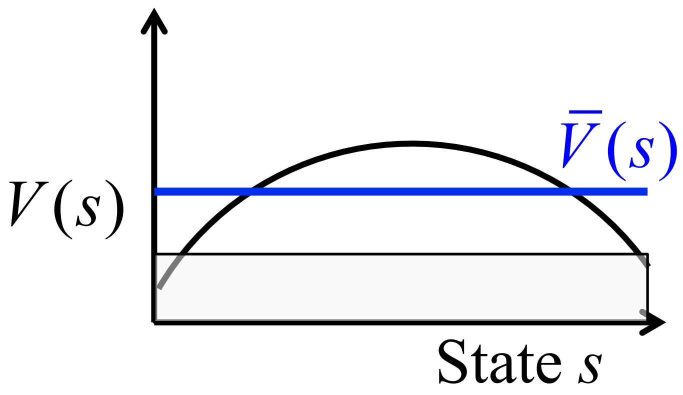


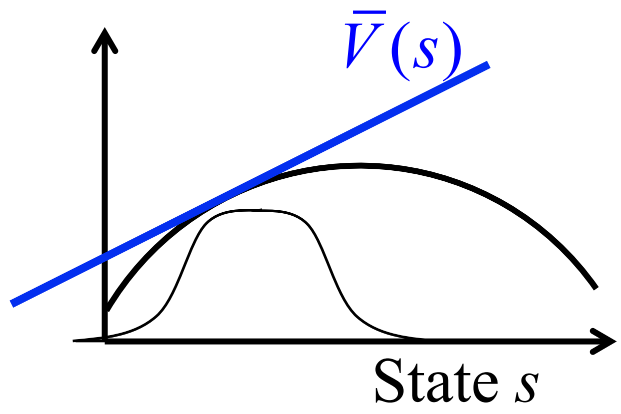
An energy storage application



Approximate dynamic programming

- Why didn't a parametric model work?
  - » Possible theory: We had to use "off policy" state sampling to estimate the regression parameters.

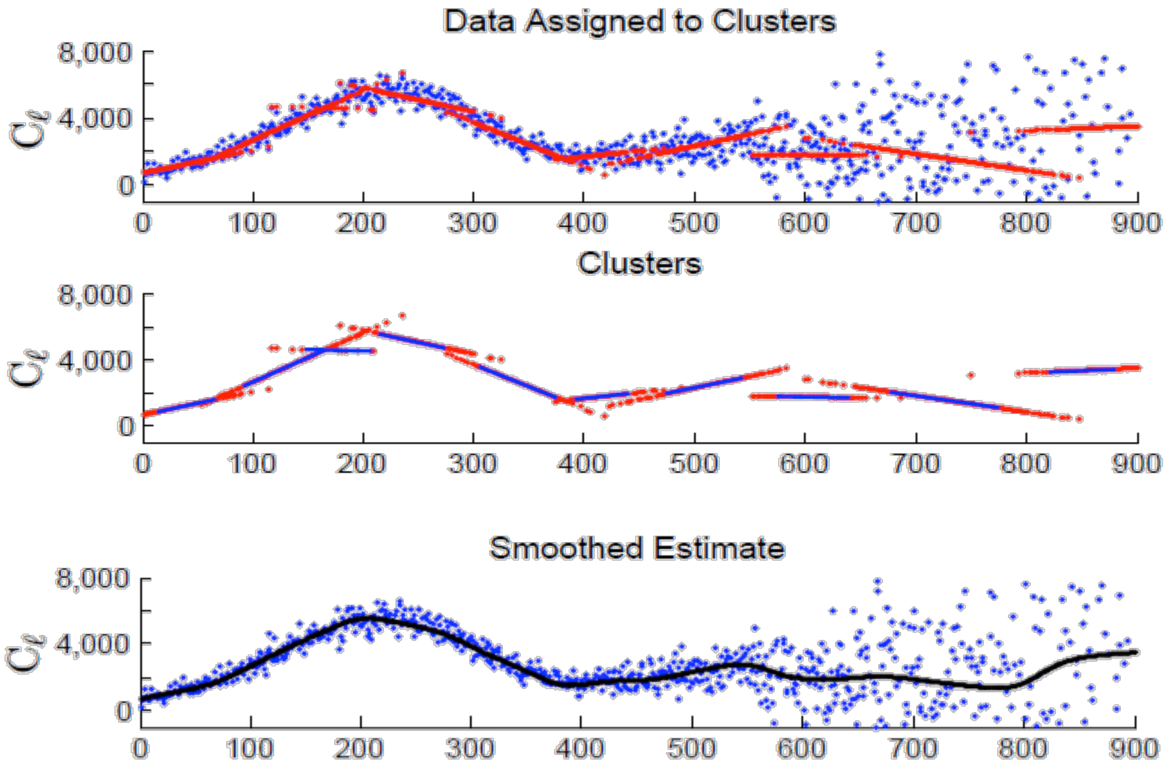




  - » Off-policy learning is *not* a problem if we use lookup tables, and probably not a problem with nonparametric approximations.
  - » Do we just need better approximation strategies?

Dirichlet process mixtures

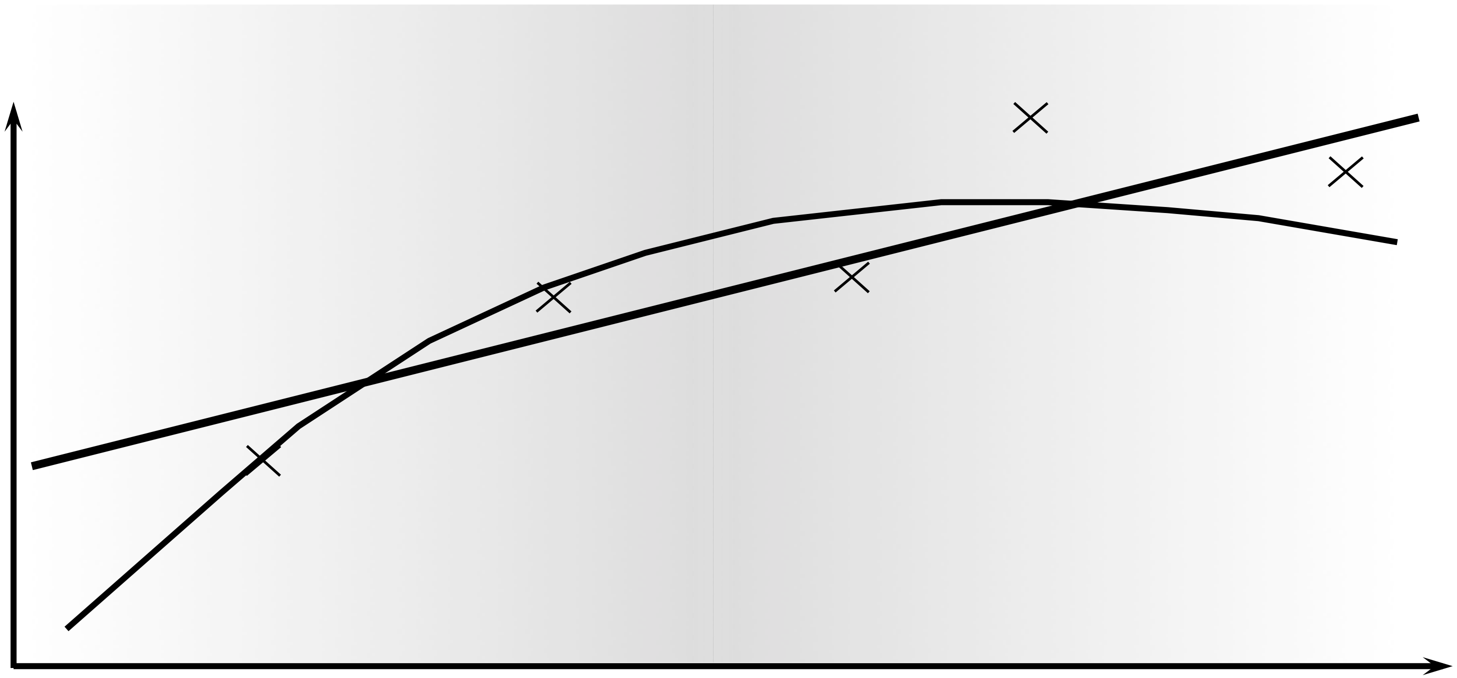
- A semi-parametric method that fits linear models around clusters (L. Hannah, D. Blei and W.B.P.)



» Works well, but clustering step is *very* slow.

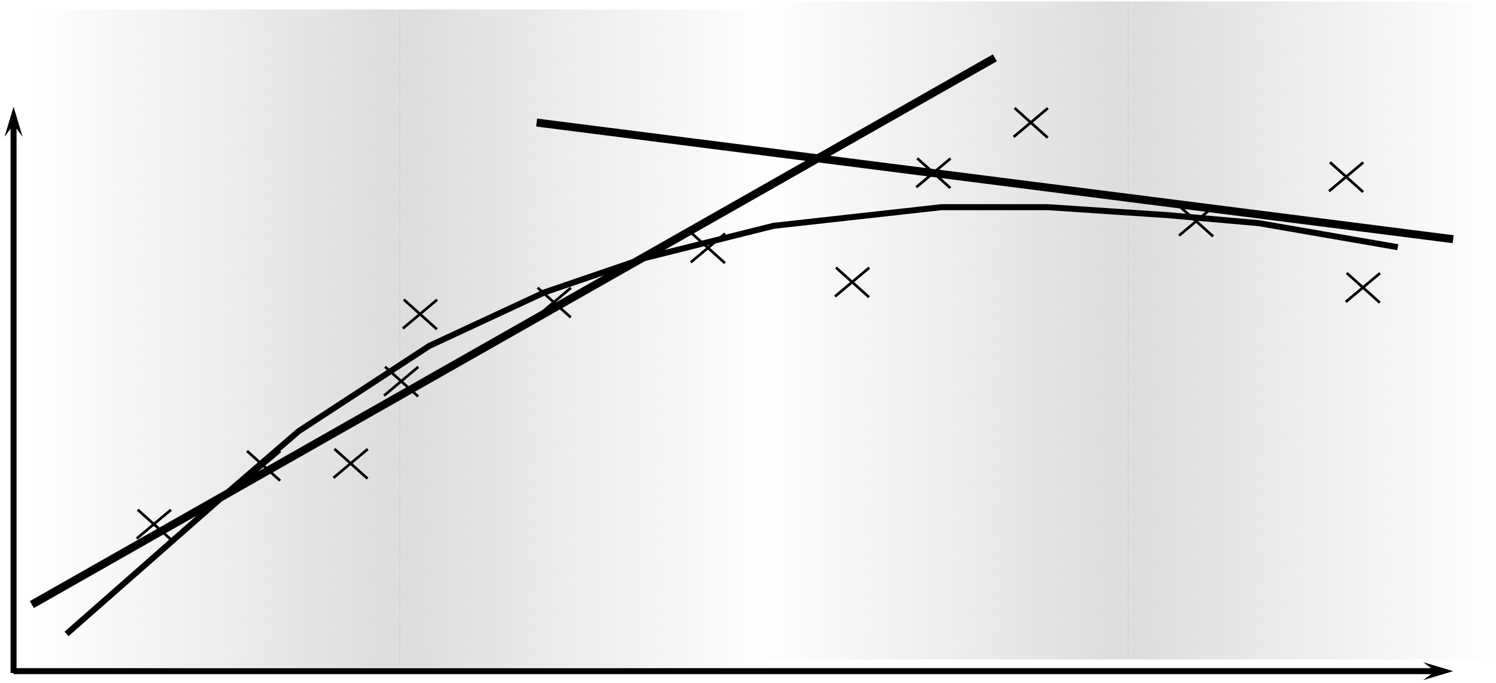
Semi-parametric methods

- New idea – Dirichlet clouds
  - » Old method (DP-GLM) – Retain entire history of observations for clustering.
  - » New method – Retain parameters of Gaussian clouds.
  - » Fit linear models for each cloud.



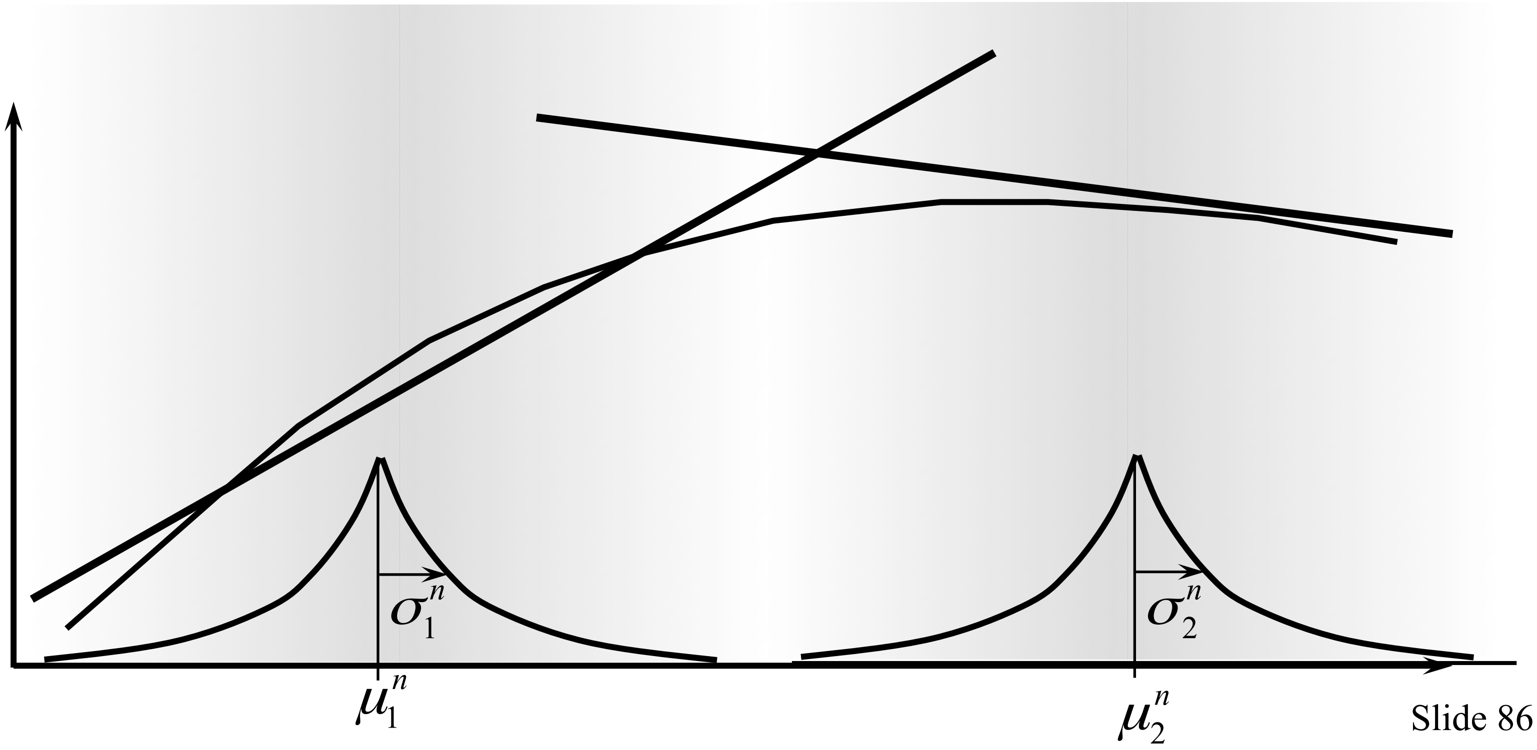
Semi-parametric methods

- New idea – Dirichlet clouds
  - » Old method (DP-GLM) – Retain entire history of observations for clustering.
  - » New method – Retain parameters of Gaussian clouds.
  - » Fit linear models for each cloud.



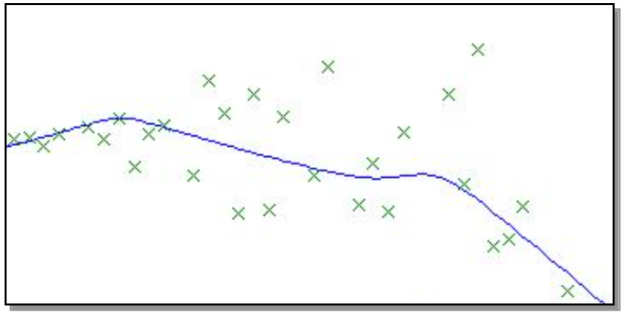
Semi-parametric methods

- New idea – Dirichlet clouds
  - » Old method (DP-GLM) – Retain entire history of observations for clustering.
  - » New method – Retain parameters of Gaussian clouds.
  - » Fit linear models for each cloud.

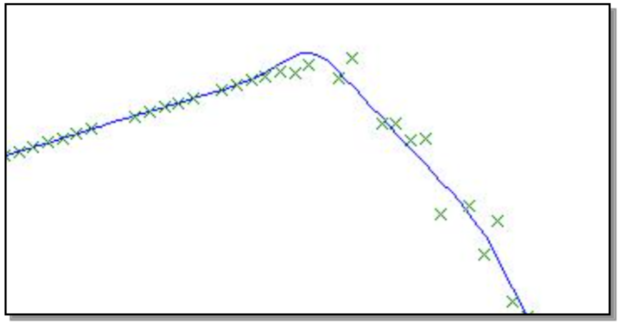


Semi-parametric methods

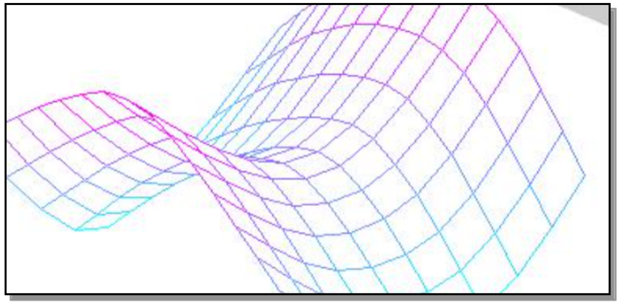
- Videos
  - » Scalar function – curved



  - » Scalar function – hockey stick



  - » Two-dimensional surface

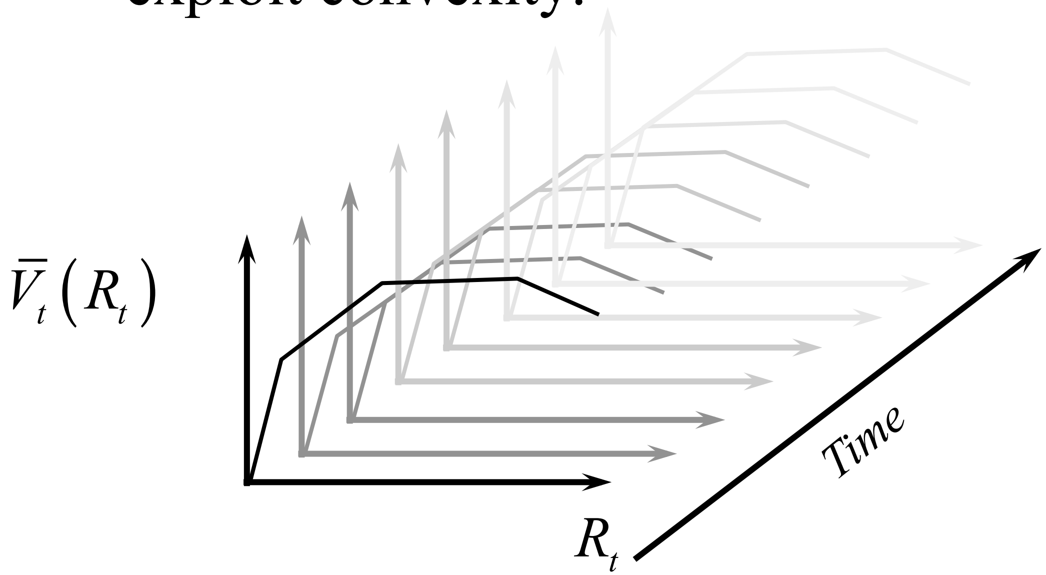


Approximate dynamic programming

- The challenge of time dependent problems
  - » Real energy storage problems are highly time dependent

$$V(\bar{S}) = \sum_{f \in \mathcal{F}} \theta_f \phi_f(S) \Rightarrow V(\bar{S}) = \sum_{f \in \mathcal{F}} \theta_{tf} \phi_f(S)$$

  - » Makes direct policy search impossible, but easy to do with Bellman error minimization, especially if we exploit convexity.



Video

Approximating dynamic programming

- What if someone gives us a forecast of supply (energy from wind) or demand?
  - » Basic storage problem

    $S_t = R_t$ = Amount of energy in storage
  - » What if we are give a forecast of future demands?

    $f_t^D = (f_{t,t+1}^D, f_{t,t+2}^D, ..., f_{t,t+T}^D)$ = Forecast by time period
- Strategies for handling a forecast:
  - » Add it to the state variable:

    $S_t = \left(R_t, f_t^D\right)$ Very hard to approximate $\bar{V}(S_t)$
  - » Imbed it in the expectation:

    $\mathrm{E}\,V(S) \Rightarrow \mathrm{E}_{f^D} V(R)$ But now we have to recompute $\bar{V}(R)$

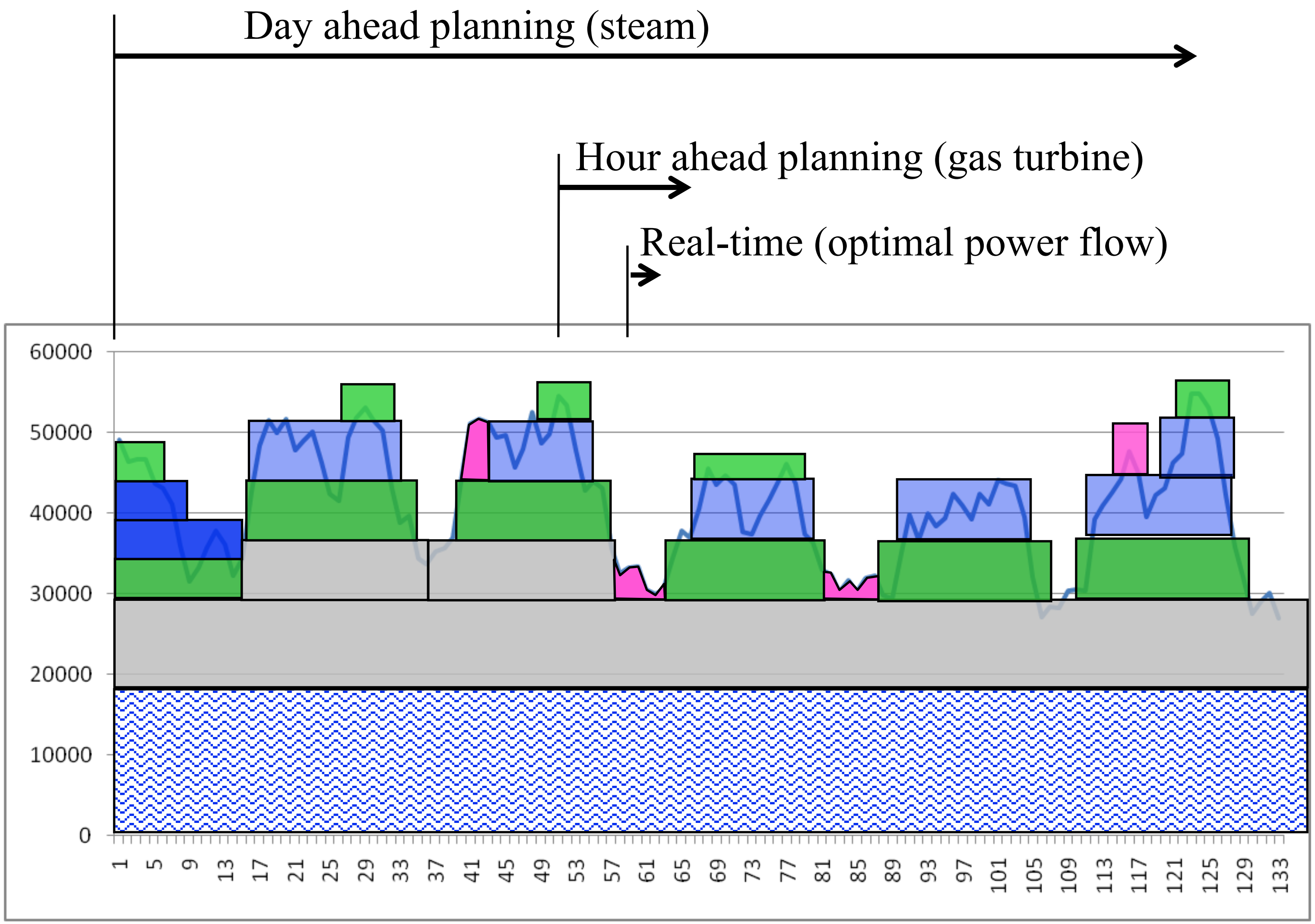
    if we change the forecast.

Lecture outline

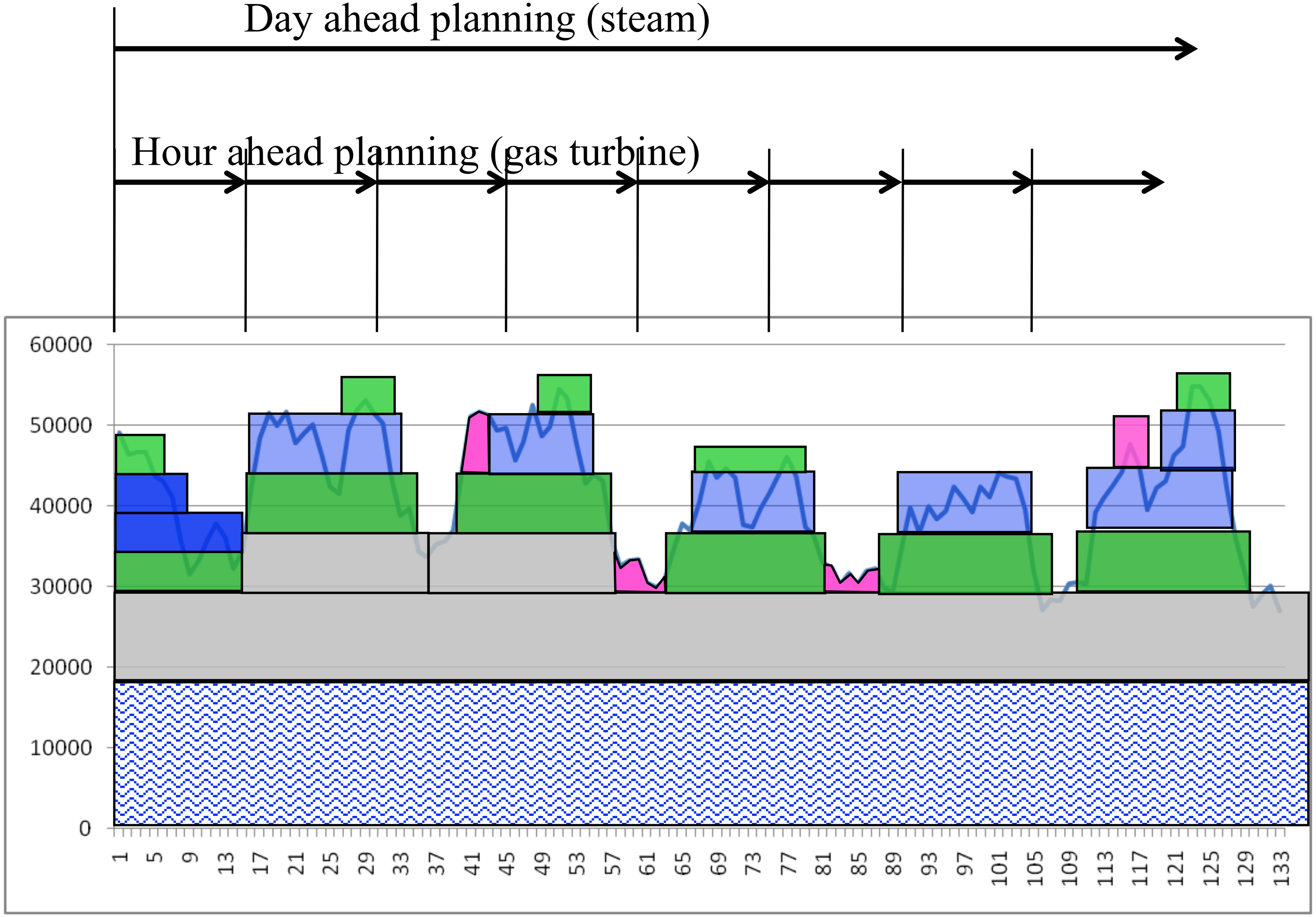
- The stochastic unit commitment problem using a hybrid lookahead and cost function approximation


Legend
Backbone Transmission
Voltage (kV)
345
500
765
Backbone Substations
Voltage (kV)
345
500
765

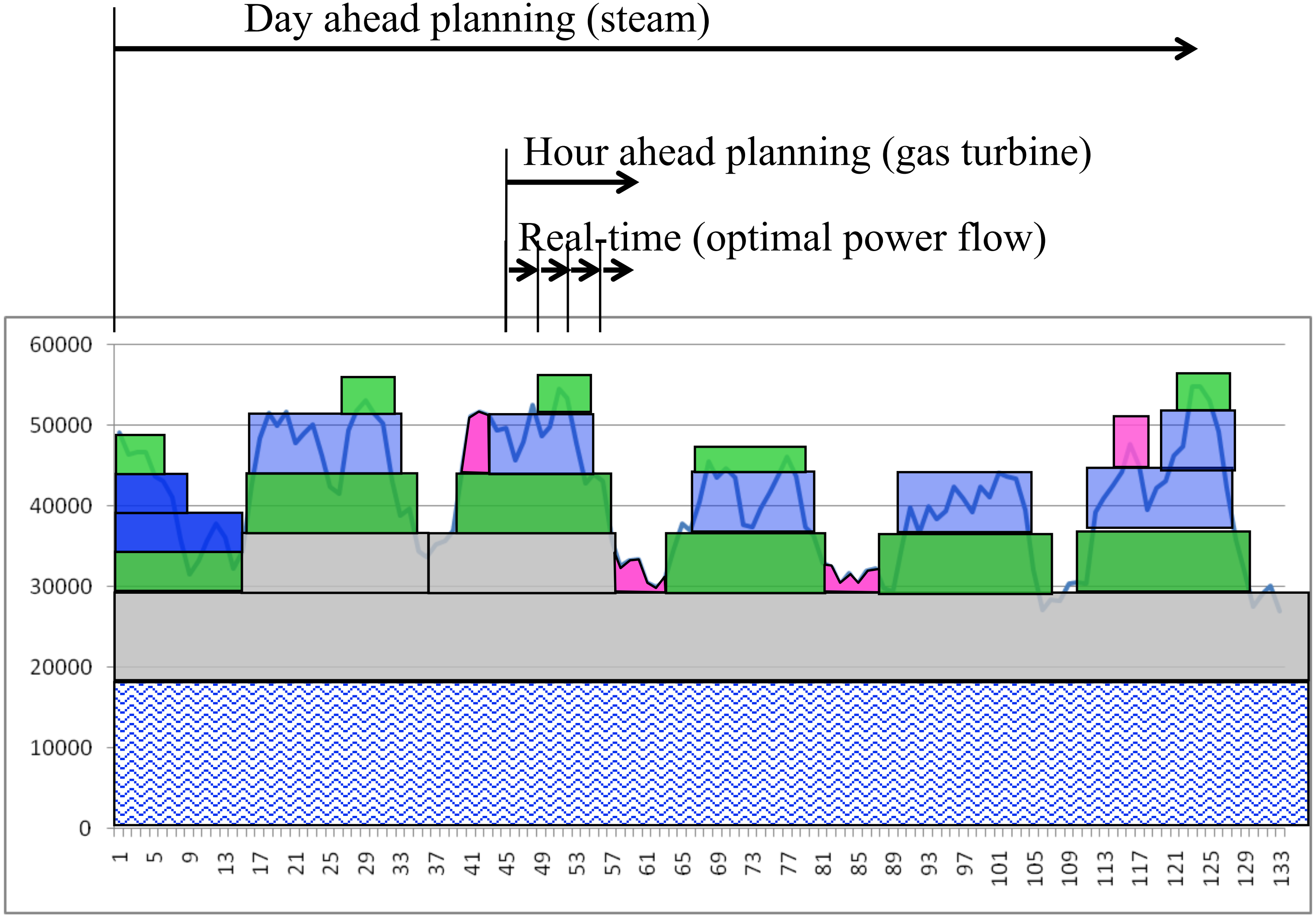
The time-staging of decisions



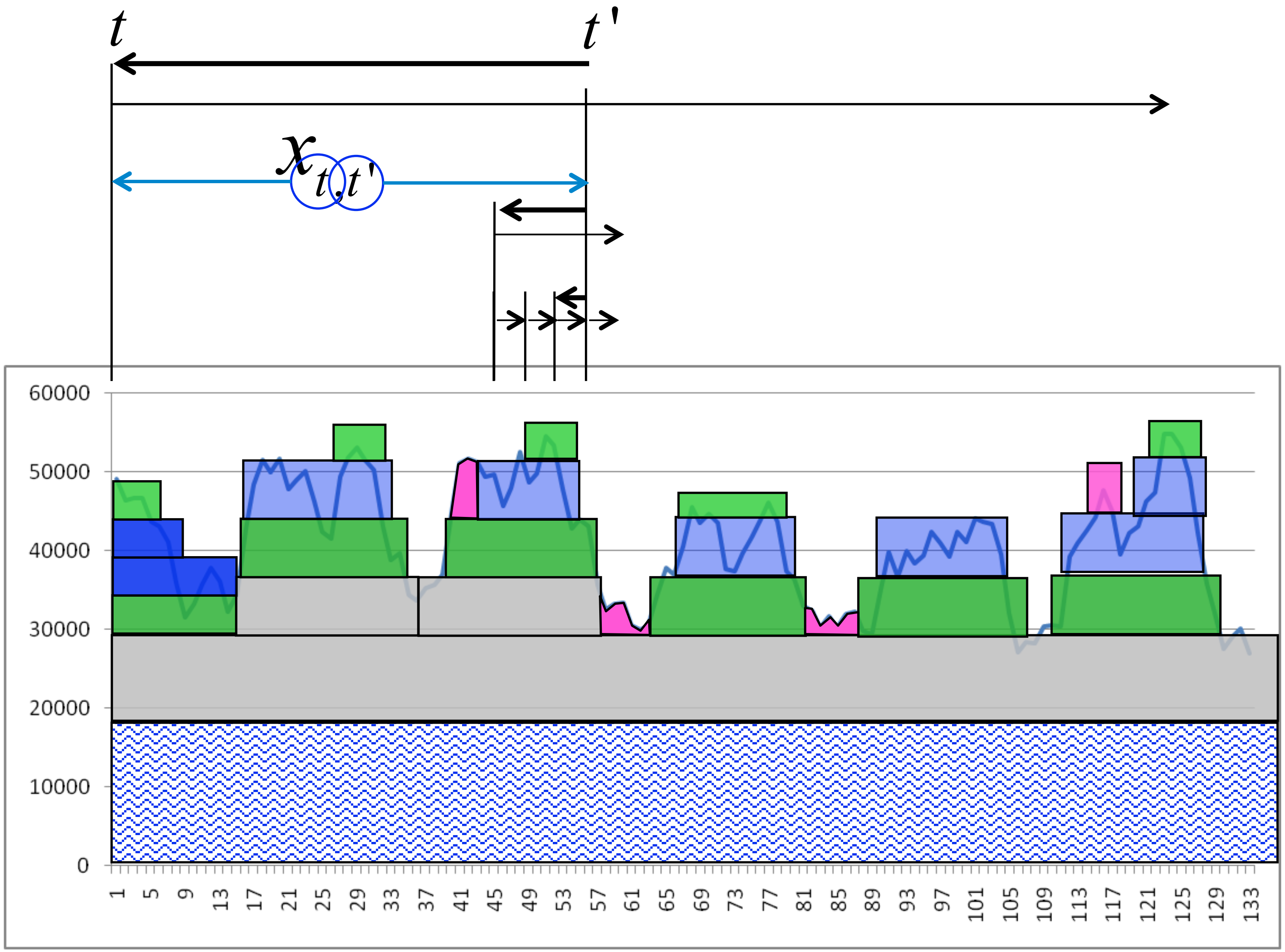
The time-staging of decisions



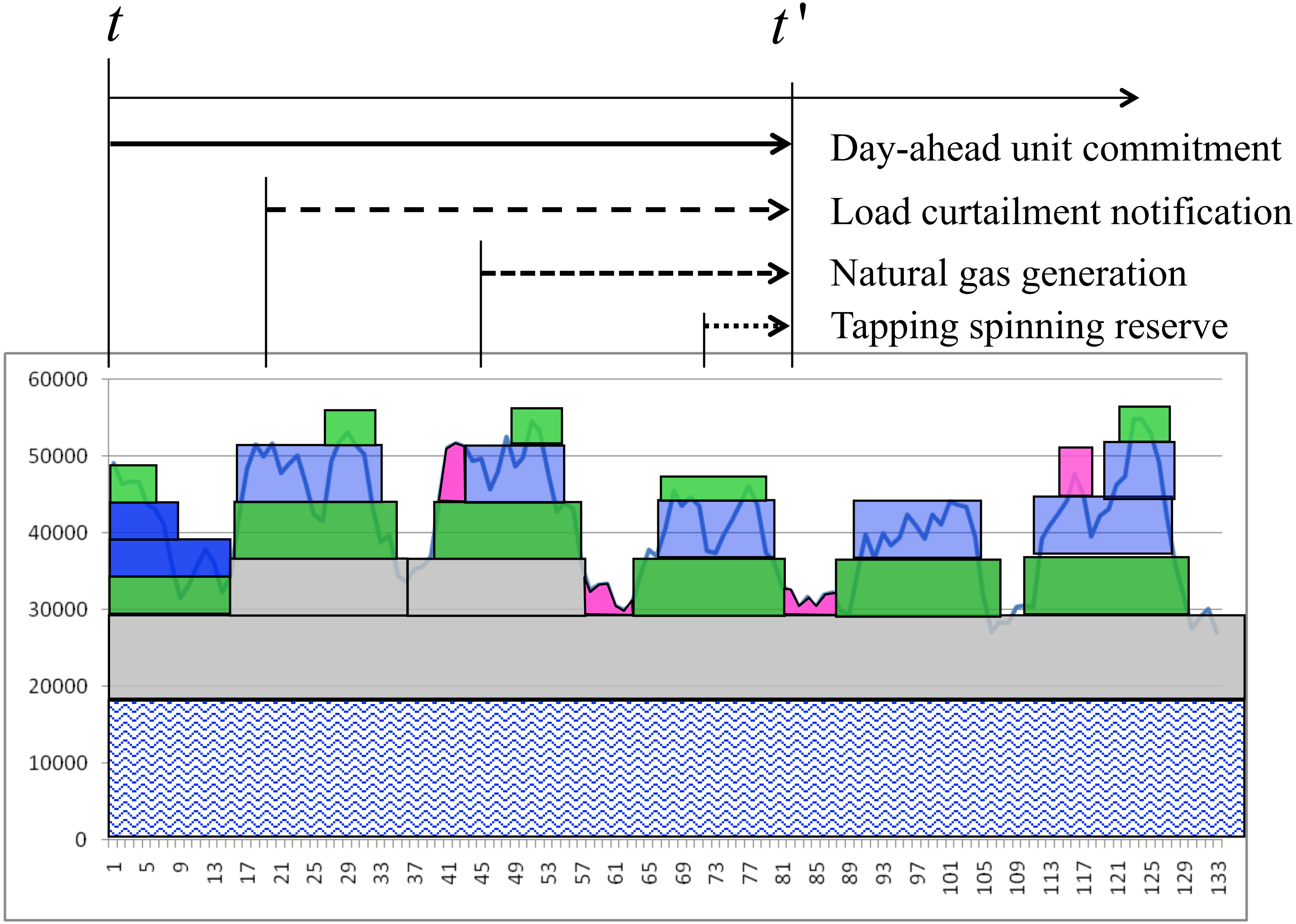
The time-staging of decisions



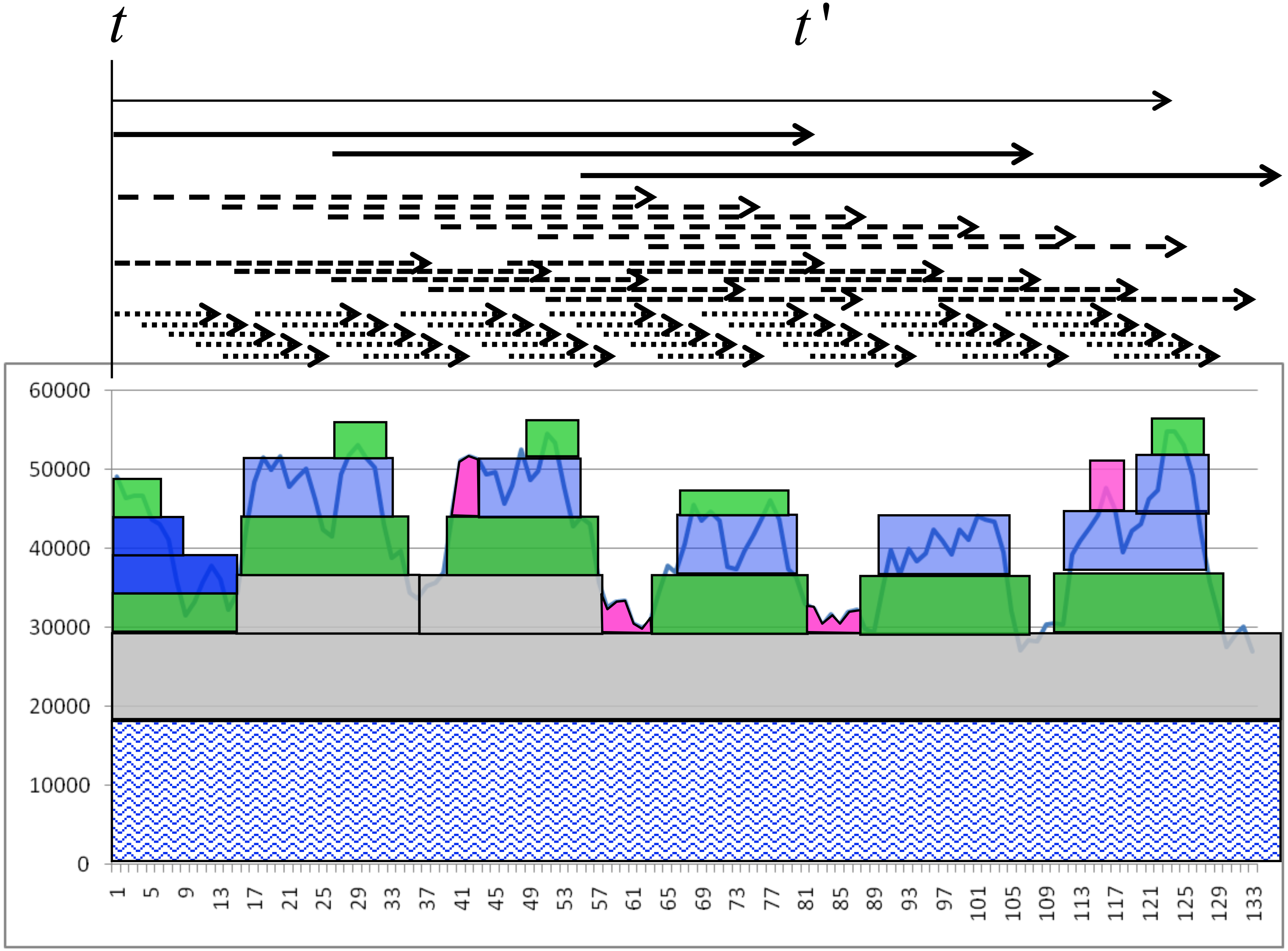
The time-staging of decisions



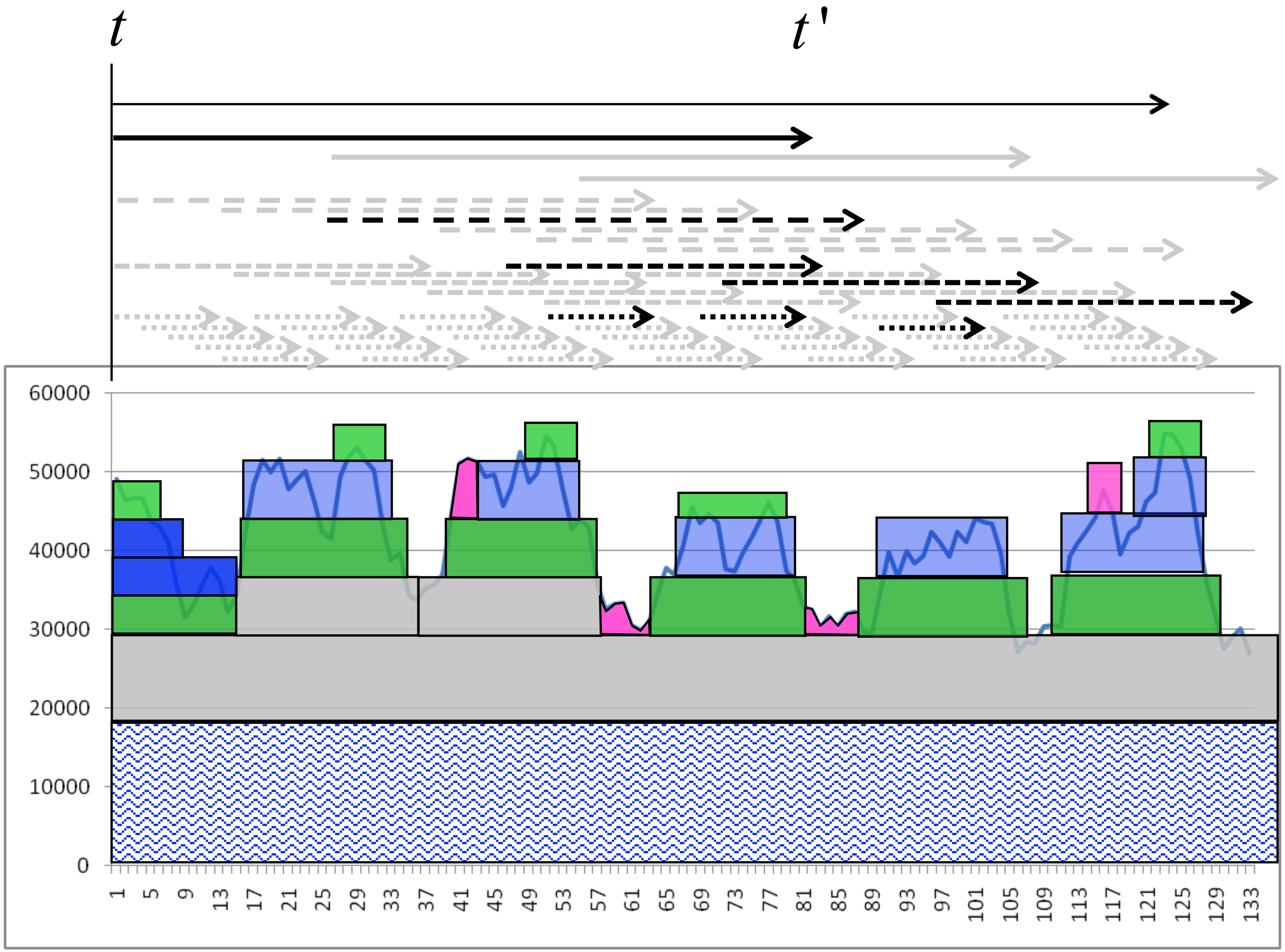
The time-staging of decisions



The time-staging of decisions



The time-staging of decisions



The stochastic unit commitment problem

- A deterministic model
  - » Optimize over all decisions at the same time

$$\min_{\substack{(x_{t'})_{t'=1,\ldots,24} \\ (y_{t'})_{t'=1,\ldots,24}}} \sum_{t'=1}^{24} C(x_{t'}, y_{t'})$$

Steam generation

Gas turbines

  - » These decisions need to made with different horizons
    - Steam generation is made day-ahead
    - Gas turbines can be planned an hour ahead or less

The stochastic unit commitment problem

- A stochastic model
  - » The decision problem at time *t:*

$$x_{t,t'} \qquad y_{t',t'}$$

$$\min_{\substack{(x_{t,t'})_{t'=1,\ldots,24} \\ (y_{t',t'})_{t'=1,\ldots,24}}} \mathrm{E} \sum_{t'=1}^{24} C(x_{t,t'}, y_{t',t'})$$

- $x_{t,t'}$ is determined at time *t*, to be implemented at time *t'*
- $y_{t',t'}$ is determined at time *t'*, to be implemented at time *t'*

  - » Important to recognize information content
    - At time *t*, $x_{t,t'}$ is deterministic.
    - At time *t*, $y_{t',t'}$ is stochastic.

The stochastic unit commitment problem

- A stochastic model
  - » We capture the information content of decisions

$$\min_{\substack{(x_{t,t'})_{t'=1,\ldots,24} \\ \pi}} \mathrm{E} \sum_{t'=1}^{24} C(x_{t,t'}, Y^{\pi}(S_{t'}))$$

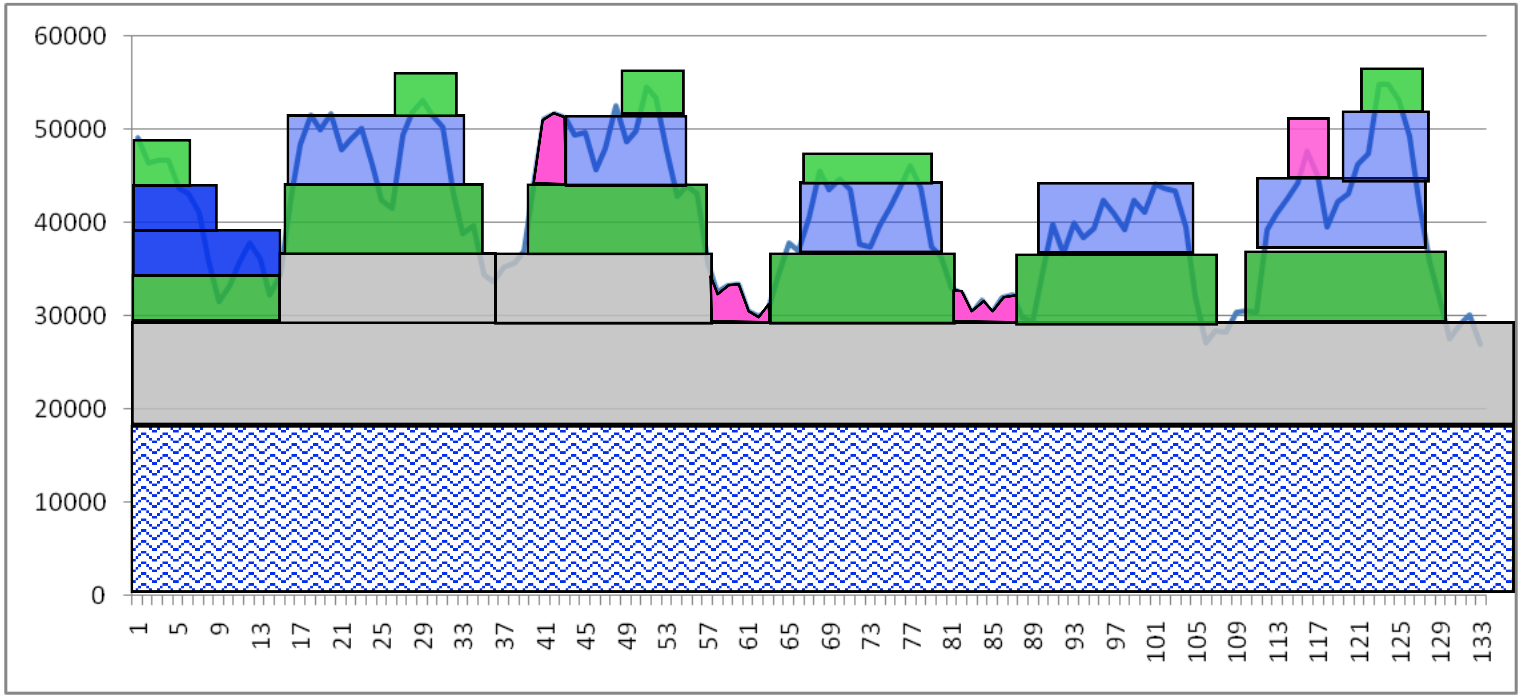
Policy

- $x_{t,t'}$ is determined at time *t*, to be implemented at time *t'*
- $y_{t',t'}$ is determined at time *t'* by the policy $Y^{\pi}(S_{t'})$

  - » The policy $Y^{\pi}(S_{t'})$ is constrained by the solution x_t which is influenced by two parameters:
    - *p* is the fraction of power allocated for spinning reserve
    - *q* is the fraction of the wind that we plan on using.

The stochastic unit commitment problem

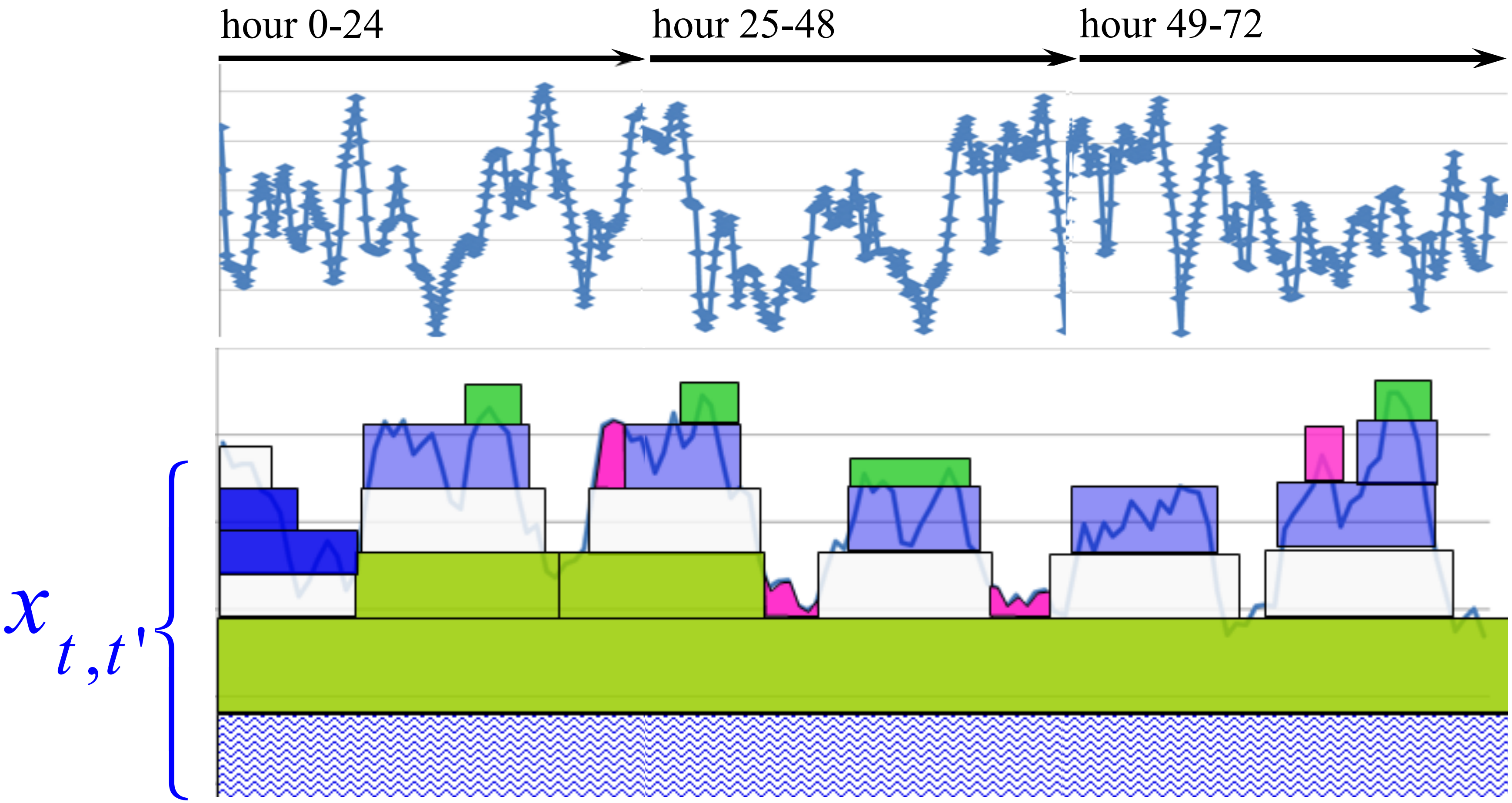
- Matching supply to demand
  - » We have to find the best way to meet demand



  - » …. Now we have to do it in the presence of significant levels of wind and solar energy.

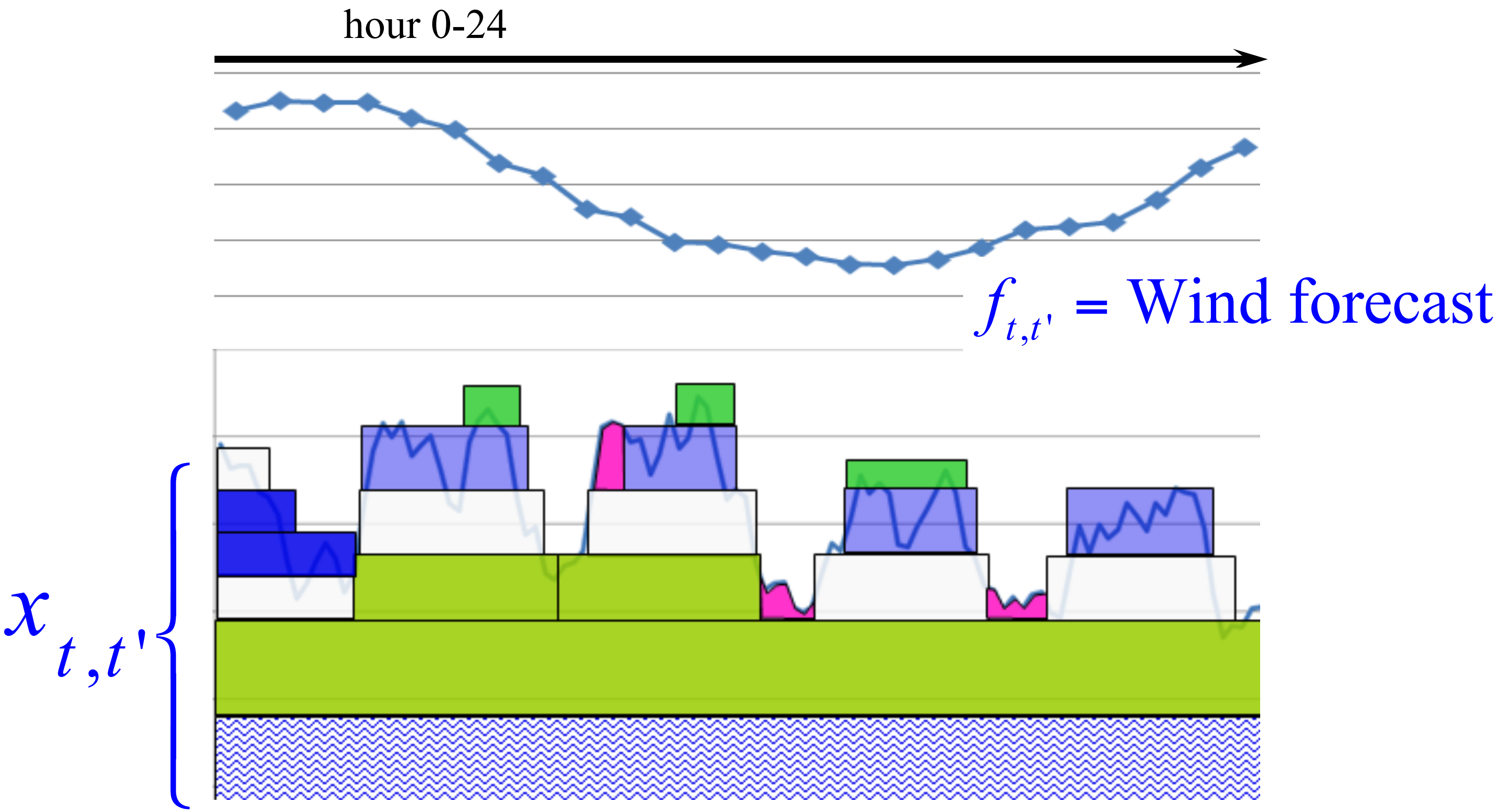
The stochastic unit commitment problem

- The unit commitment problem
  - » Rolling forward with perfect forecast of actual wind, demand, …



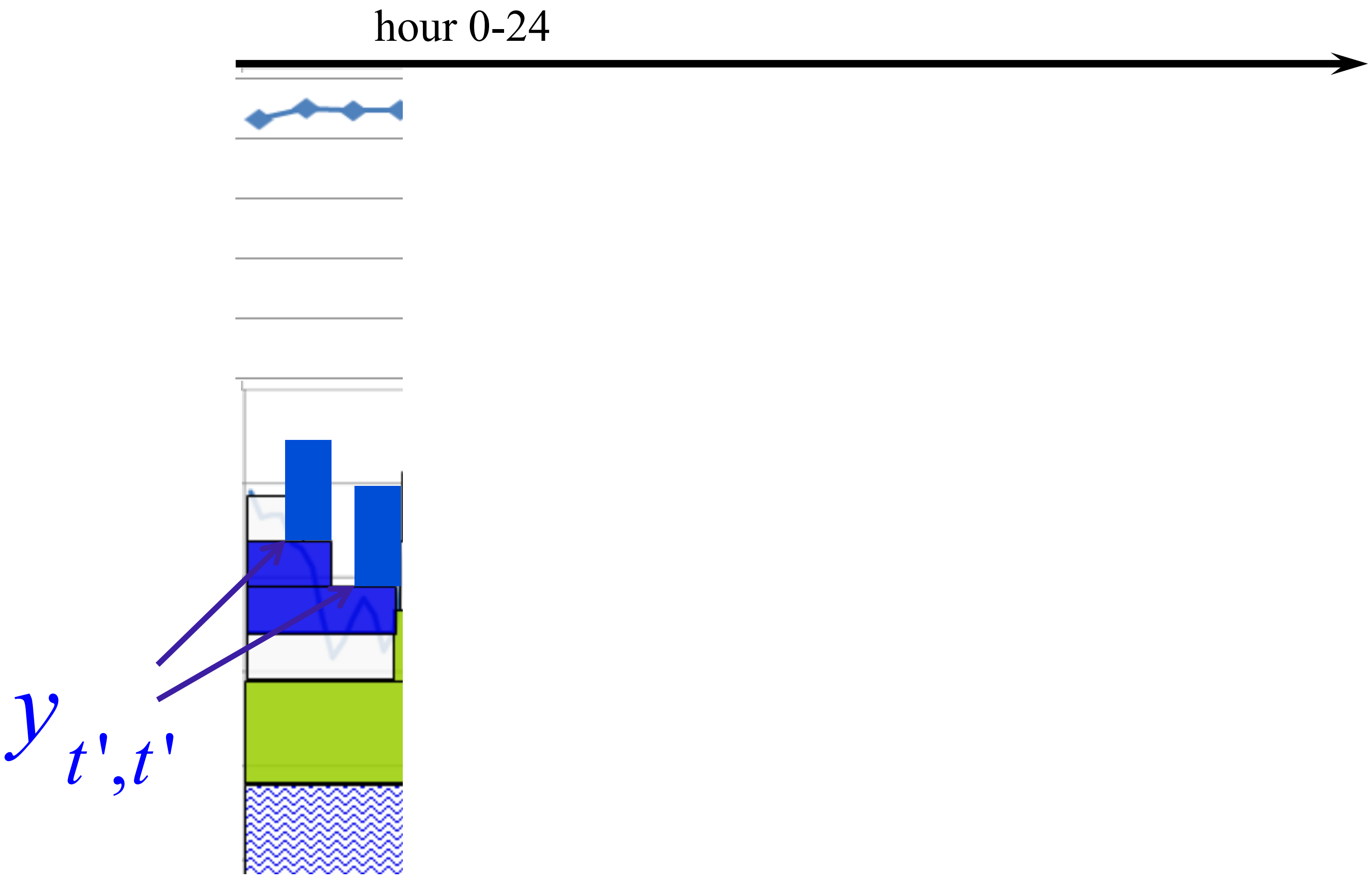
The stochastic unit commitment problem

- When planning, we have to use a *forecast* of energy from wind, then live with what *actually* happens.



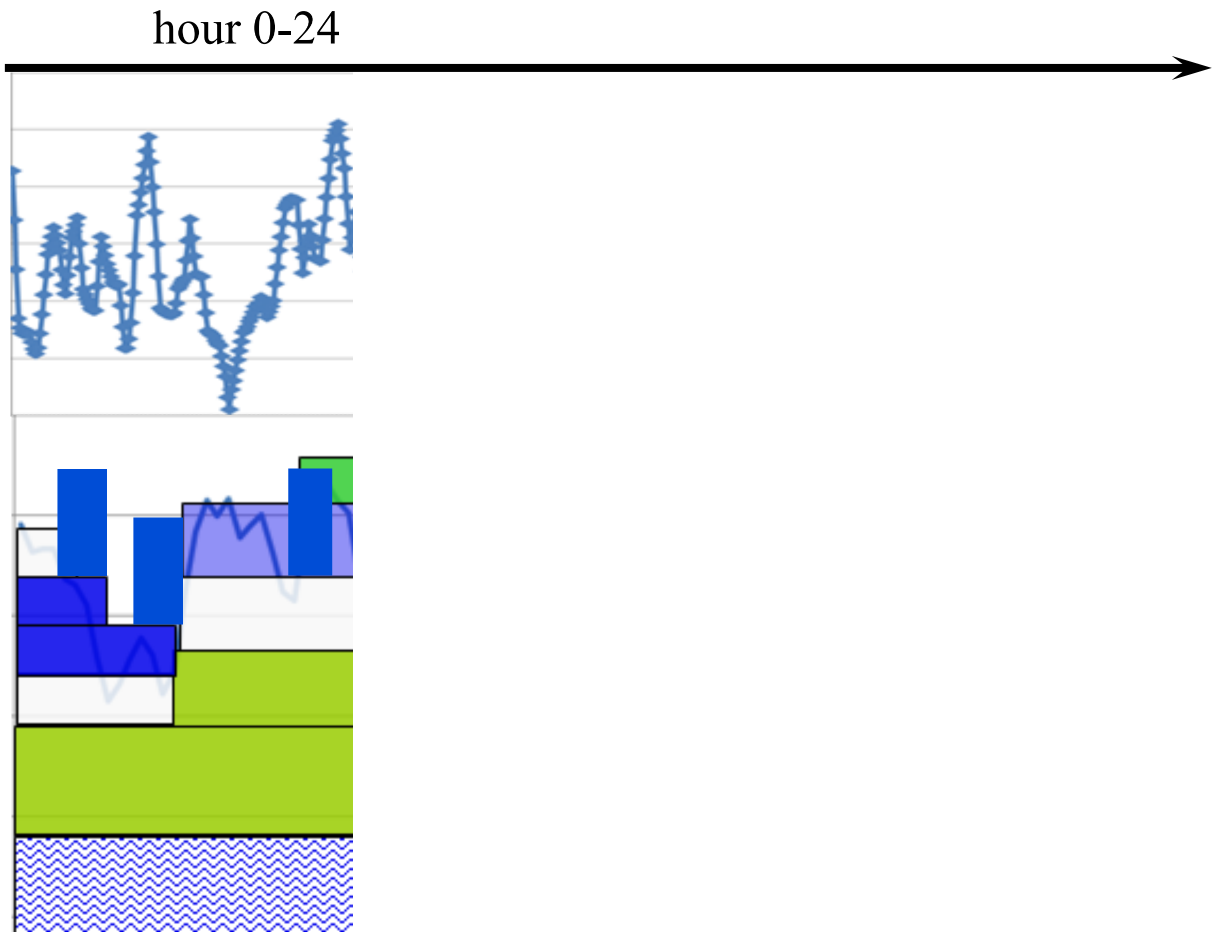
The stochastic unit commitment problem

- The unit commitment problem
  - » Stepping forward observing actual wind, making small adjustments



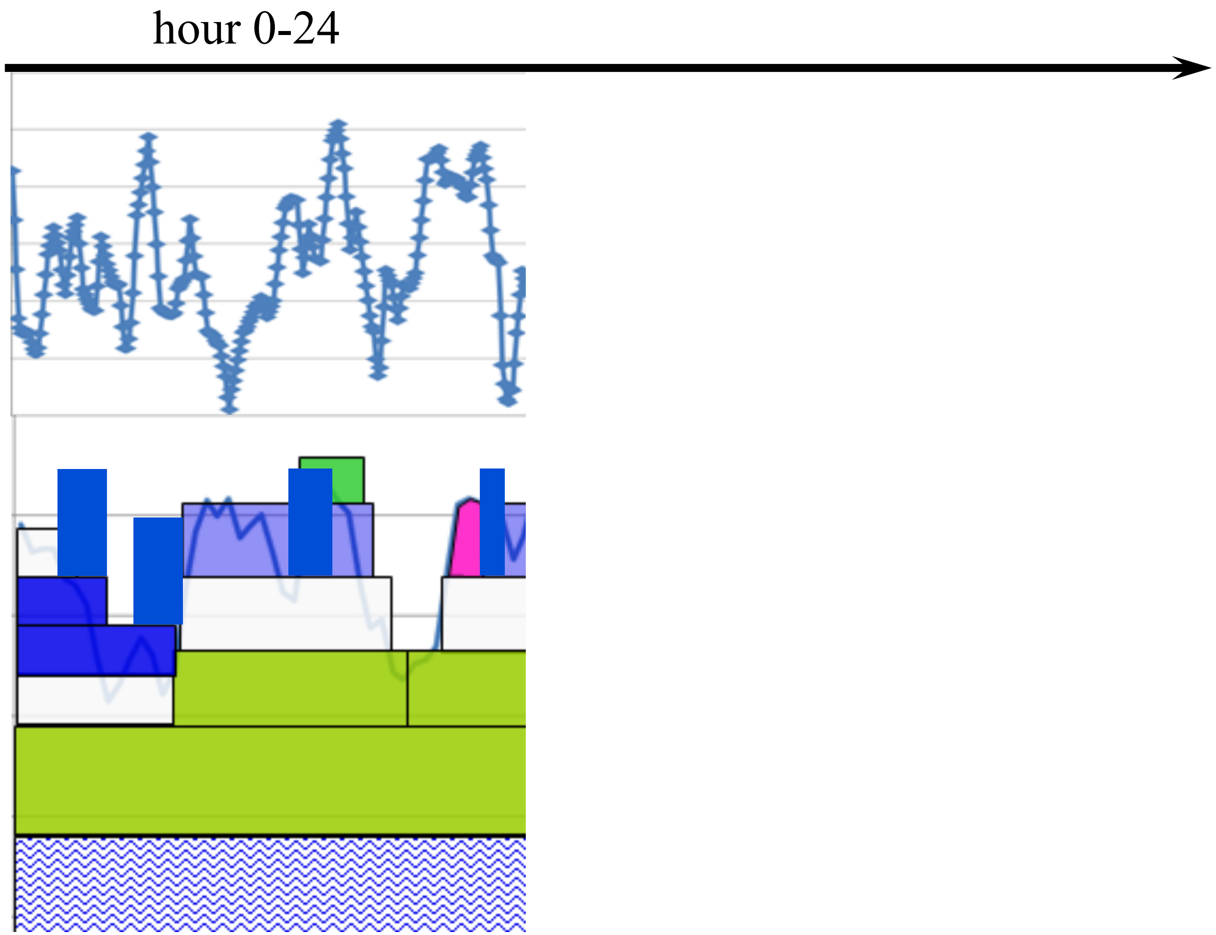
The stochastic unit commitment problem

- The unit commitment problem
  - » Stepping forward observing actual wind, making small adjustments



The stochastic unit commitment problem

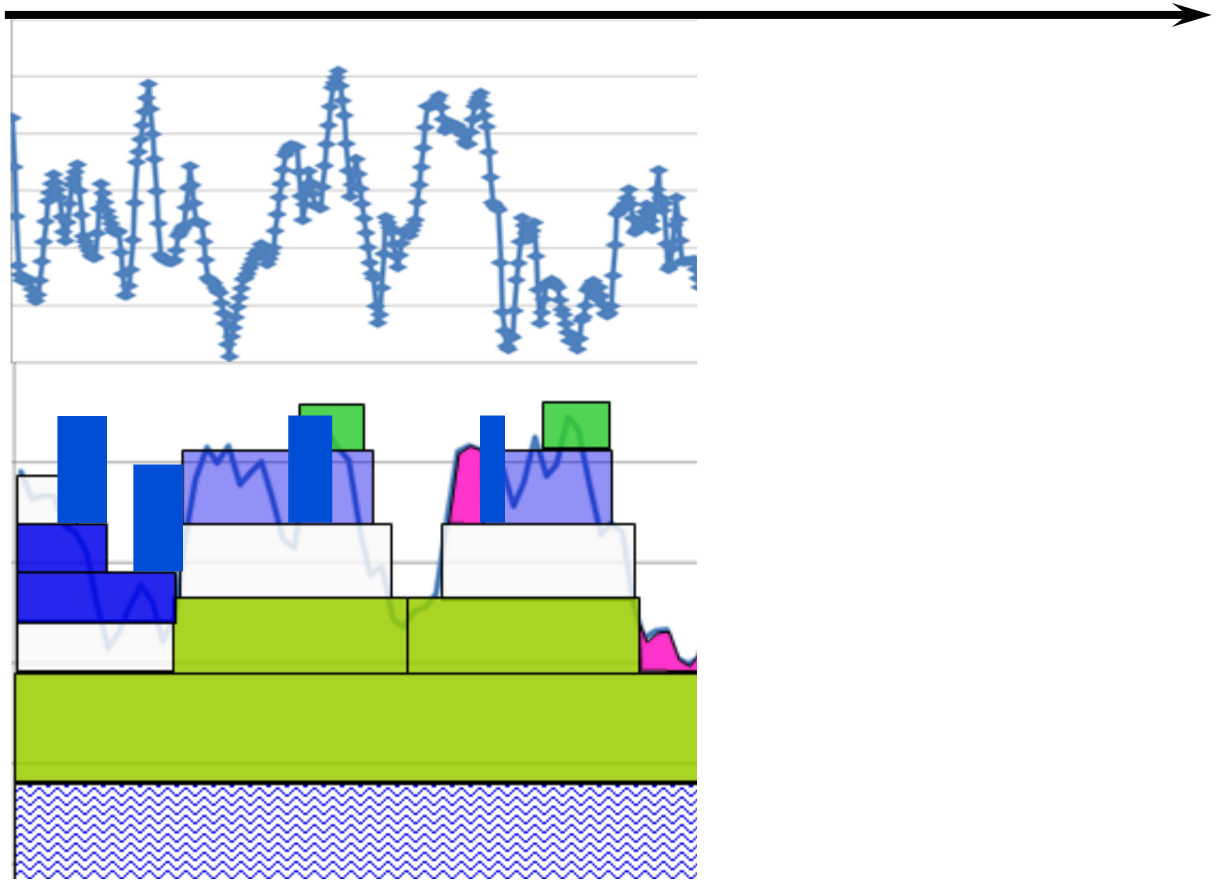
- The unit commitment problem
  - » Stepping forward observing actual wind, making small adjustments



The stochastic unit commitment problem

- The unit commitment problem
  - » Stepping forward observing actual wind, making small adjustments

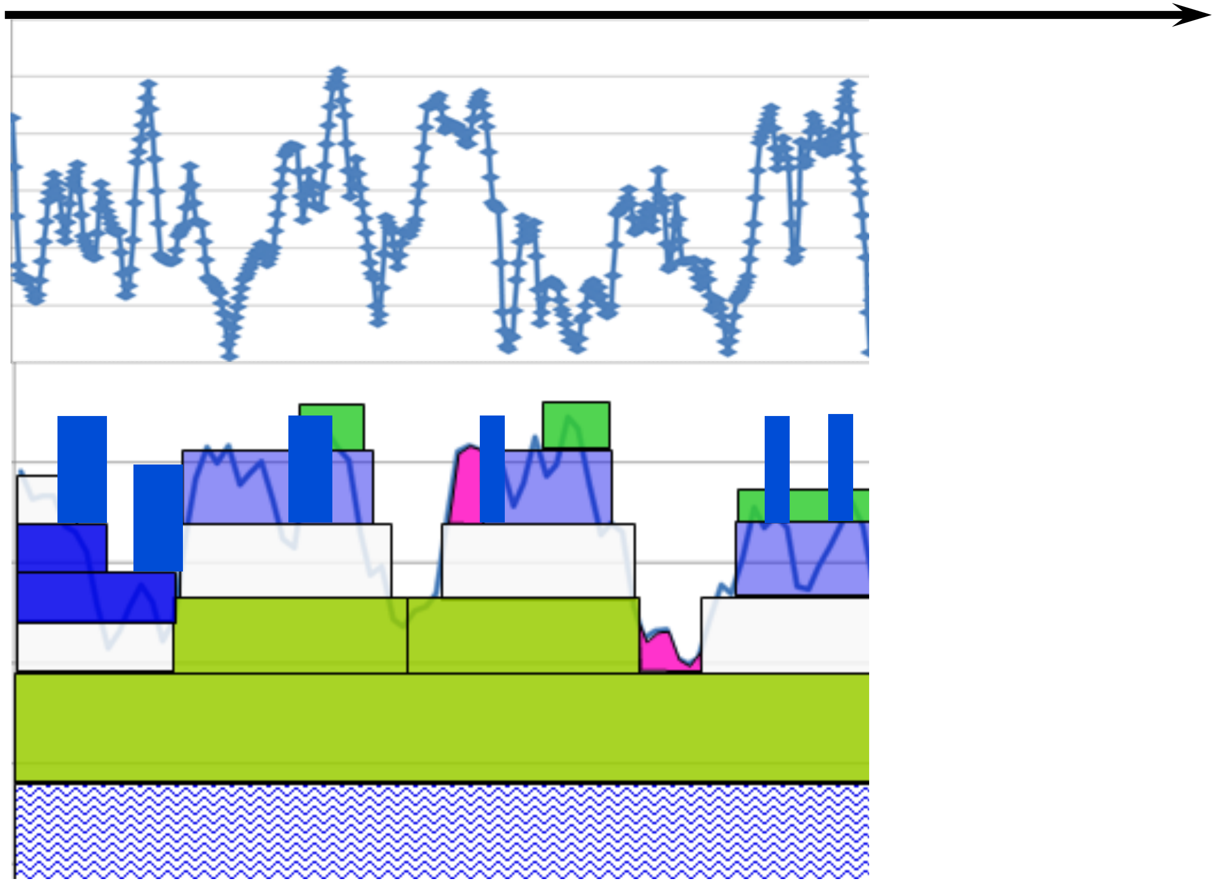
hour 0-24



The stochastic unit commitment problem

- The unit commitment problem
  - » Stepping forward observing actual wind, making small adjustments

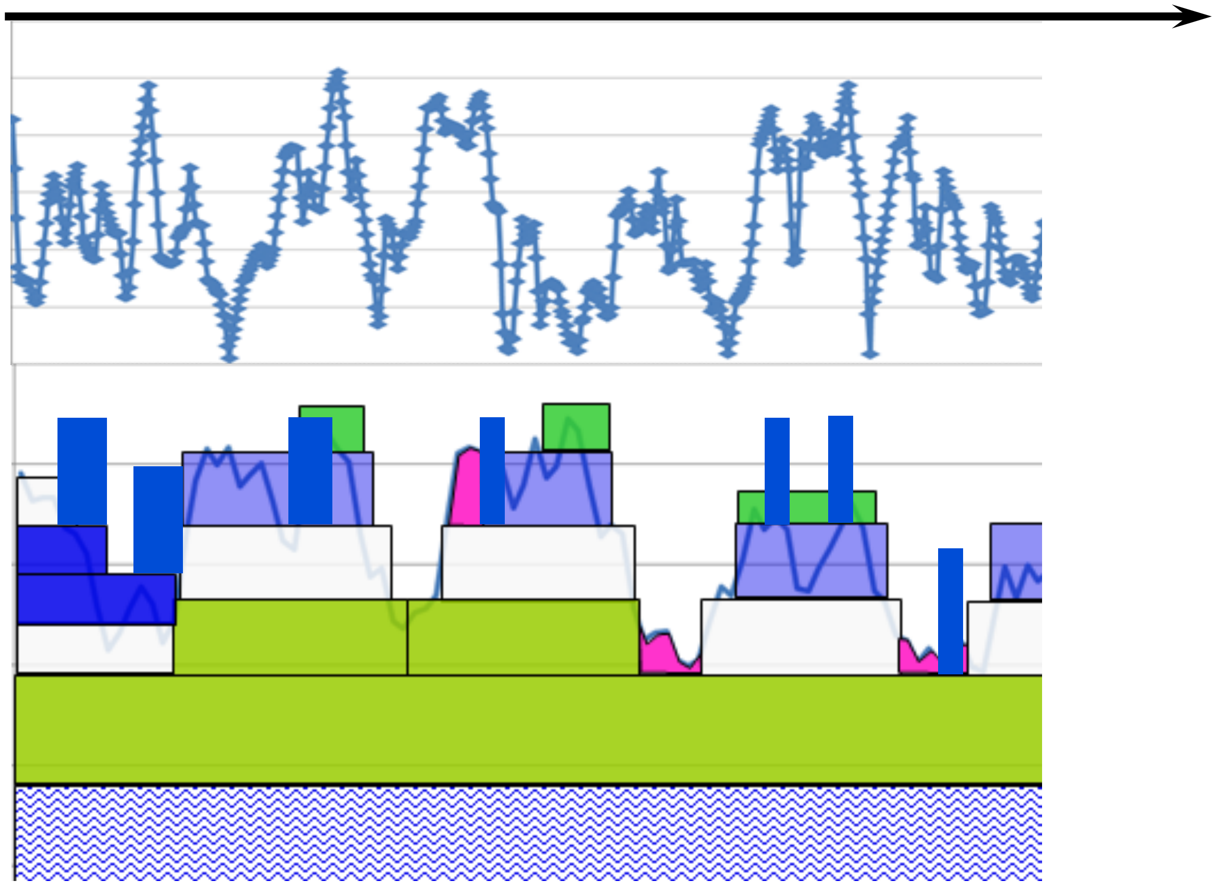
hour 0-24



The stochastic unit commitment problem

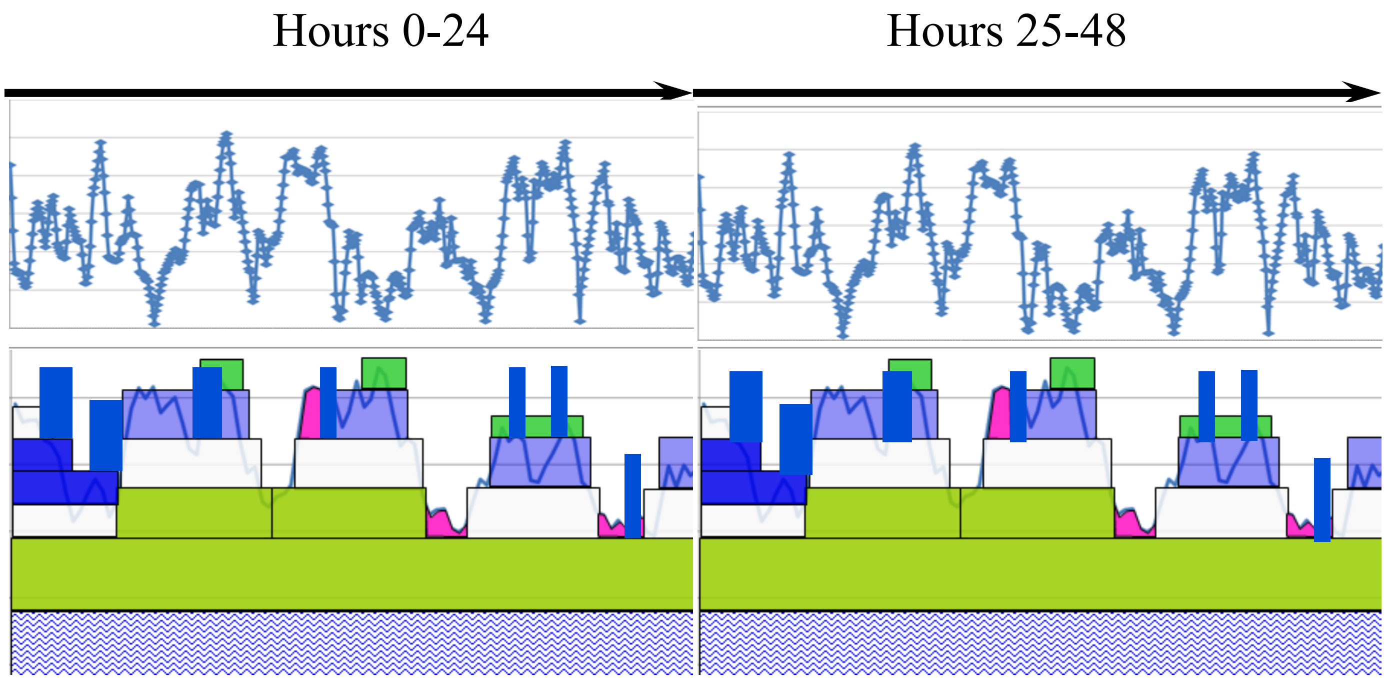
- The unit commitment problem
  - » Stepping forward observing actual wind, making small adjustments

hour 0-24

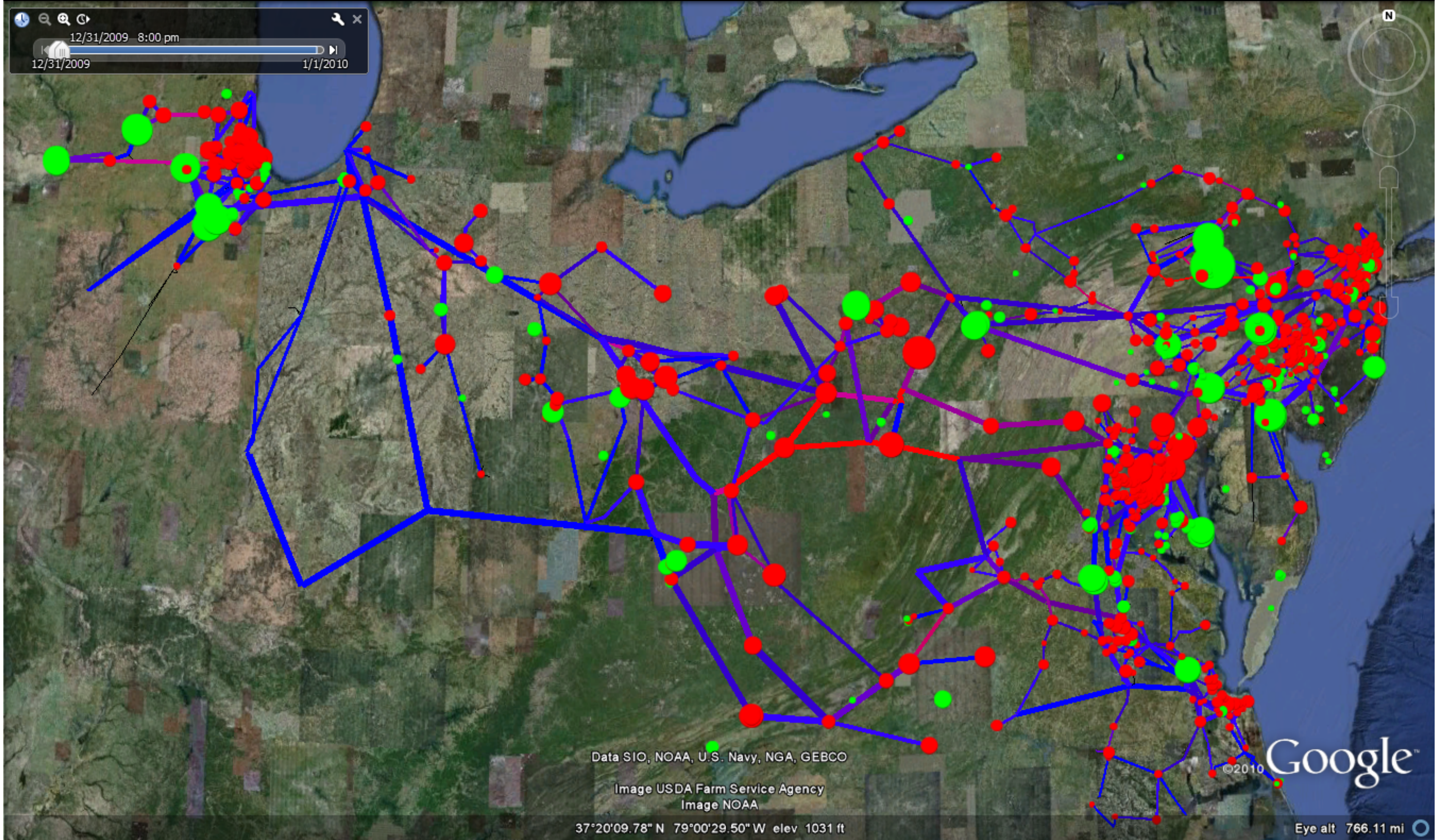


The stochastic unit commitment problem

- The unit commitment problem
  - » Stepping forward observing actual wind, making small adjustments



220 kv grid, 40 percent wind



The value of wind

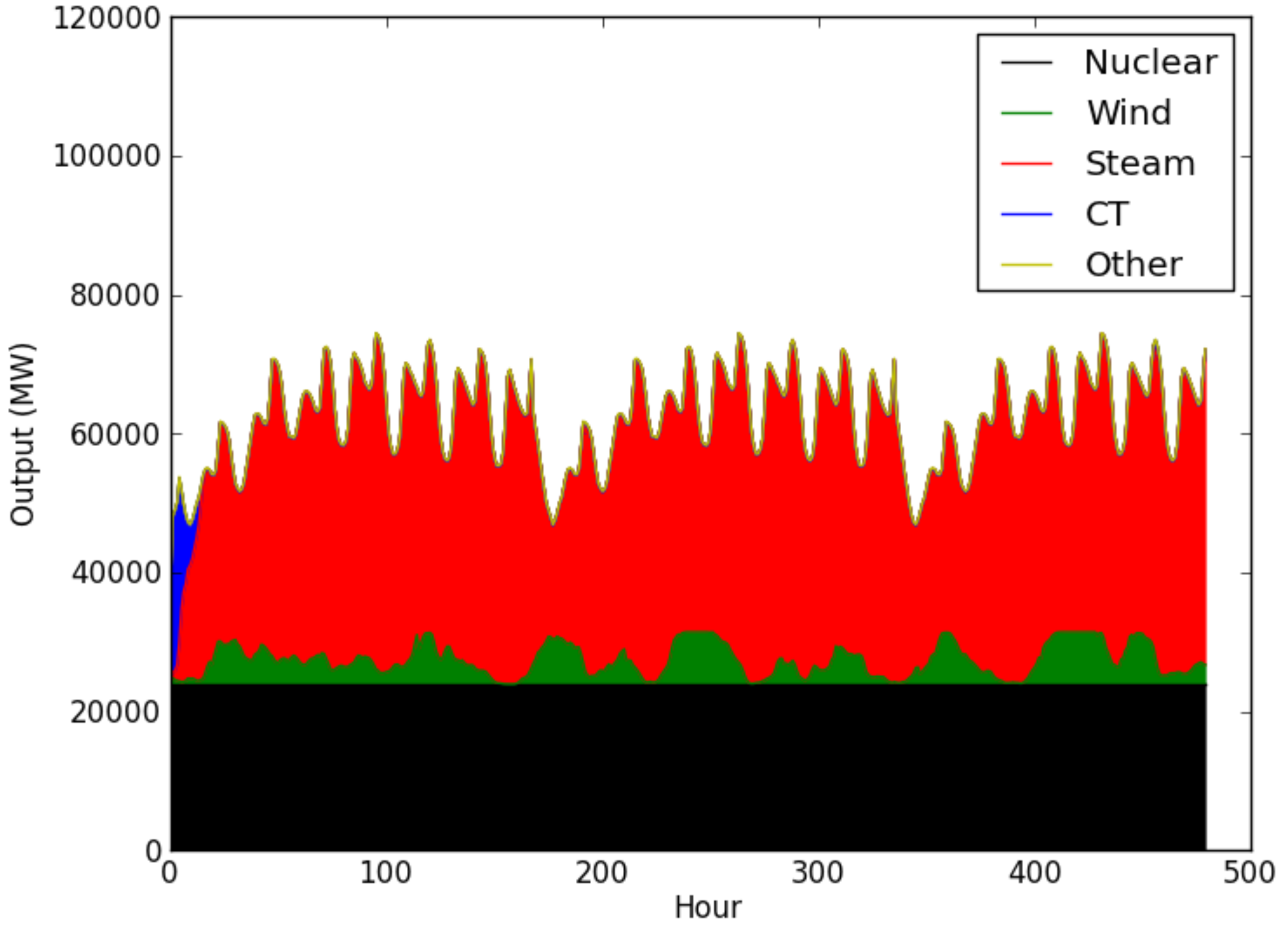
- Research question: What is the relative cost of uncertainty vs variability?

  » Scenario 1: Deterministic wind – Wind is variable, but we can forecast it perfectly

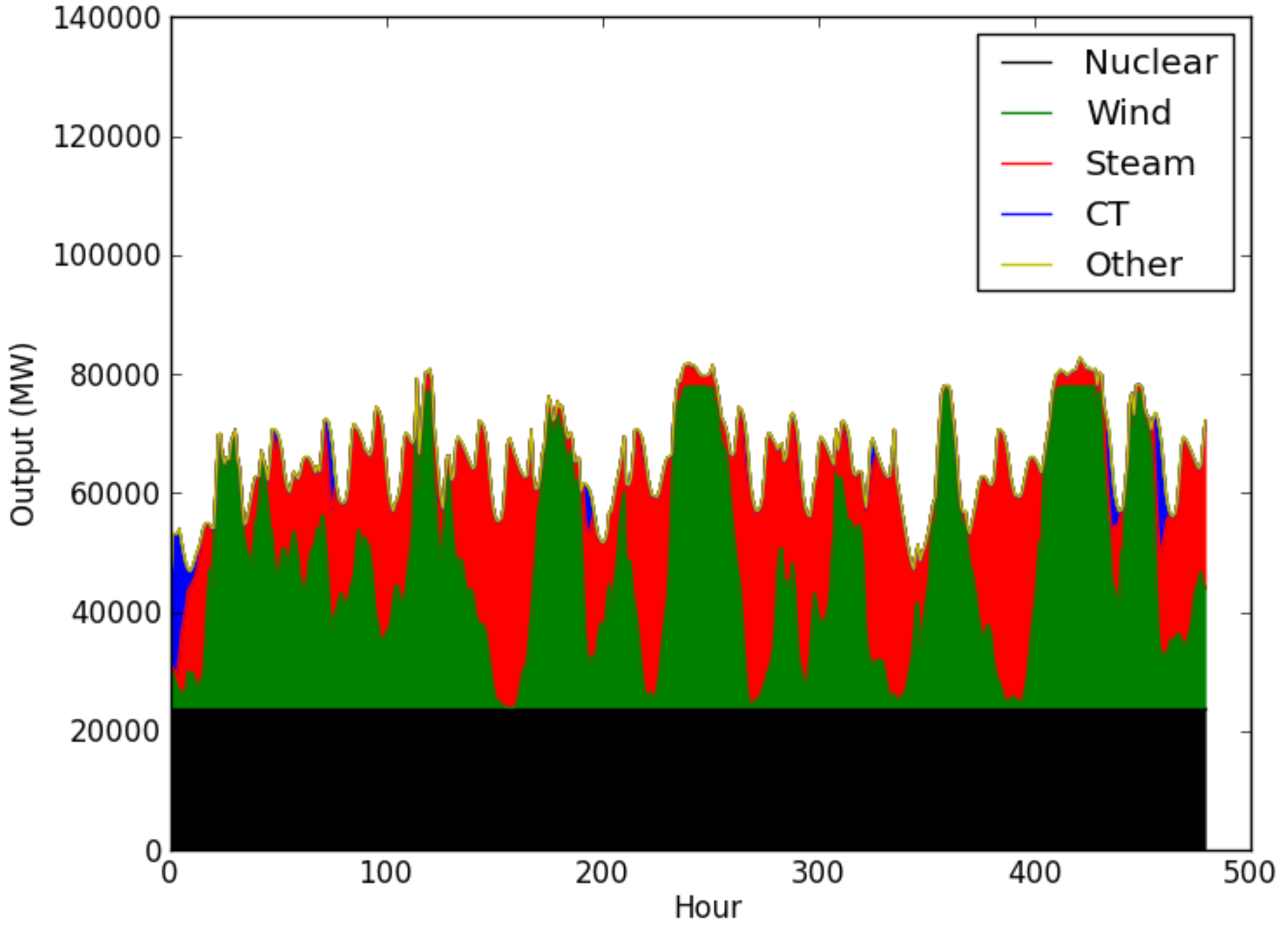
  » Scenario 2: Stochastic wind – We forecast wind, but the actual does not match the forecast

  » Scenario 3: Constant wind – Wind generates energy at a constant (and perfectly known) value

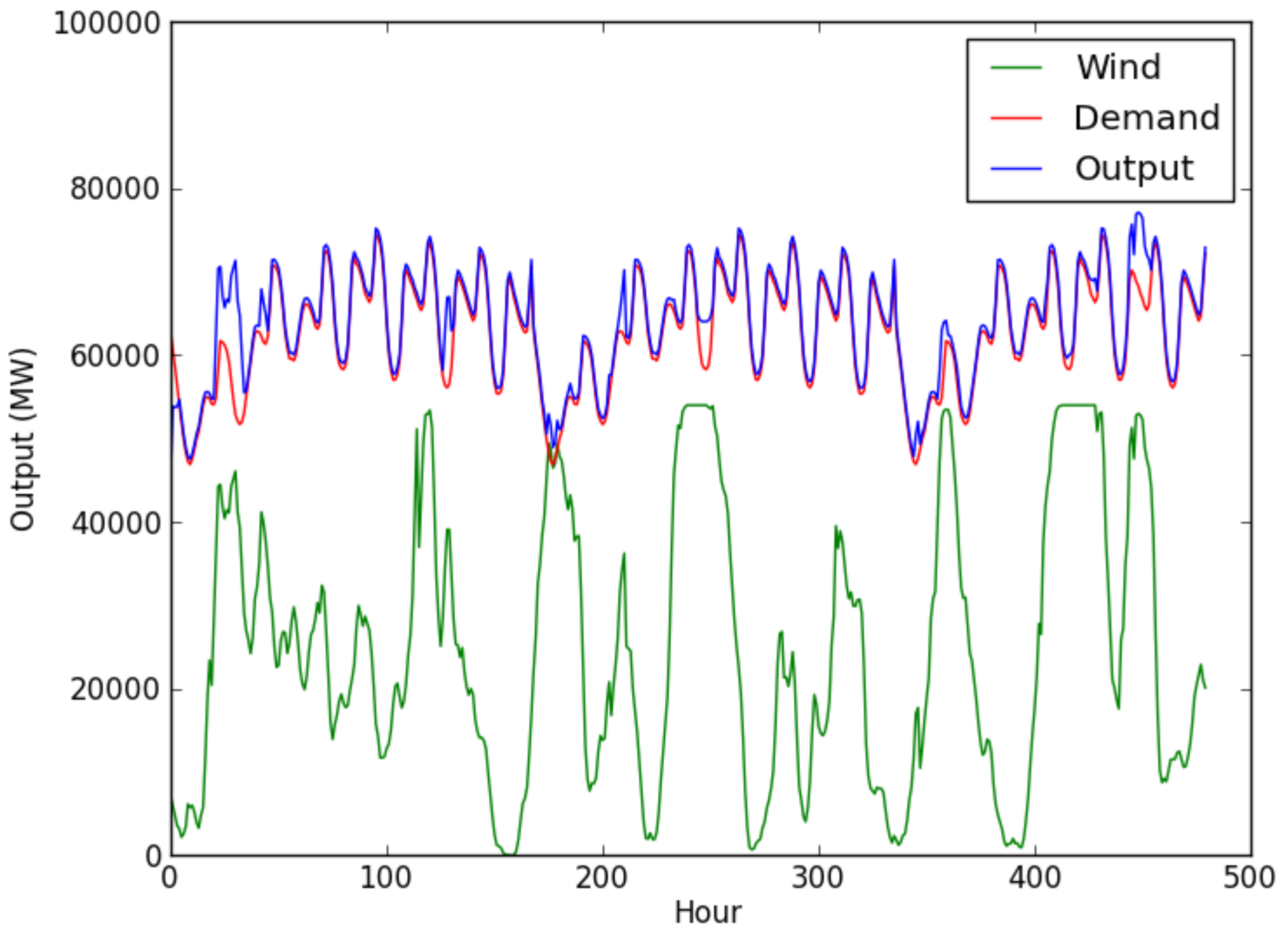
Perfect forecast – 5 percent wind



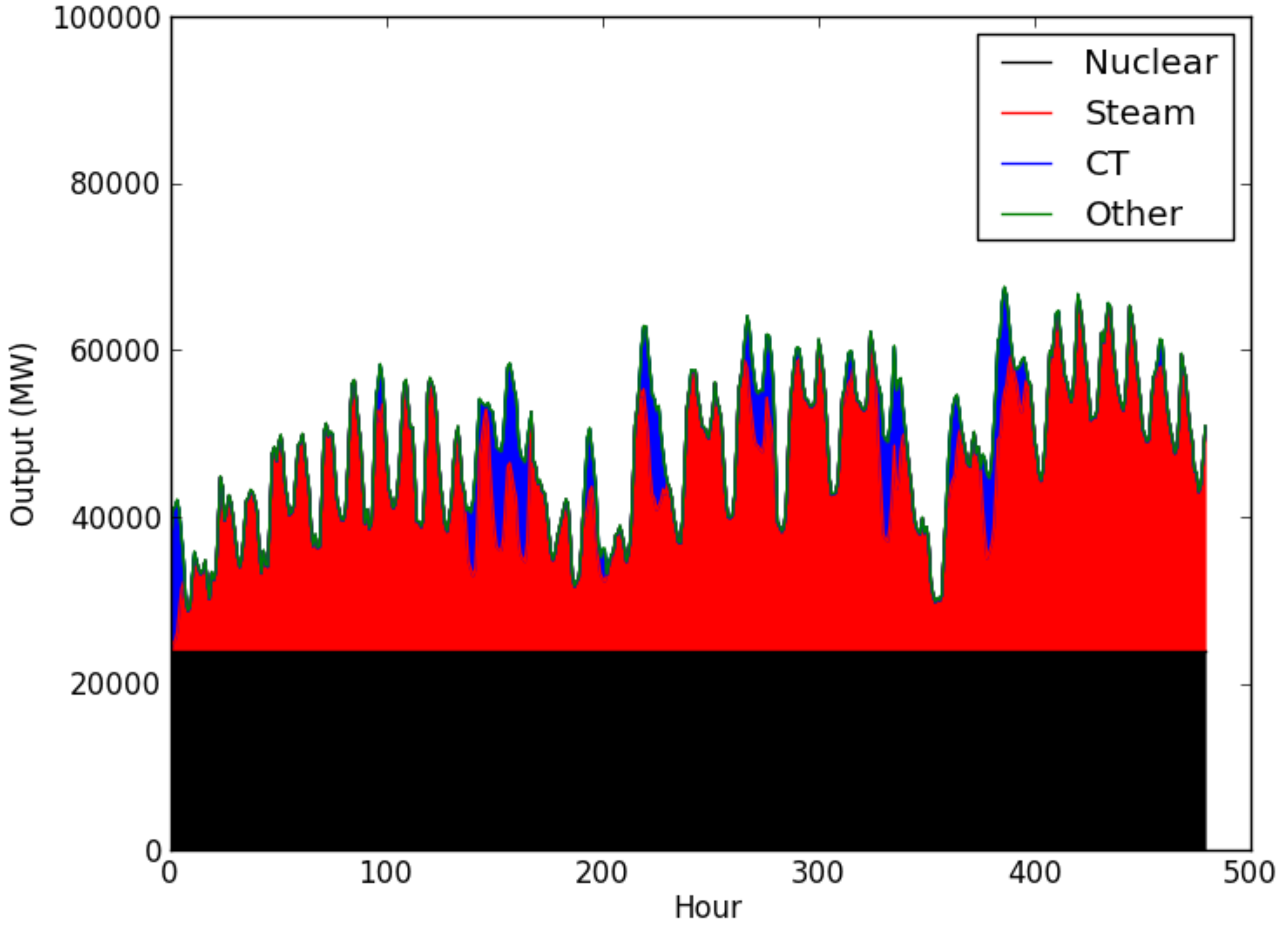
Perfect forecast – 40 percent wind



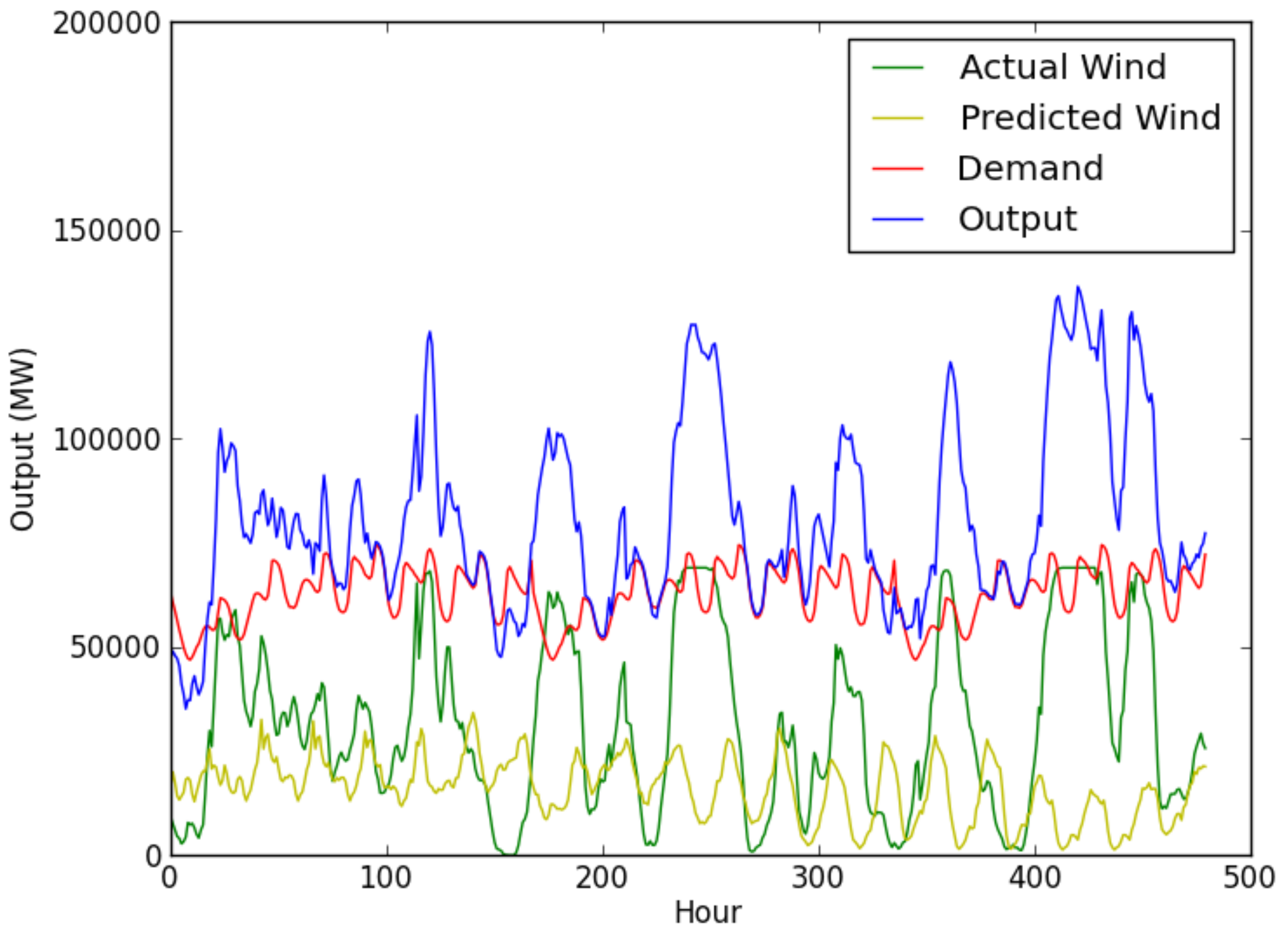
Perfect forecast – 40 percent wind



Imperfect forecast – 40 percent wind

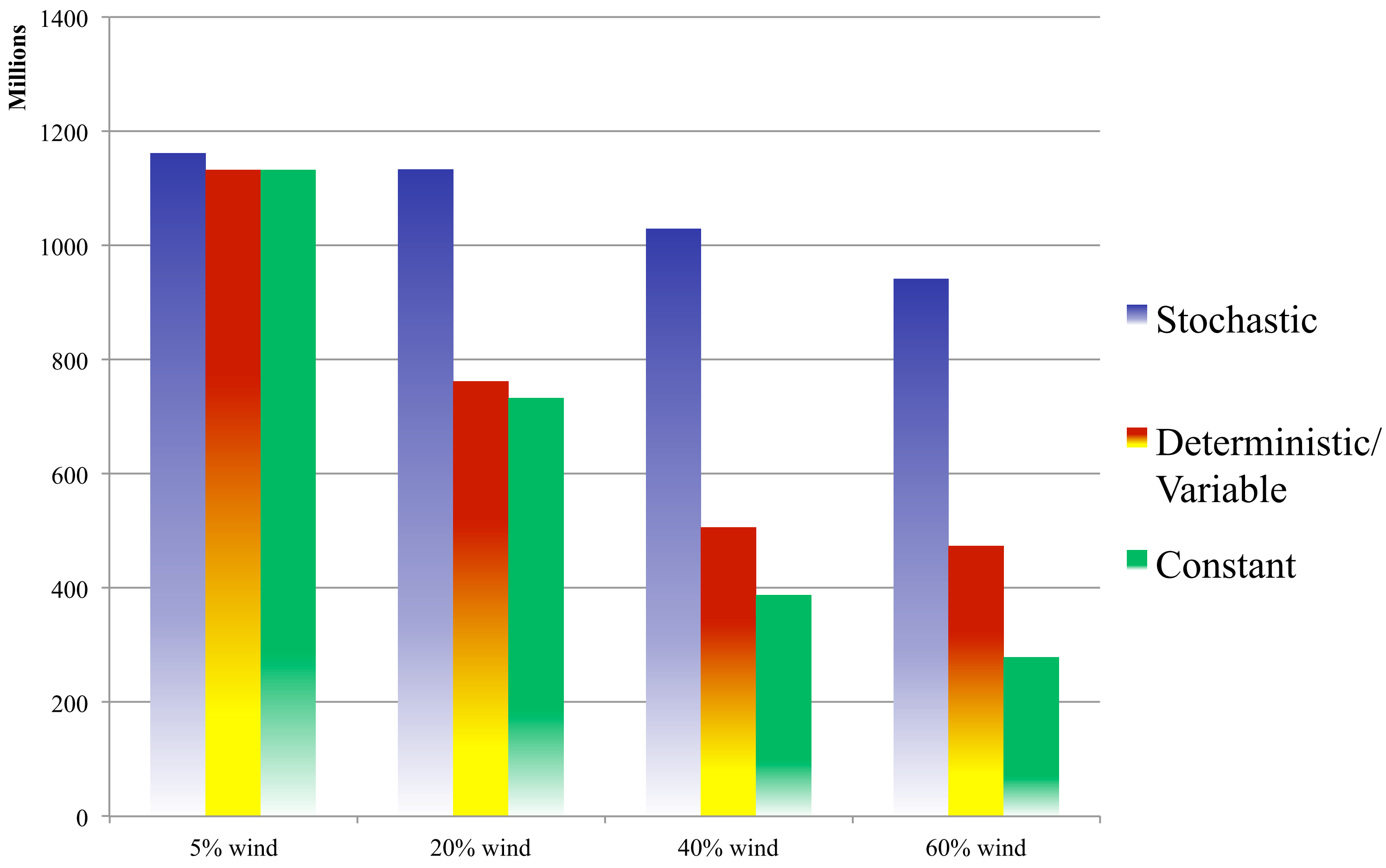


Imperfect forecast – 40 percent wind



The stochastic unit commitment problem

- The effect of modeling uncertainty in wind



Lecture outline

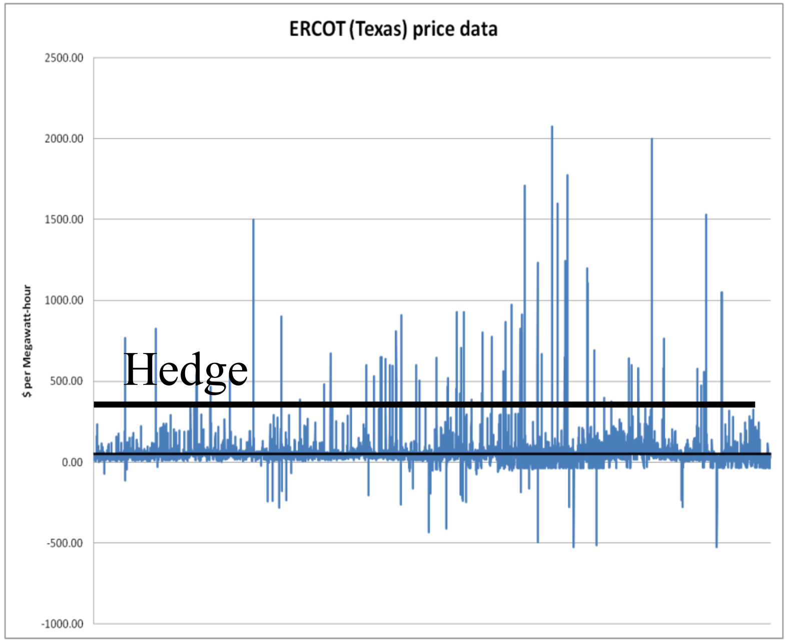
- How do we achieve robust behaviors?
  - You do not just blindly solve a stochastic optimization – you have to understand the nature of uncertainty, and identify specific, implementable strategies for dealing with uncertainty.

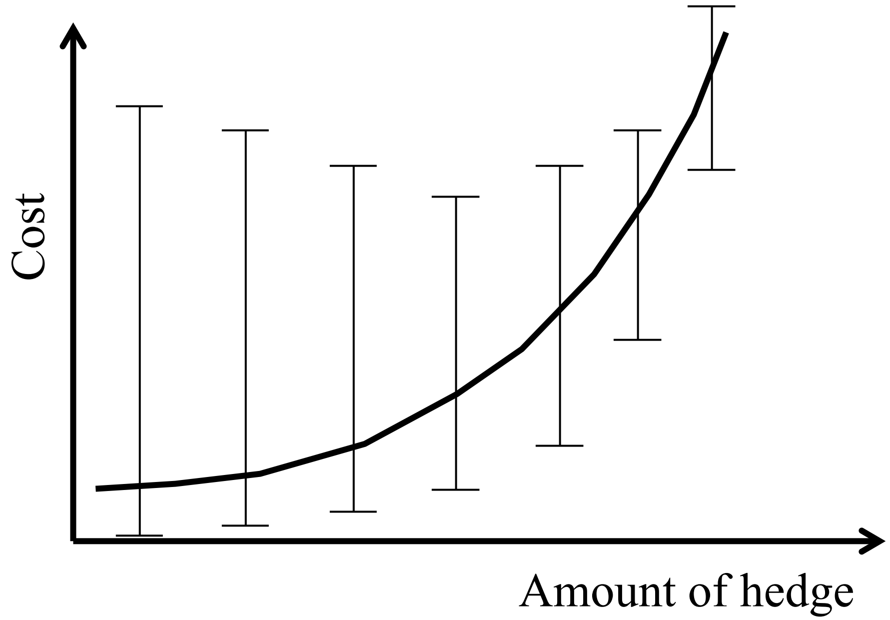
Working with uncertainty

- Working with uncertainty is not magic.
  - » You have to identify the types of the uncertainties you are working with…
  - » … quantify the risks and rewards, so you know what you are trying to achieve….
  - » …. and then identify the types of strategies that are best suited to deal with the uncertainties you are facing.

Hedging risk

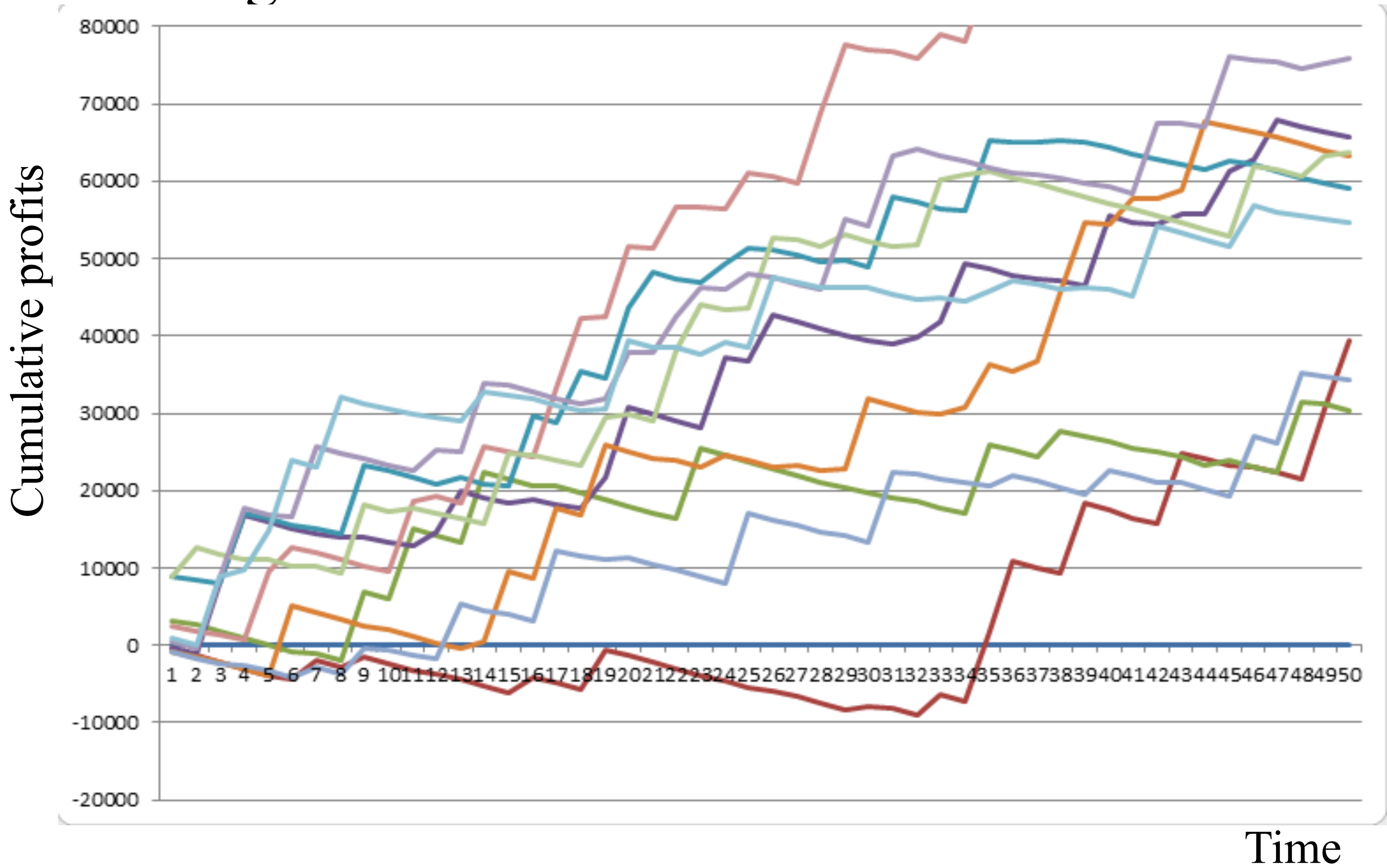
- Hedging electricity prices
  - » We can sign a contract to deliver electricity purchased on the spot market, exposing us to spikes.
  - » We can protect ourselves by purchasing hedge contracts. This reduces risk, but reduces profits.





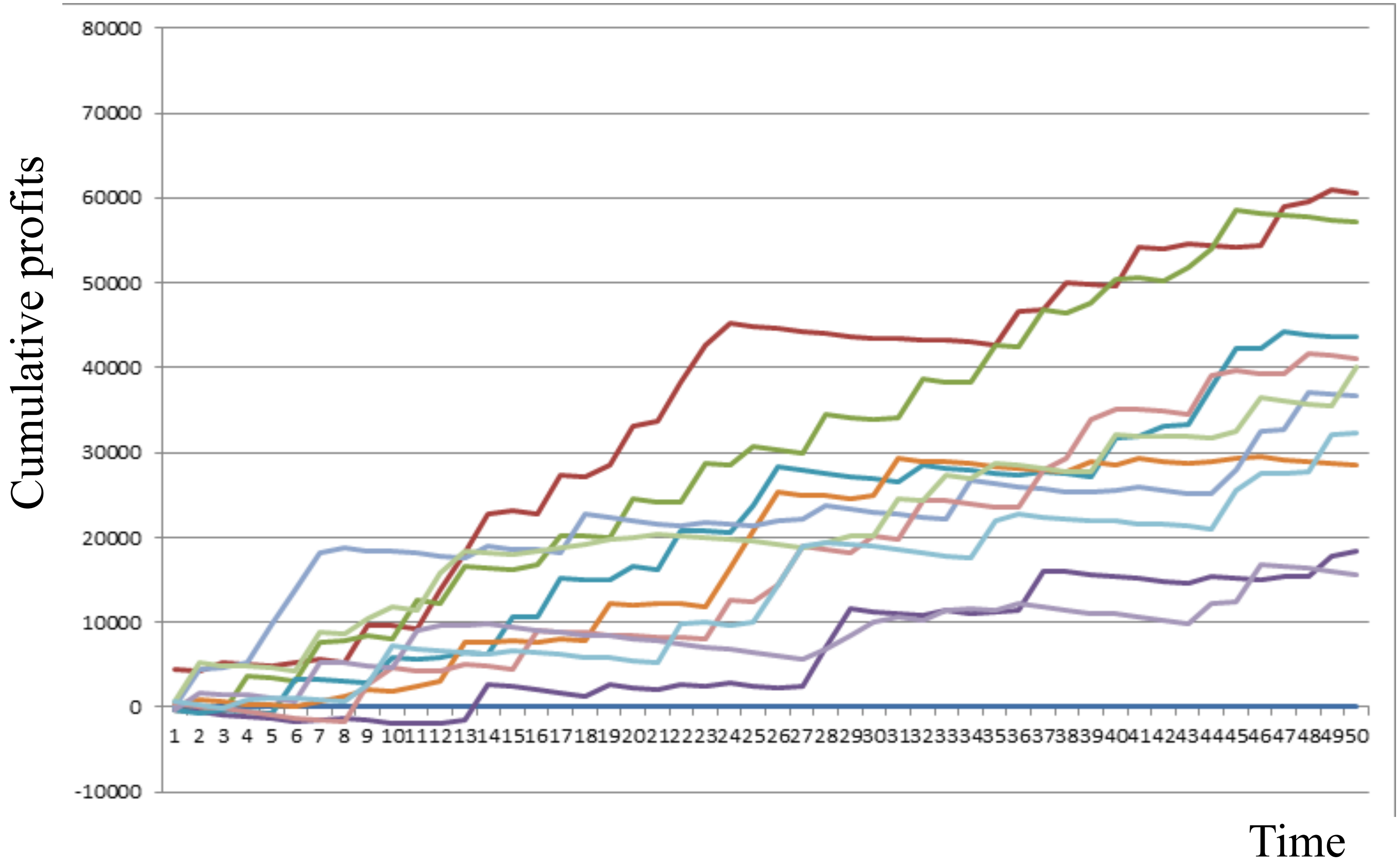
Hedging risk

- Hedge 25 MWh



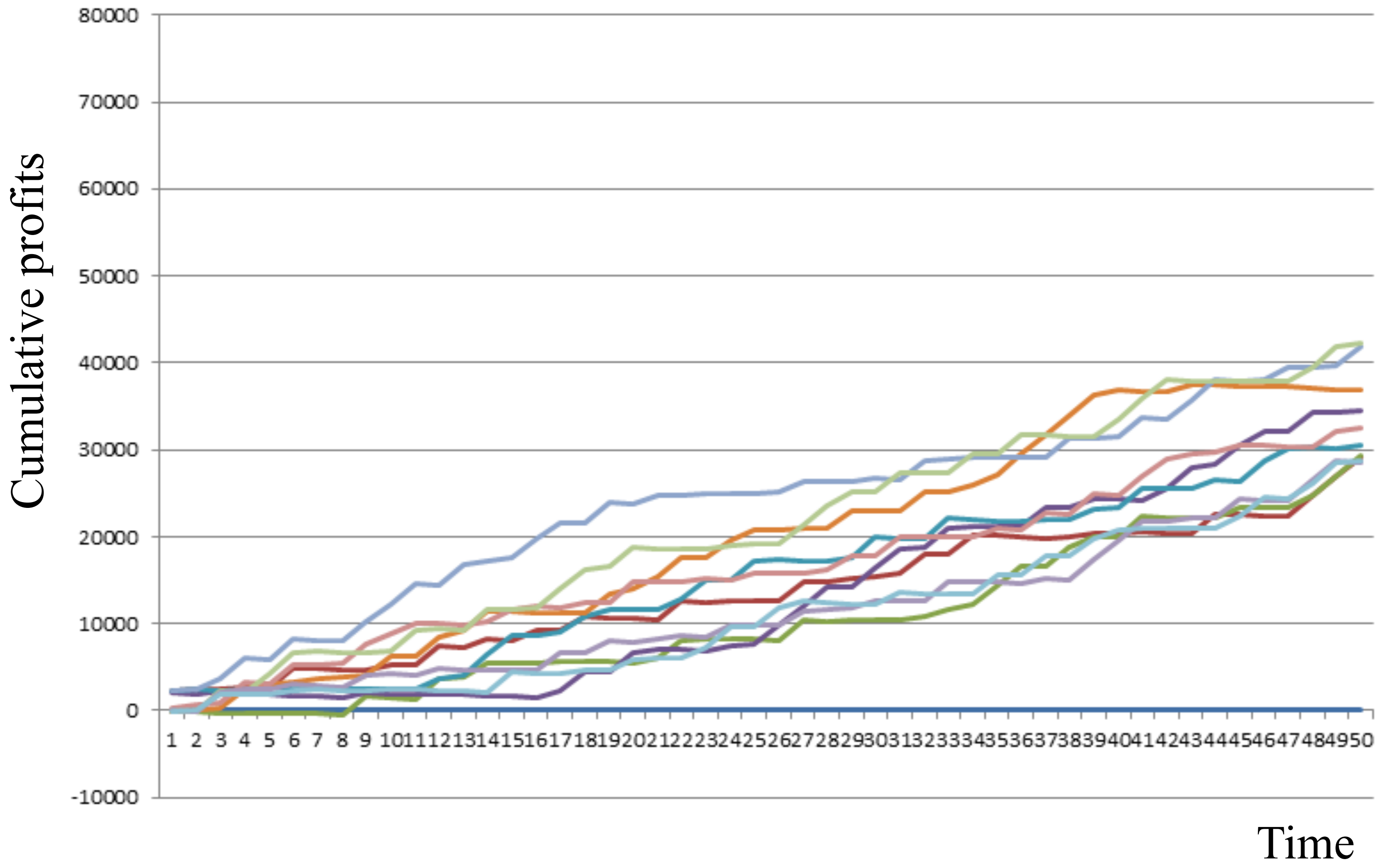
Hedging risk

- Hedge 35 MWh



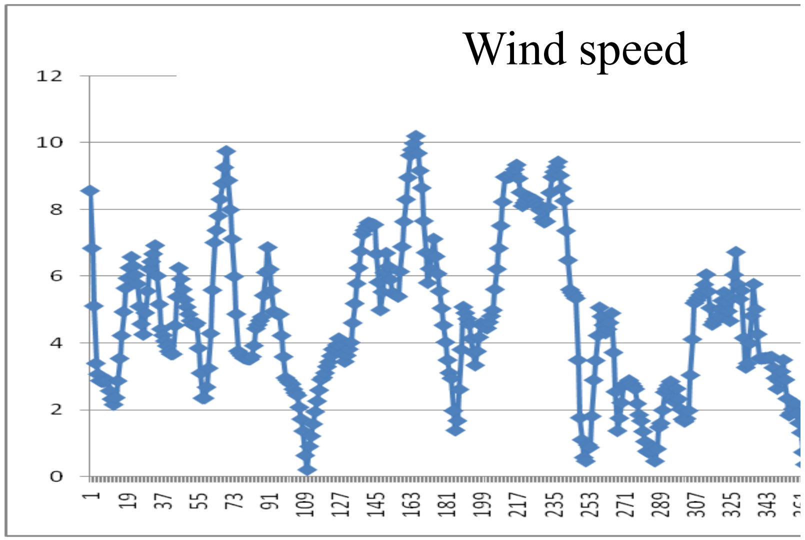
Hedging risk

- Hedge 70 MWh



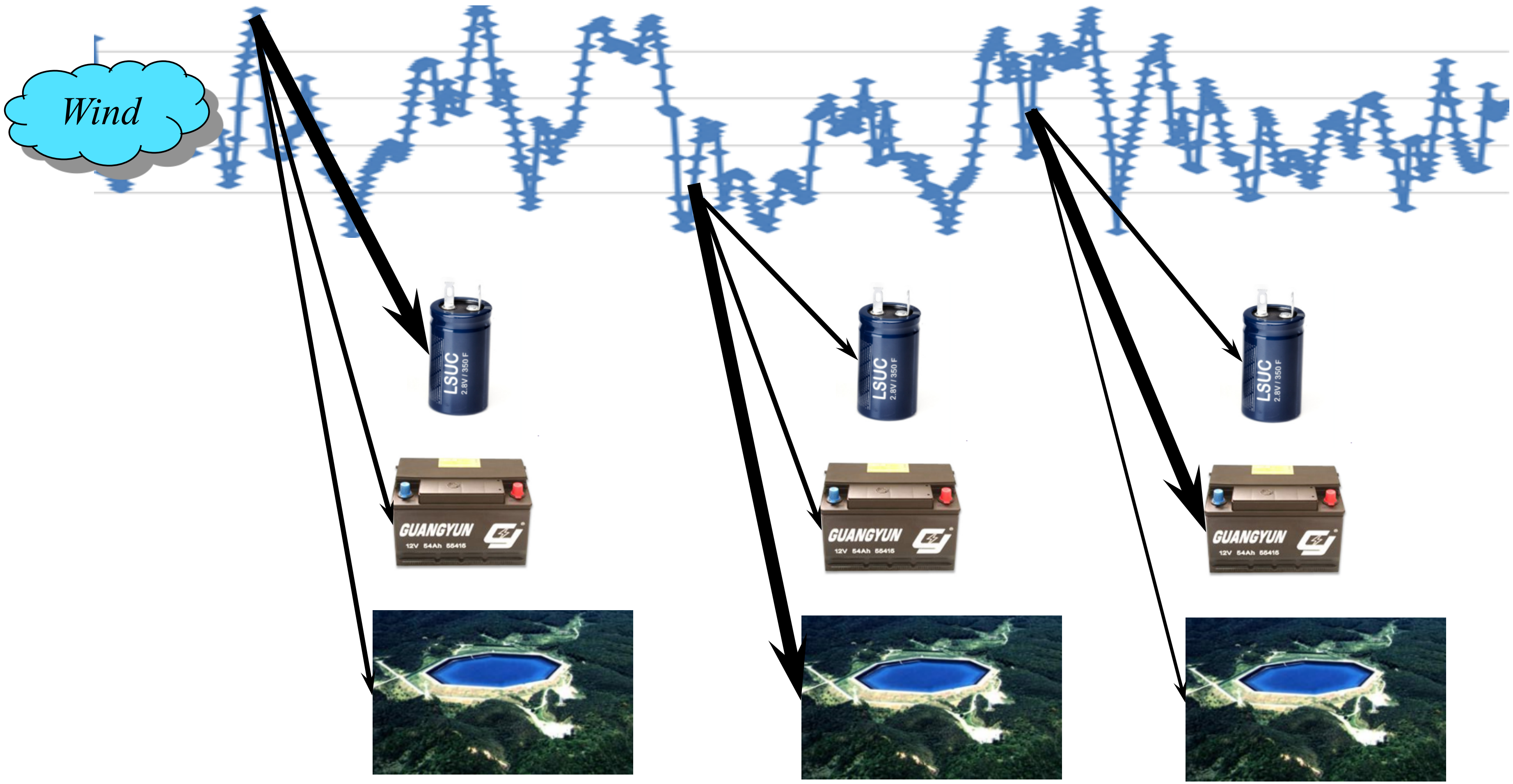
Storage

- Storage as a hedge against variations in
  - » Supply (wind, solar)
  - » Load
  - » Purchase cost of natural gas, electricity.
  - » Market price of selling energy

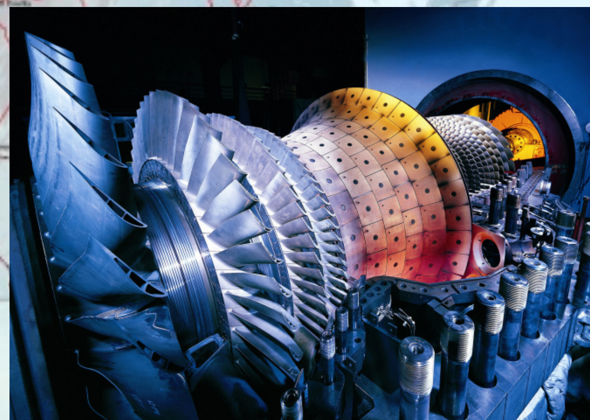


Energy storage portfolios

- Designing a dynamic storage control policy for portfolios of storage devices.

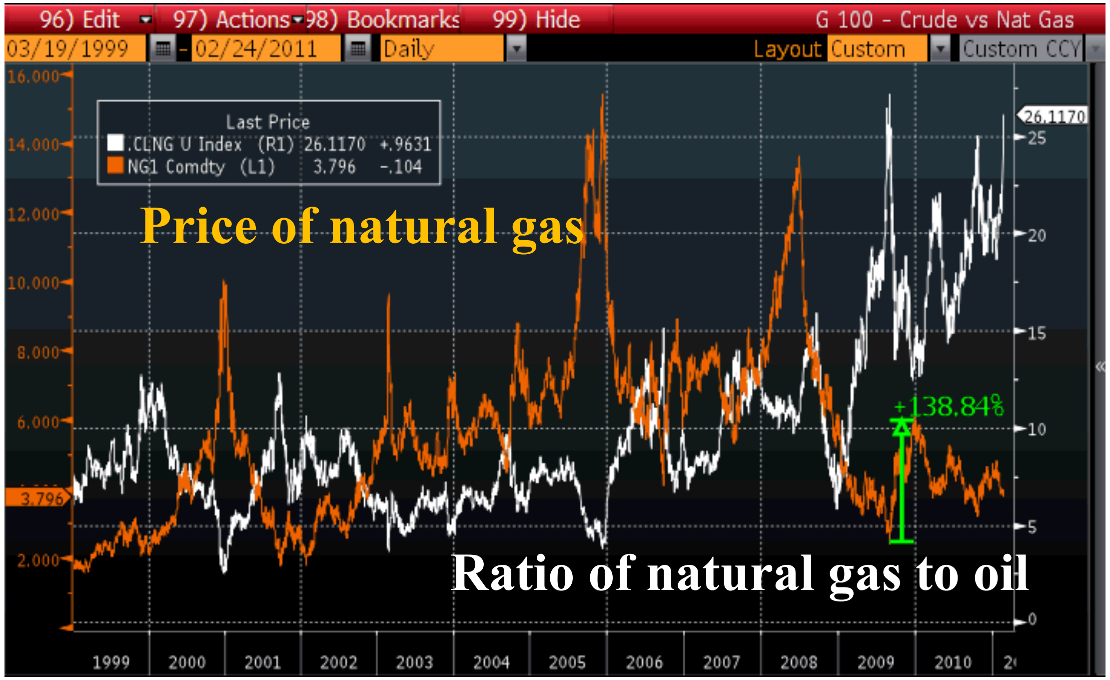


Meeting variability with *portfolios* of generation with mixtures of *dispatchability*

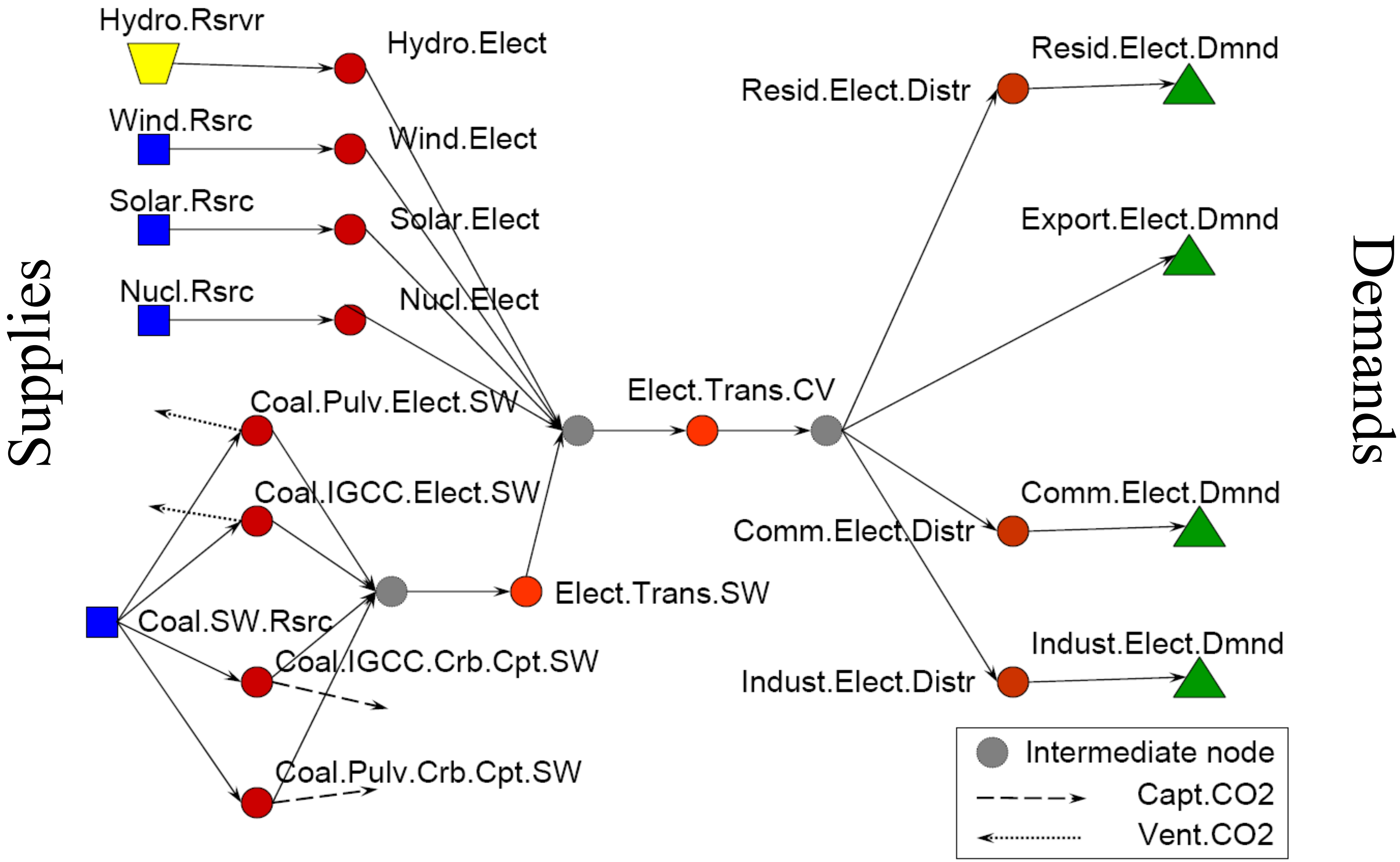


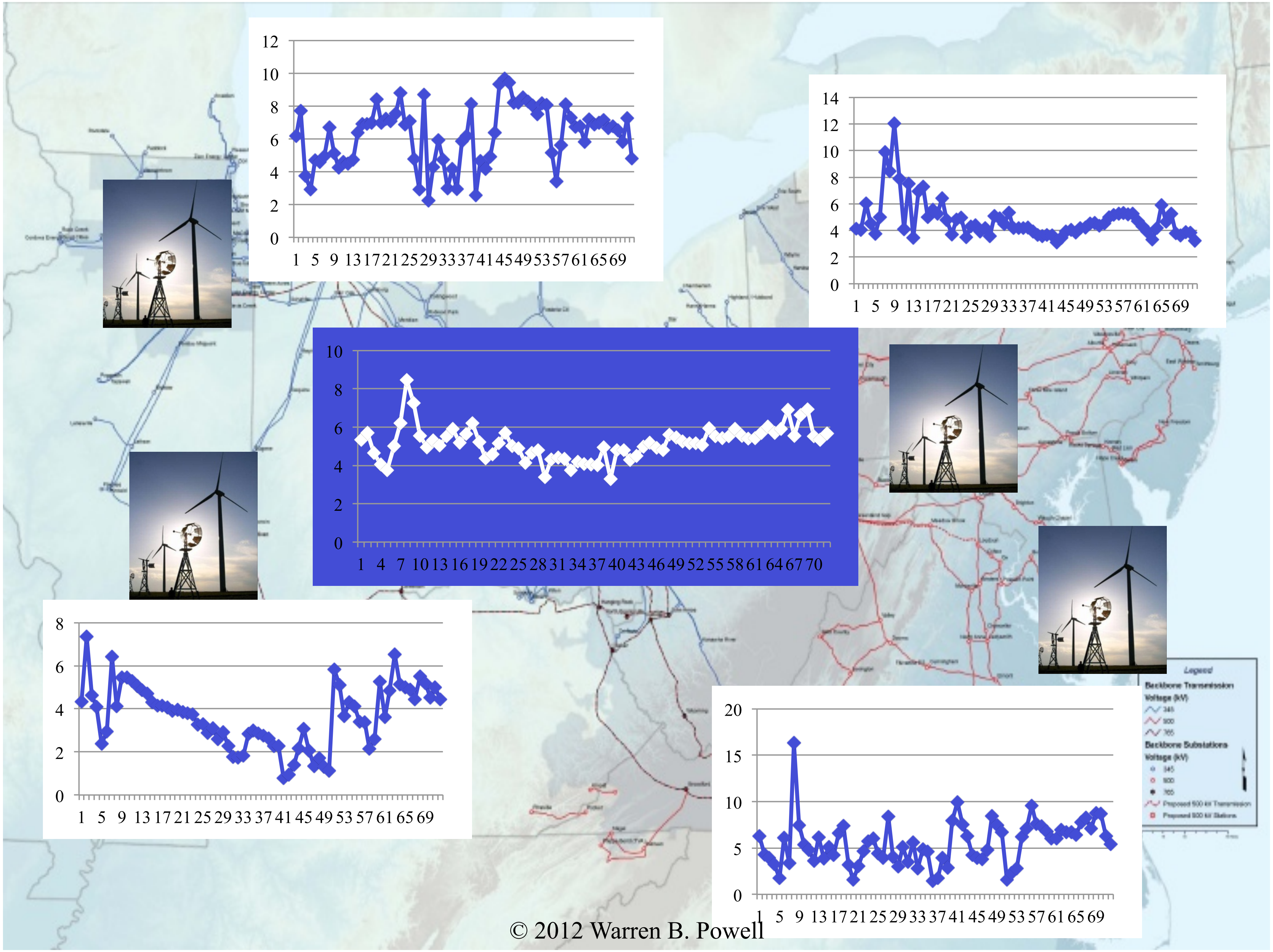
Commodity prices

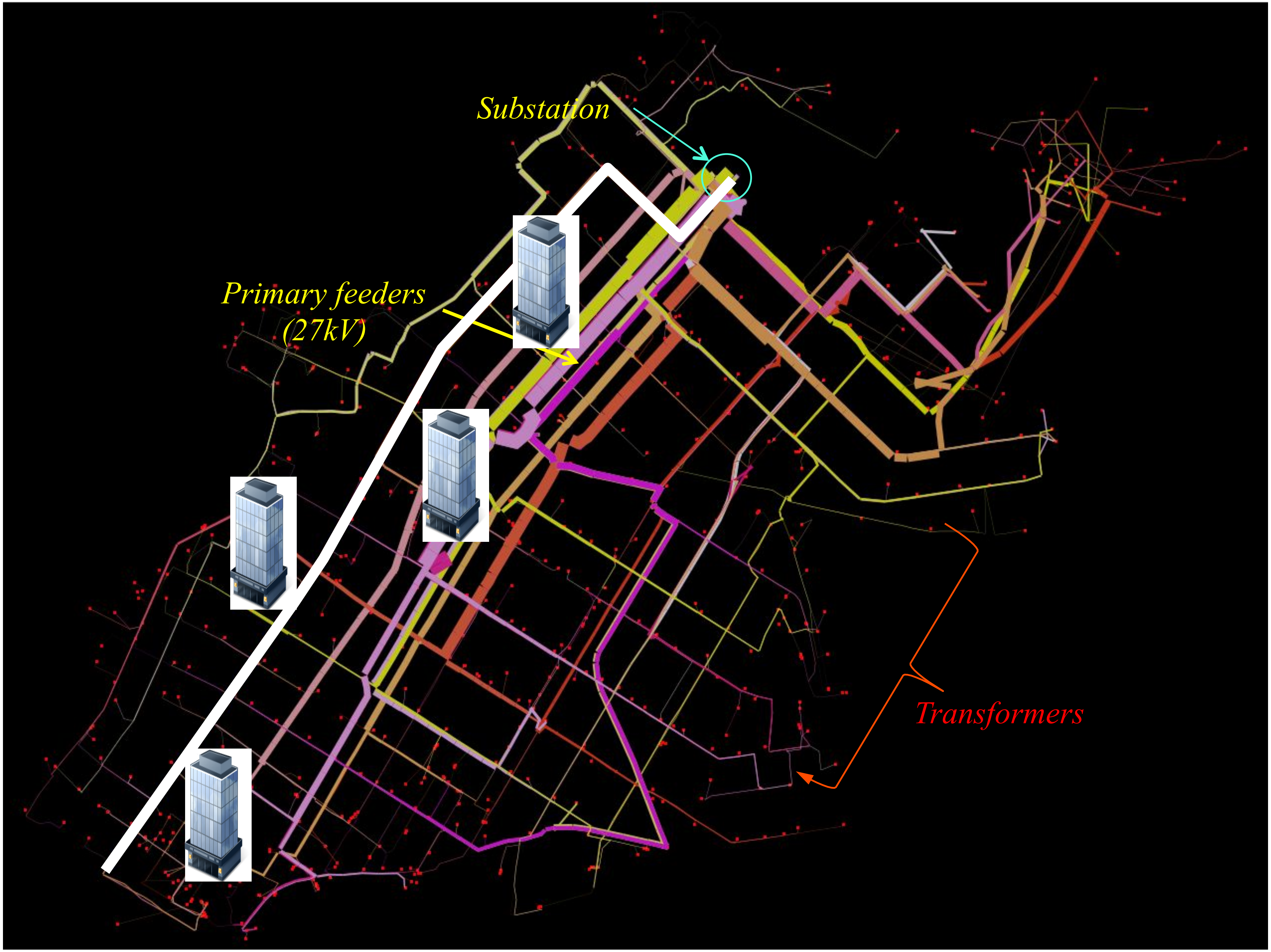
- Dual use power plants
  - » There is tremendous uncertainty in the *relative* cost of natural gas and oil.
  - » Plants which can burn gas and oil provide generators with the option to switch, limiting their exposure to price spikes of one commodity.

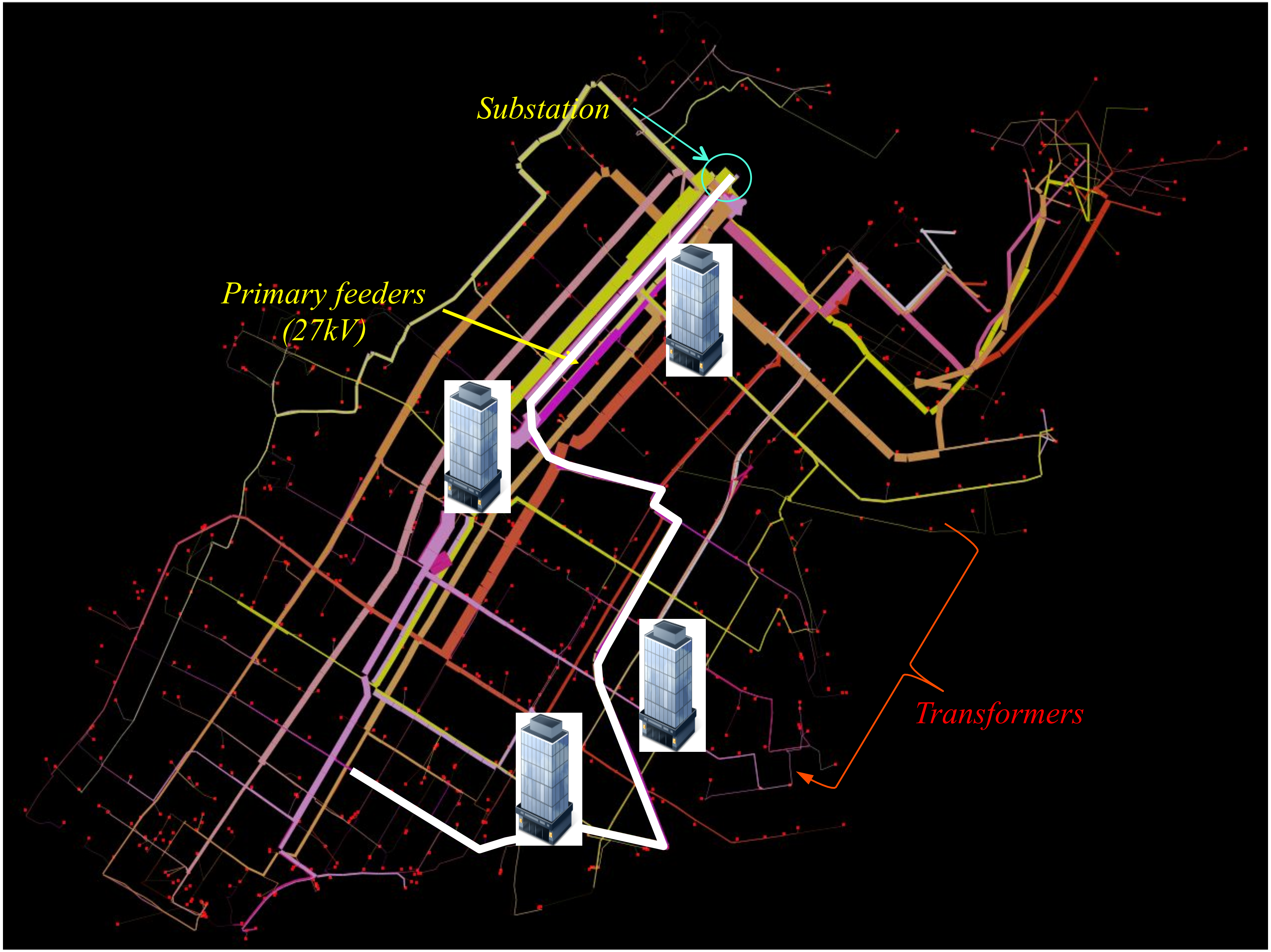


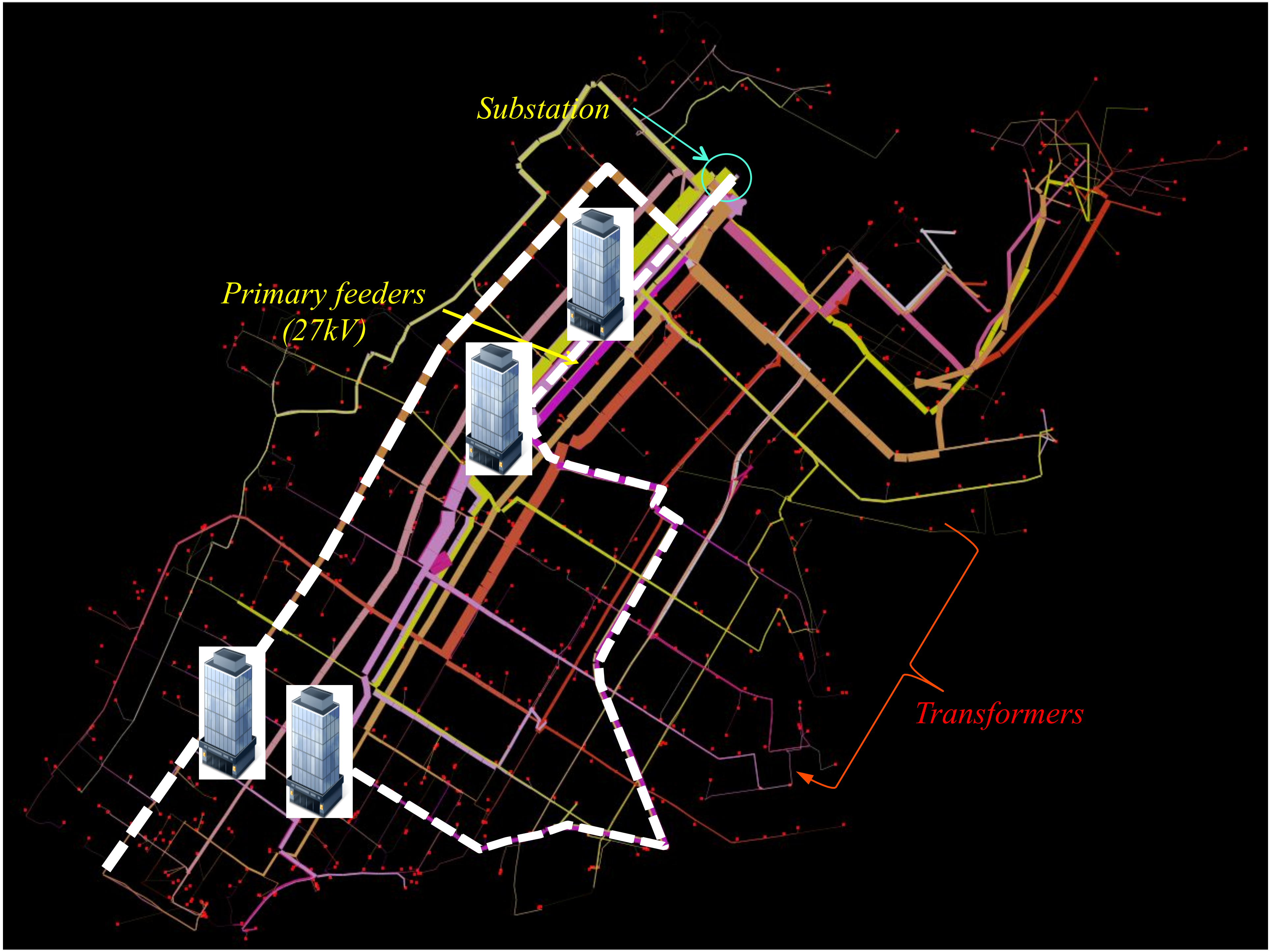
Substitutable resources



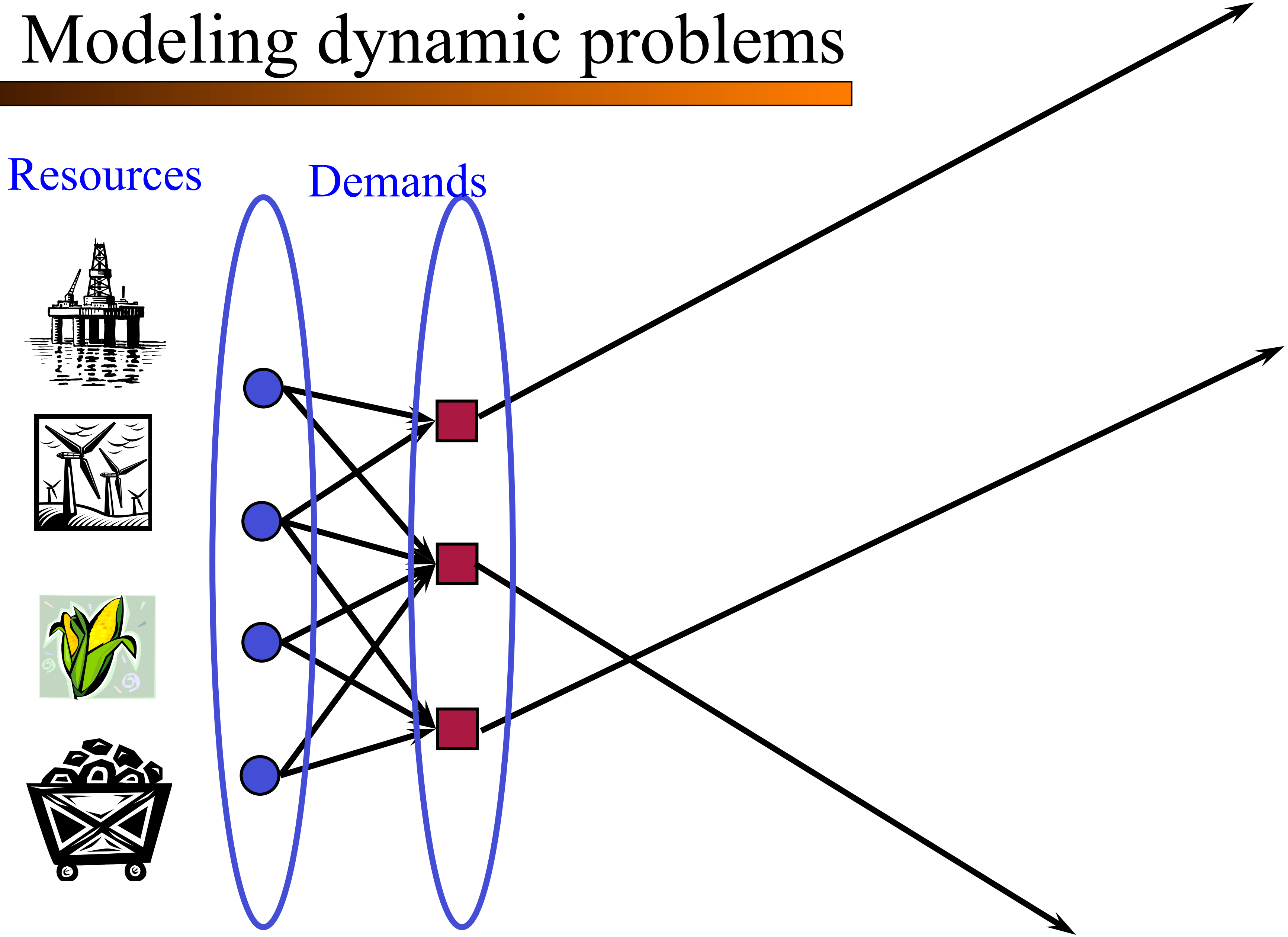

0 2 4 6 8 10 12
1 5 9 13 17 21 25 29 33 37 41 45 49 53 57 61 65 69
0 2 4 6 8 10 12 14
1 5 9 13 17 21 25 29 33 37 41 45 49 53 57 61 65 69
0 2 4 6 8 10
1 4 7 10 13 16 19 22 25 28 31 34 37 40 43 46 49 52 55 58 61 64 67 70
0 2 4 6 8
1 5 9 13 17 21 25 29 33 37 41 45 49 53 57 61 65 69
0 5 10 15 20
1 5 9 13 17 21 25 29 33 37 41 45 49 53 57 61 65 69


Substation
Primary feeders
(27kV)
Transformers

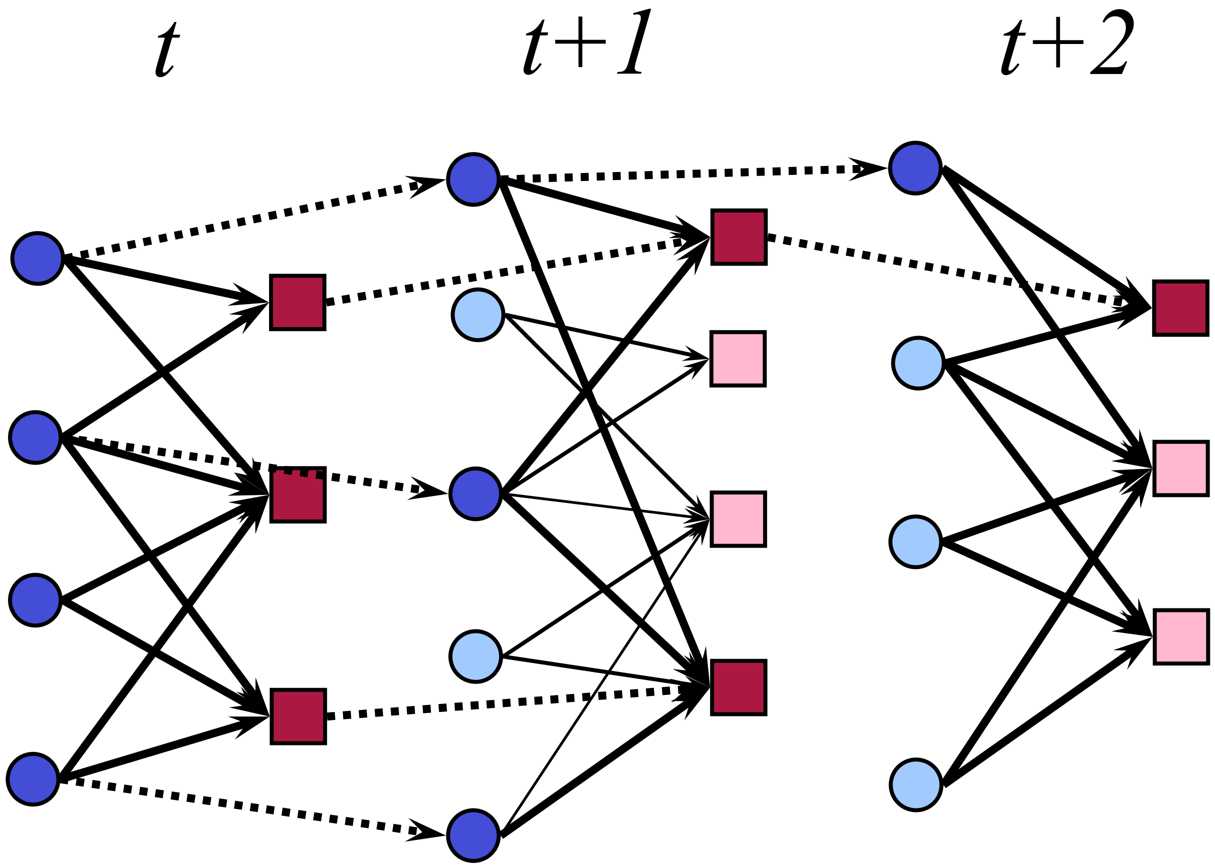

Substation
Primary feeders
(27kV)
Transformers


Substation
Primary feeders
(27kV)
Transformers

Modeling dynamic problems



Modeling dynamic problems



Modeling dynamic problems