Models and Algorithms for Energy Markets With High Penetrations of Renewables

CPAIOR 2013
IBM T.J. Watson

May 18, 2013

Warren B. Powell
PENSA Laboratory
Princeton University
Dept of Operations Research and Financial Engineering
http://energysystems.princeton.edu

The PENSA team

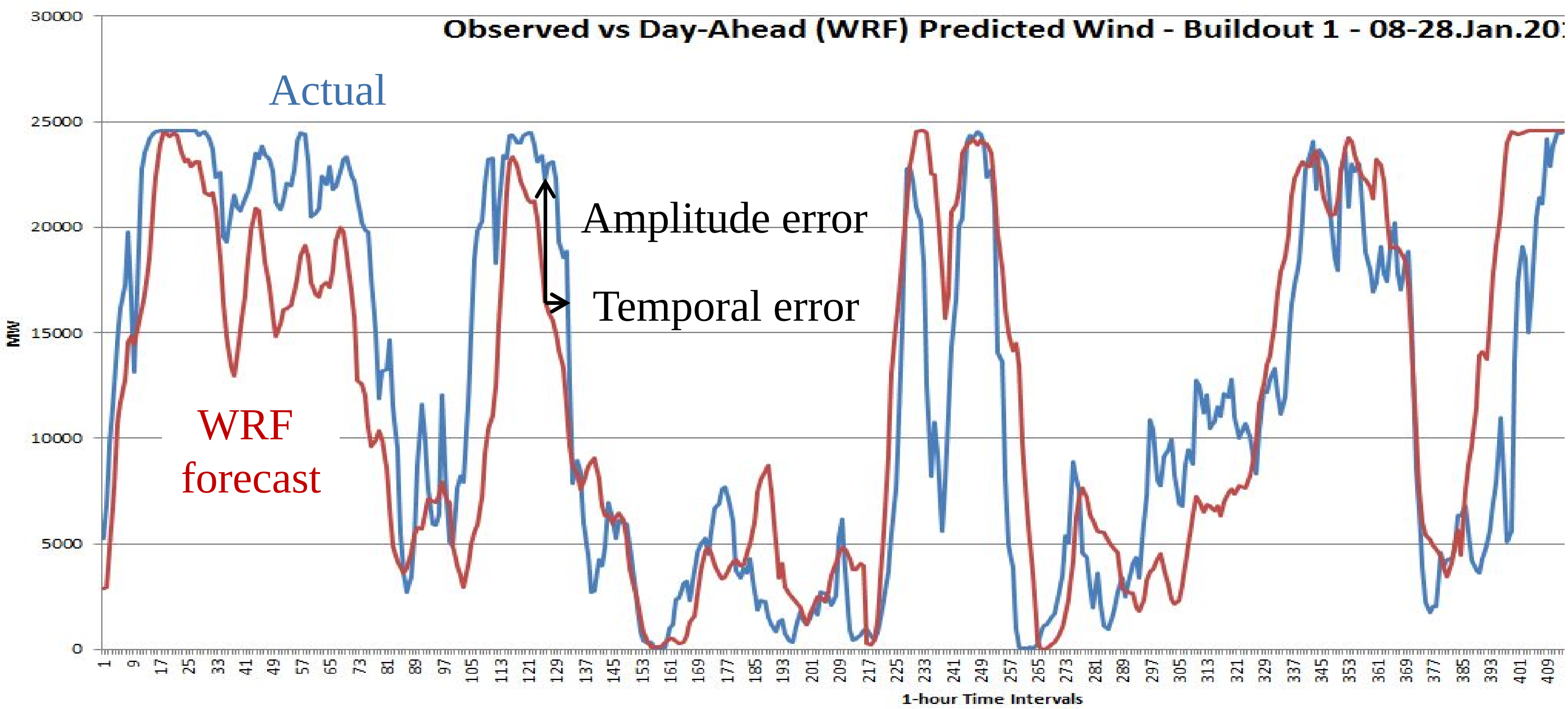
- Faculty
  - » Warren Powell (Director)
  - » Ronnie Sircar (ORFE)
  - » Craig Arnold (MAE)
  - » Elie Bou-Zeid (CEE)
  - » Rob Socolow (MAE)
  - » Sigurd Wagner (EE)
- Graduate students
  - » DH Lee (COS)
  - » Daniel Salas (CBE)
  - » Jinzhen Jin (CEE)
  - » Danial Jiang (ORFE)
  - » Harvey Cheng (EE)
  - » Vincent Pham (ORFE)
  - » Warren Scott (ORFE)
- Staff/post-docs
  - » **Hugo Simao (deputy director)**
  - » **Boris Defourny**
  - » Ricardo Collado
  - » Somayeh Moazeni
  - » Javad Khazaei
  - » Michael Coulon
- Undergraduate interns (2012)
  - » Tarun Sinha (MAE)
  - » Stephen Wang (ORFE)
  - » Henry Chai (ORFE)
  - » Ryan Peng (ORFE)
  - » Christine Feng (ORFE)
  - » Joe Yang (ORFE)
  - » Austin Wang (ORFE)

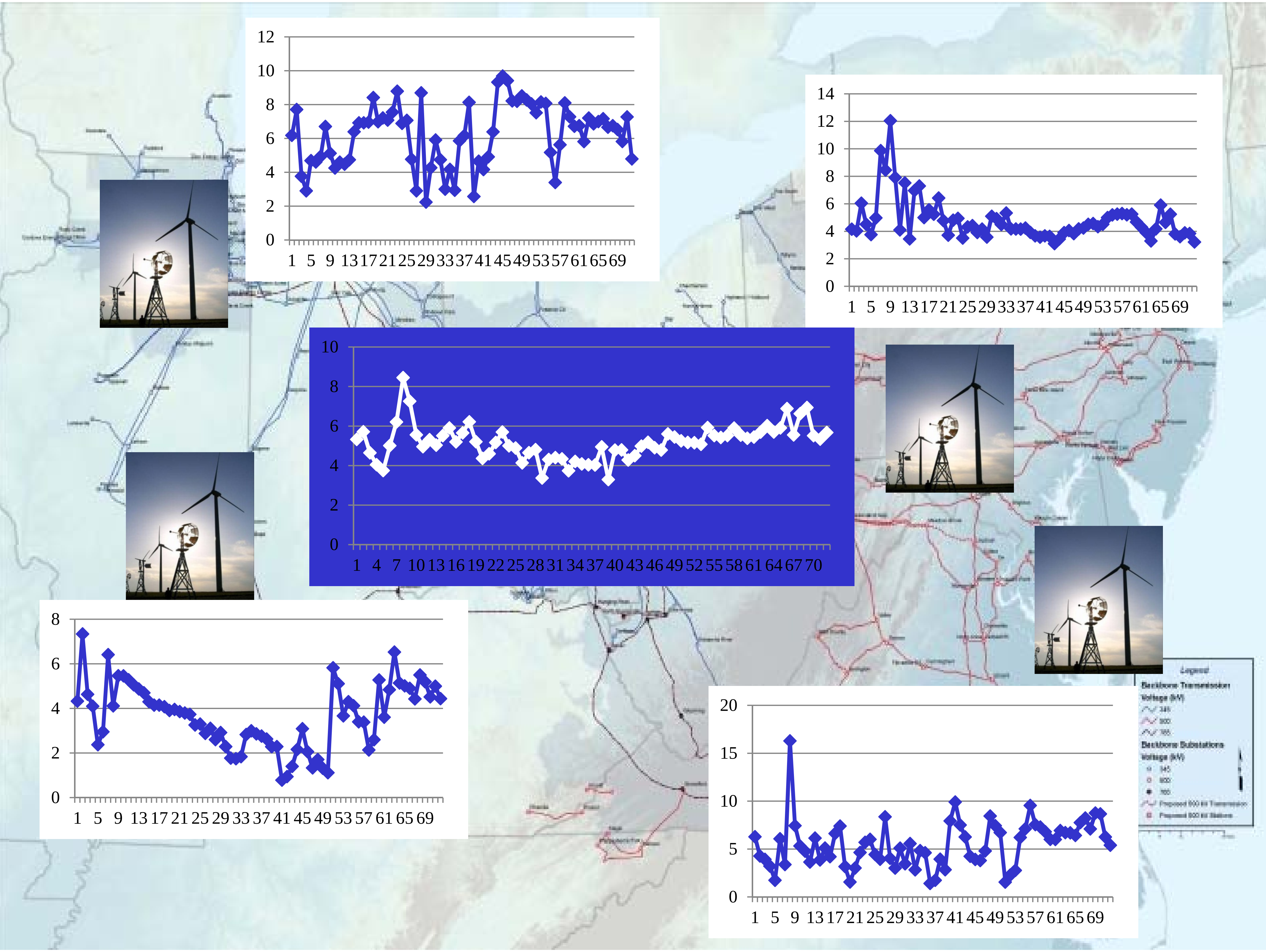
Lecture outline

- Types of uncertainty
- Modeling stochastic, dynamic systems
- Optimizing energy storage
  - Using Bellman error minimization
  - Using policy search and optimal learning
- SMART-ISO – Robust unit commitment using a lookahead policy
- Observations

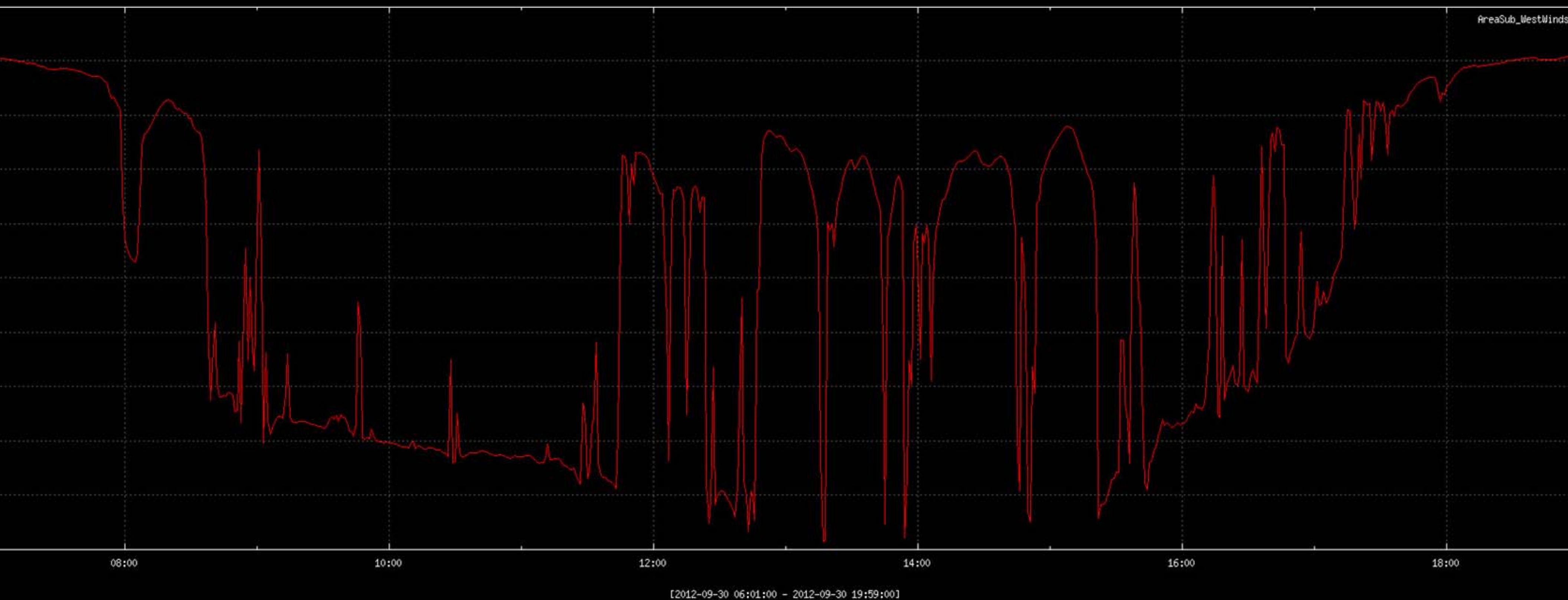
Modeling wind forecast errors

- We have to model two types of errors:
  - » Amplitude error – Errors in how much energy will be generated by wind.
  - » Temporal error – Errors in the timing of wind shifts.



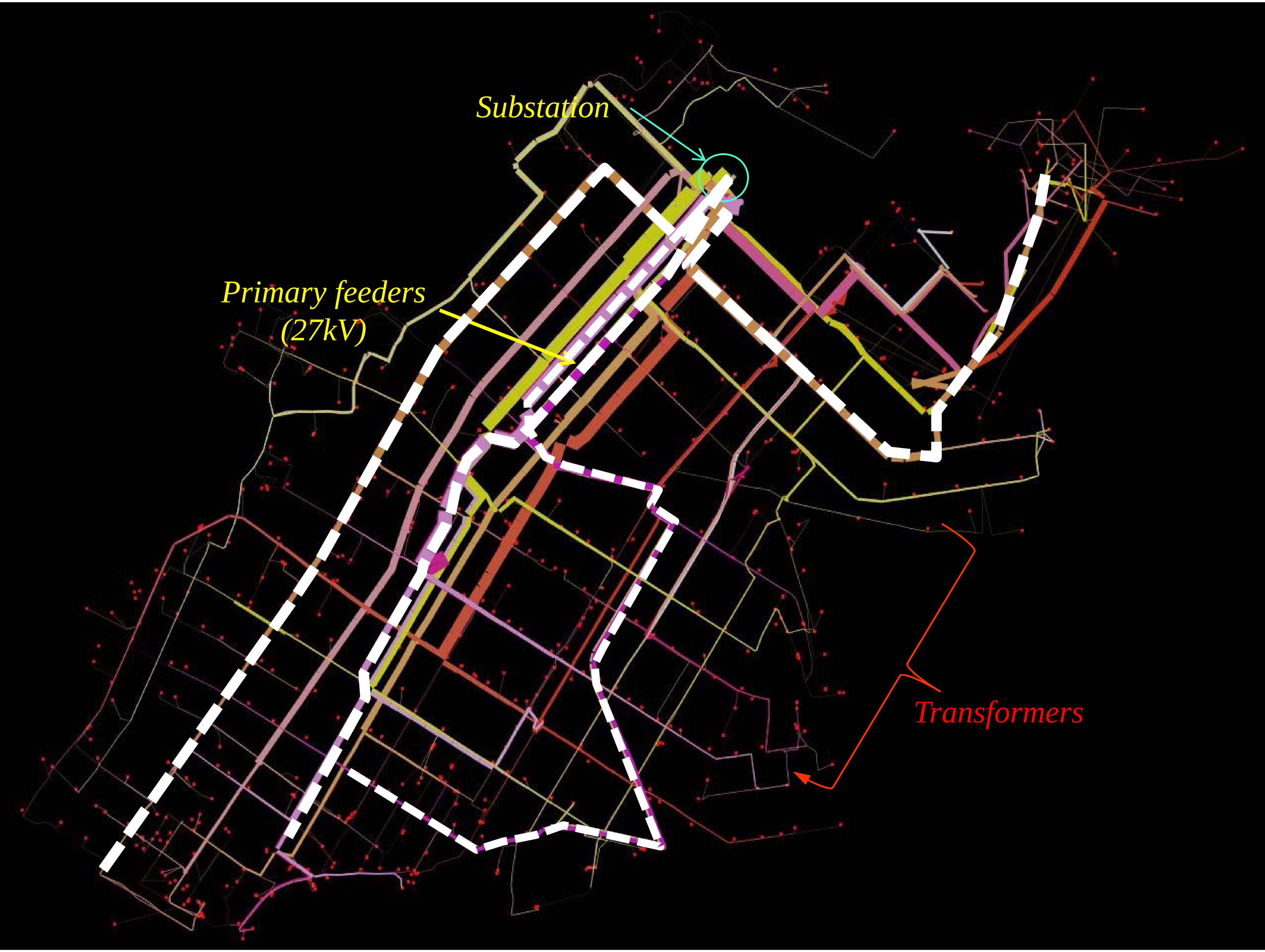


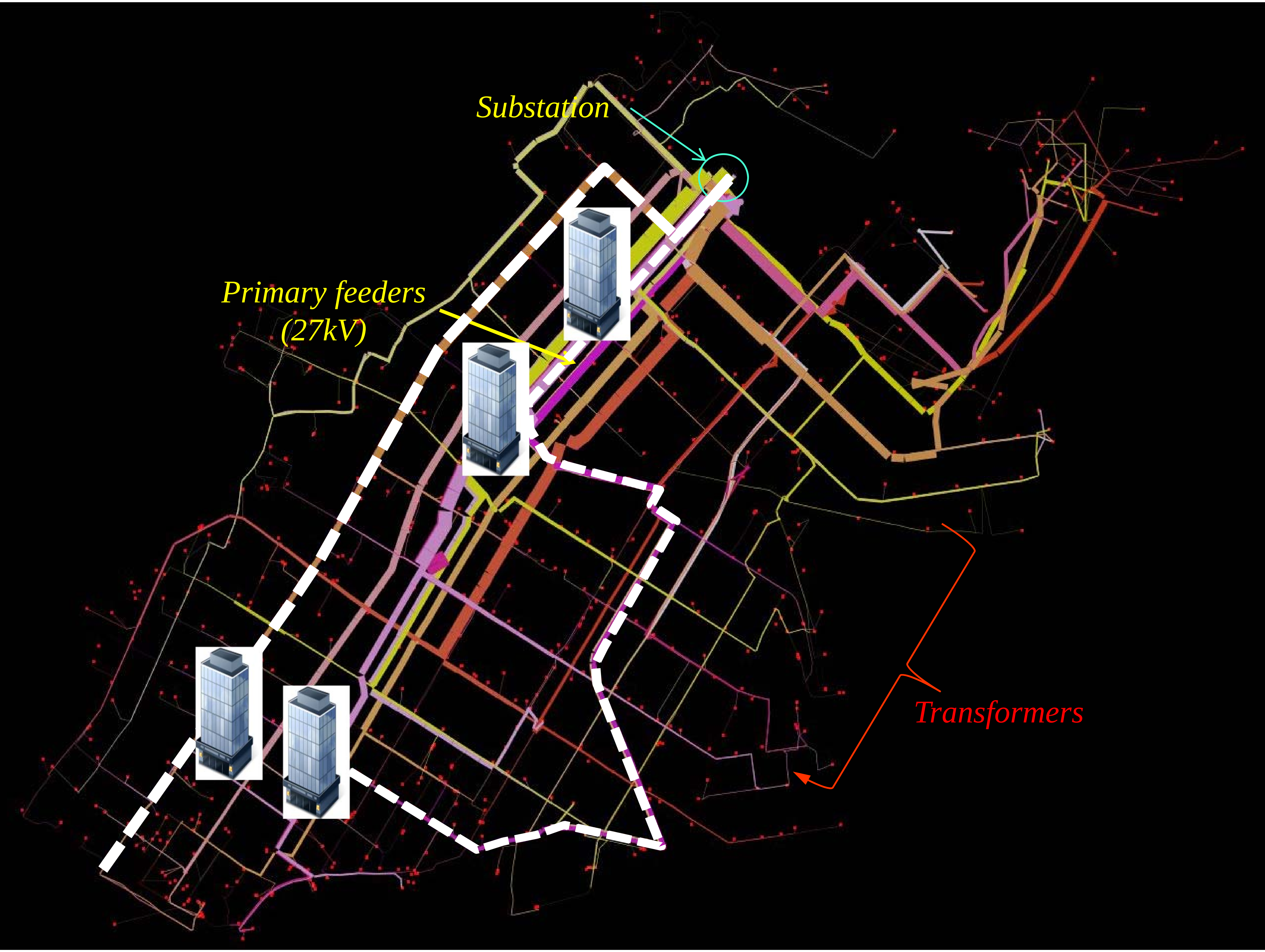
12
10
8
6
4
2
0
1 5 9 13 17 21 25 29 33 37 41 45 49 53 57 61 65 69
14
12
10
8
6
4
2
0
1 5 9 13 17 21 25 29 33 37 41 45 49 53 57 61 65 69
10
8
6
4
2
0
1 4 7 10 13 16 19 22 25 28 31 34 37 40 43 46 49 52 55 58 61 64 67 70
8
6
4
2
0
1 5 9 13 17 21 25 29 33 37 41 45 49 53 57 61 65 69
20
15
10
5
0
1 5 9 13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


AreaSub_WestWinds
08:00
10:00
12:00
14:00
16:00
18:00
[2012-09-30 06:01:00 - 2012-09-30 19:59:00]

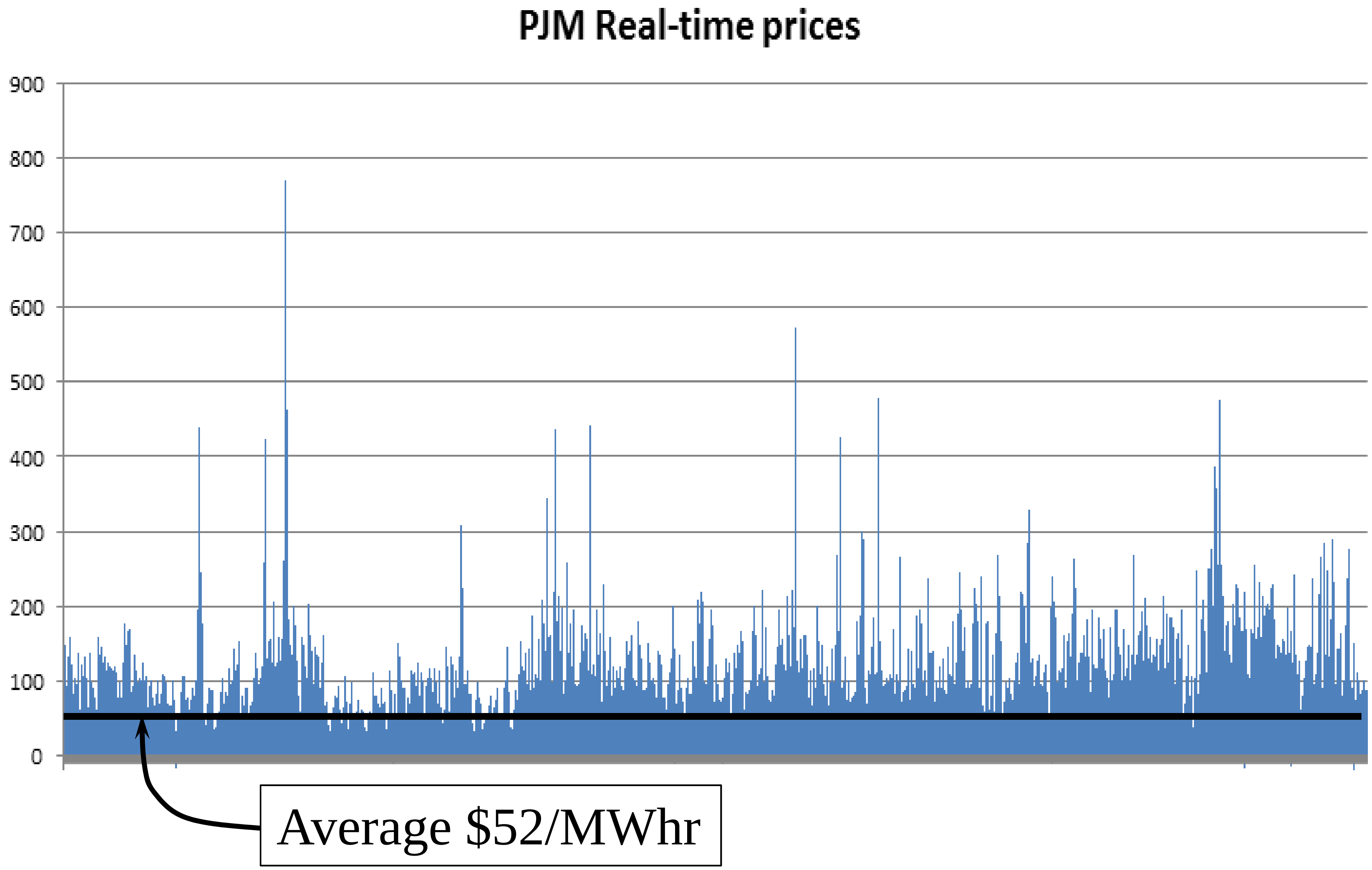

Gray Buildings © 2008 Sanborn
Image © 2011 DigitalGlobe
©2010 Google
Imagery Dates: Oct 1, 2006 - Jun 18, 2010
40°45'20.50" N 73°59'08.54" W elev 567 ft
Eye alt 2007 ft


Secondary
mesh (120V)


Substation
Primary feeders
(27kV)
Transformers


Substation
Primary feeders
(27kV)
Transformers

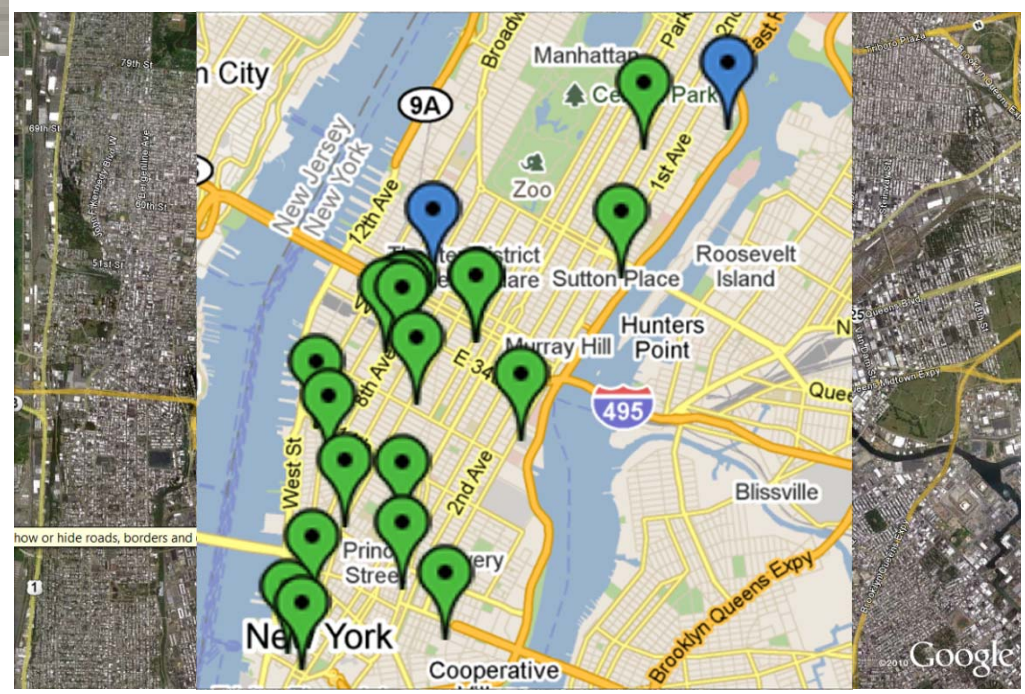
Electricity spot prices



Electric vehicles

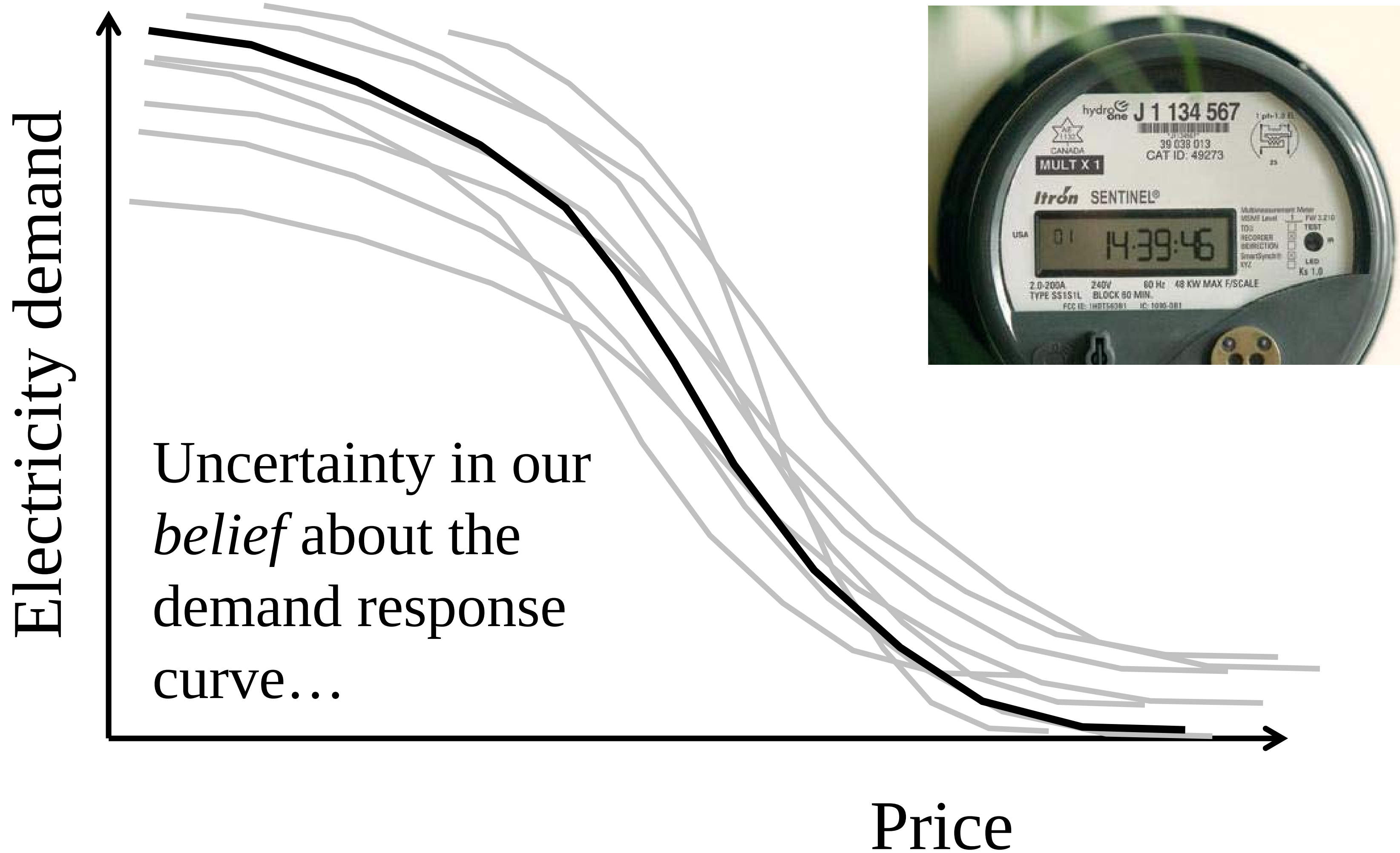
- The vehicle
  - How do we dynamically price electricity at charging stations?
  - How will drivers adapt to limited/slow recharging?

- The grid
  - What impact will EV's have on grid capacity?
  - How can buildings be managed to handle anticipated shortages?



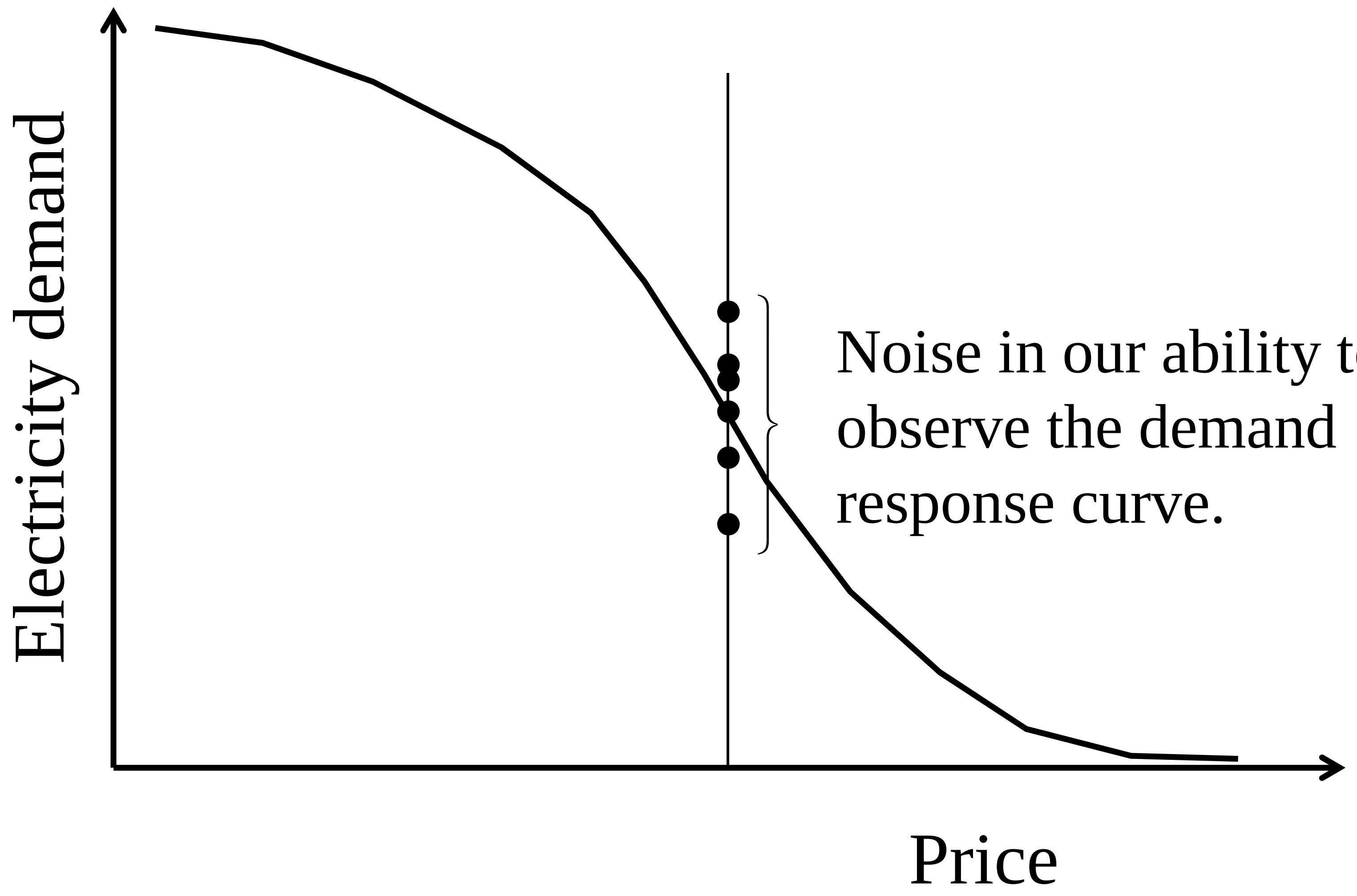
Uncertainty in models

- How will the market respond to price signals?



Uncertainty in models

- ❑ Making observations



Disasters

- Hurricane Sandy
  - » Once in 100 years?
  - » Rare convergence of events
  - » But, meteorologists did an amazing job of forecasting the storm.

- The power grid
  - » Loss of power creates cascading failures (lack of fuel, inability to pump water)
  - » How to plan?
  - » How to react?

Lecture outline

- Types of uncertainty
- Modeling stochastic, dynamic systems
- Optimizing energy storage
  - Using Bellman error minimization
  - Using policy search and optimal learning
- SMART-ISO – Robust unit commitment using a lookahead policy
- Observations

Deterministic modeling

- For deterministic problems, we speak the language of mathematical programming
  - » For static problems

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

  - » For time-staged problems

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

Arguably Dantzig's biggest contribution, more so than the simplex algorithm, was his articulation of optimization problems in a standard format, which has given algorithmic researchers a common language.


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

Modeling dynamic problems

- We lack a standard language for modeling sequential, stochastic decision problems.
  - » In the slides that follow, we propose to model problems along five fundamental dimensions:
    - State variables
    - Decision variables
    - Exogenous information processes
    - Transition function
    - Objective function
  - » This framework is widely followed in the control theory community, and almost completely ignored in operations research and computer science.

Modeling dynamic problems

- The system state:

$$S_t = (R_t, I_t, K_t) = \text{System state, where:}$$

R_t = Resource state (physical state)

Energy investments, energy storage, ...

Status of generators

I_t = Information state

State of the technology (costs, performance)

Market prices (oil, coal)

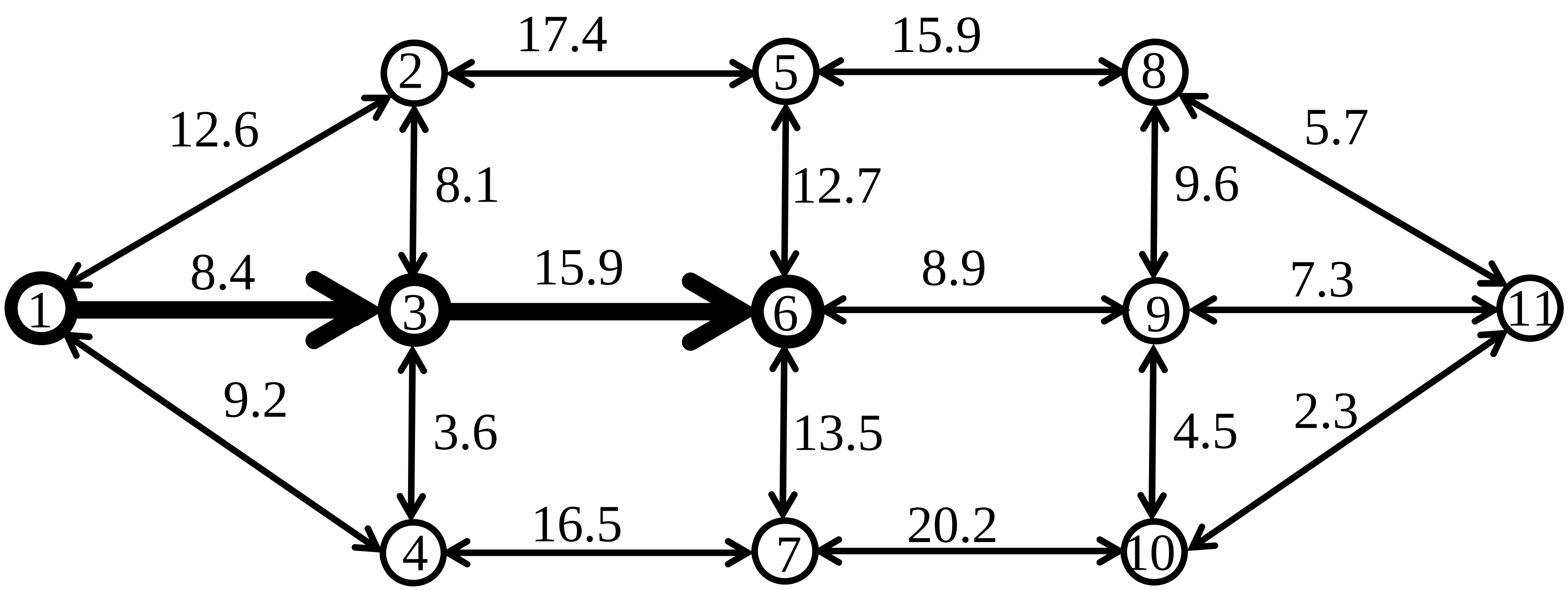
K_t = Knowledge state ("belief state")

Belief about the impact of electricity prices on demands

Belief about the effect of fertilizer on algal blooms

Modeling dynamic problems

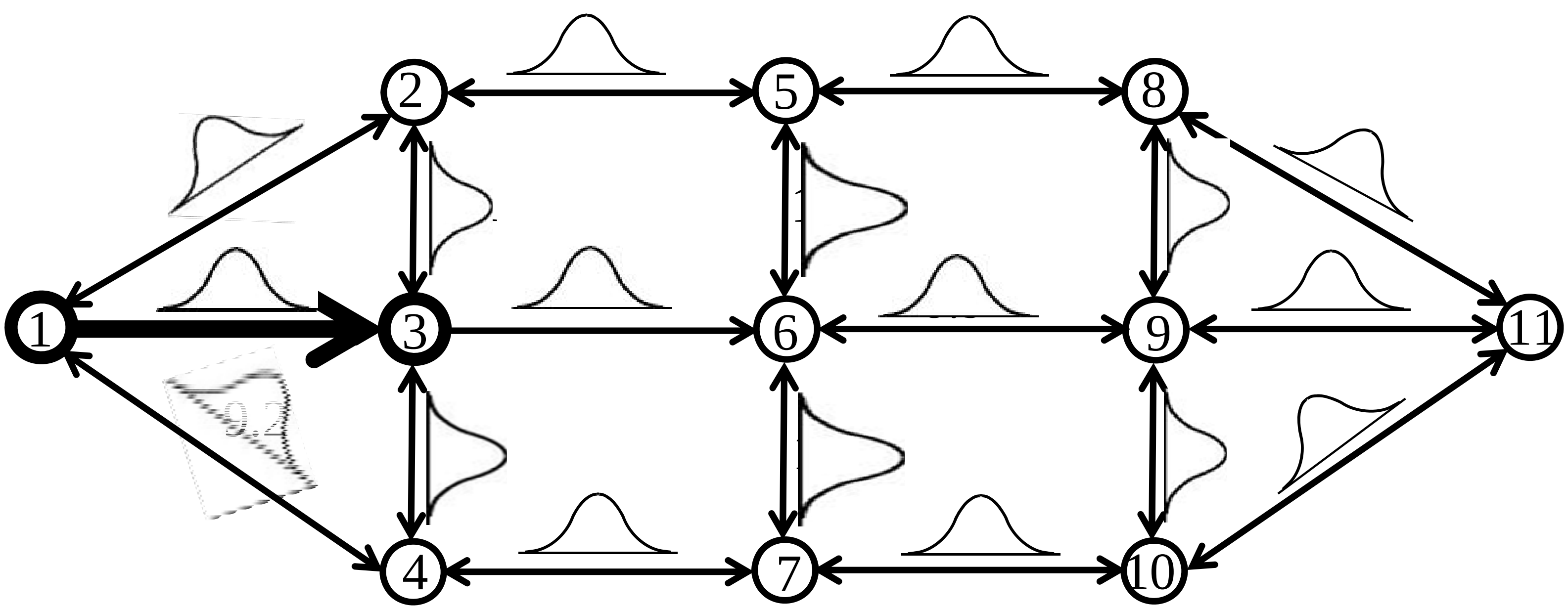
- Illustrating state variables
  - » A deterministic graph



$$S_t = (N_t) = 6$$

Modeling dynamic problems

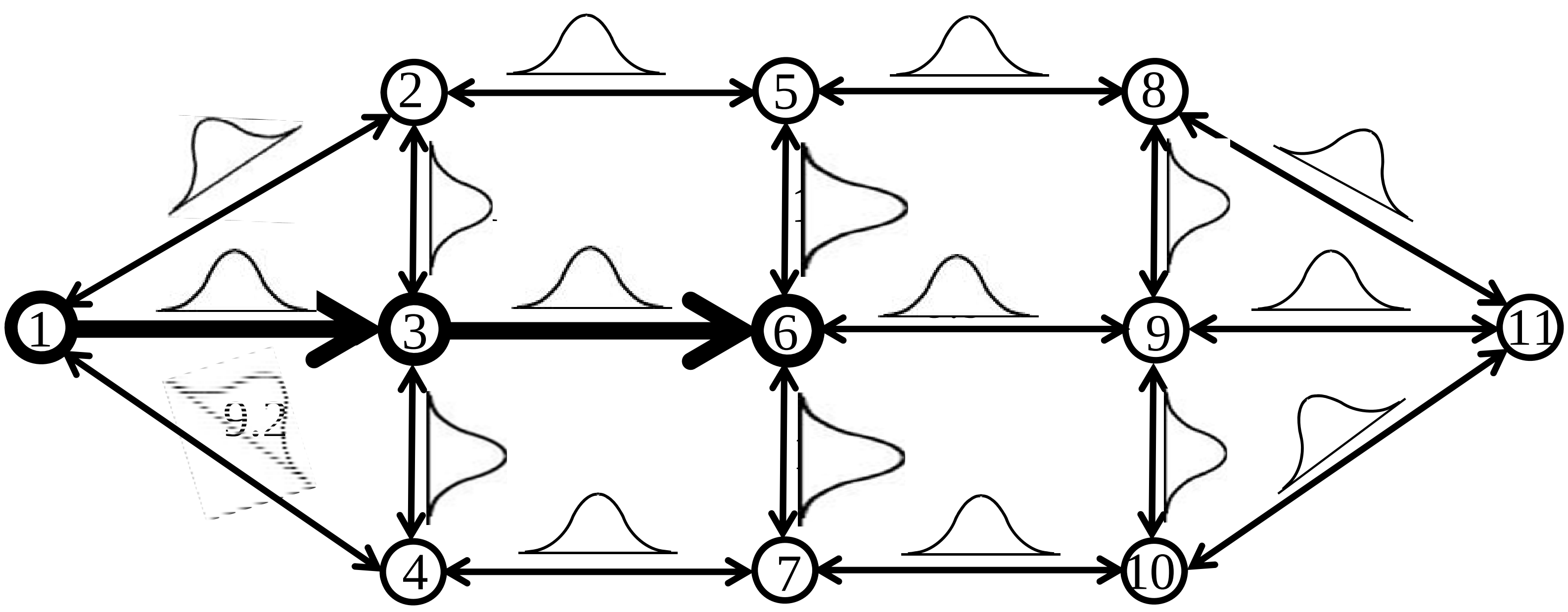
- Illustrating state variables
  - » A stochastic graph



$$S_t = ?$$

Modeling dynamic problems

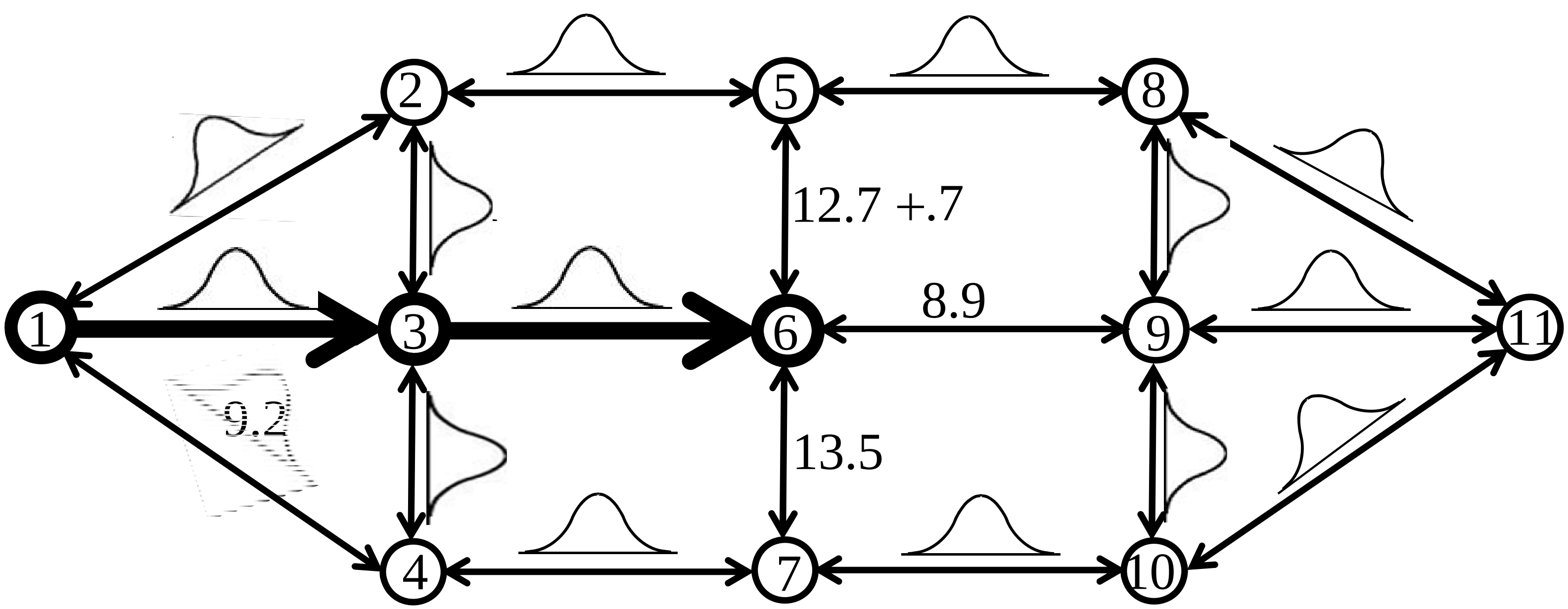
- Illustrating state variables
  - » A stochastic graph



$$S_t = \left(N_t, \left(c_{t,N_t,j} \right)_j \right) = \left(6, (12.7, 8.9, 13.5) \right)$$

Modeling dynamic problems

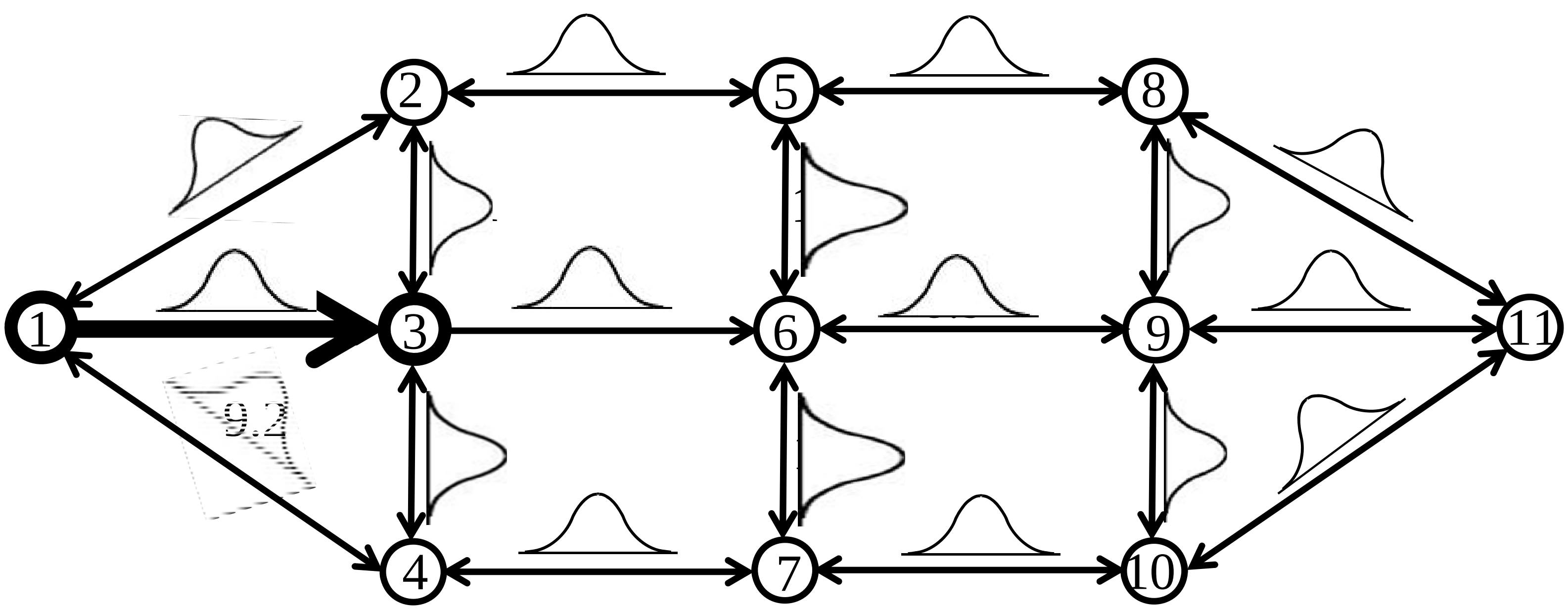
- Illustrating state variables
  - » A stochastic graph with left turn penalties



$$S_t = \Big(\underbrace{N_t}_{R_t}, \underbrace{\left(c_{t,N_t,j}\right)_j, N_{t-1}}_{I_t}\Big) = \left(6, (12,7,8.9,13.5), 3\right)$$

Modeling dynamic problems

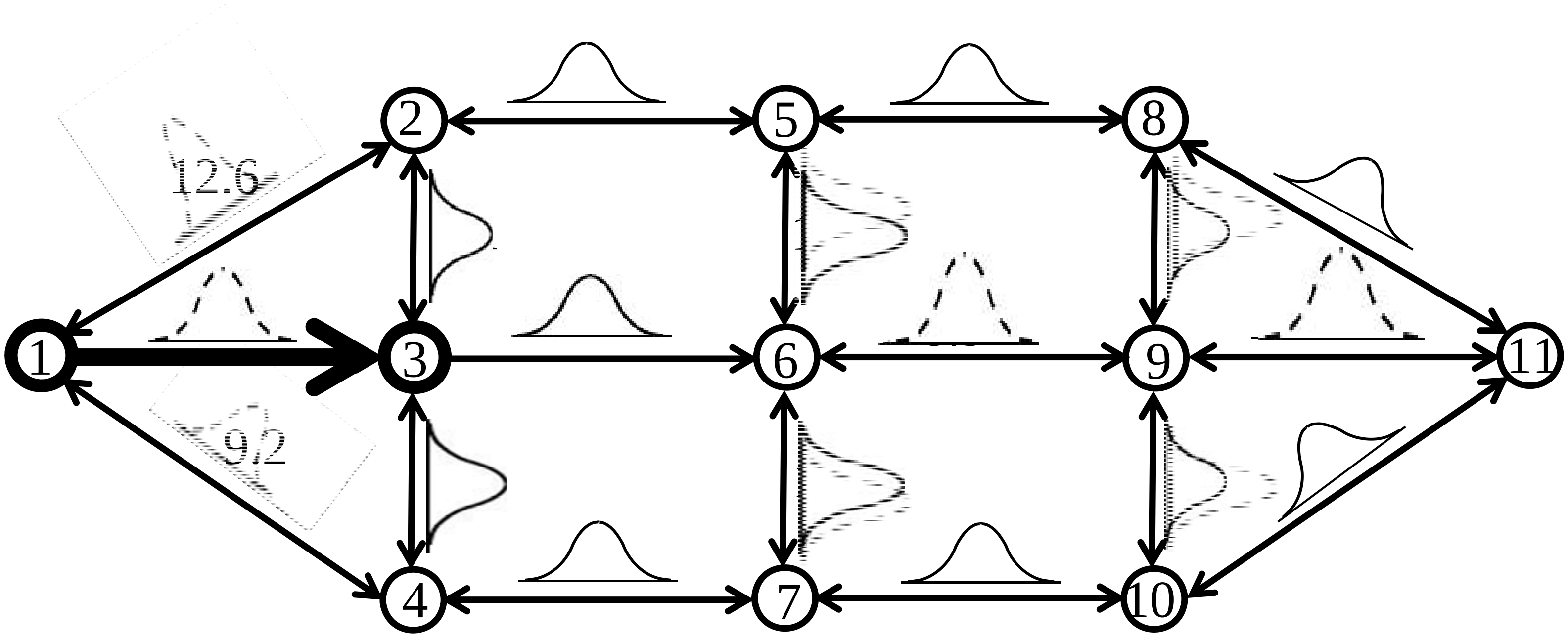
- Illustrating state variables
  - » A stochastic graph with generalized learning



$$S_t = ?$$

Modeling dynamic problems

- Illustrating state variables
  - » A stochastic graph with generalized learning



$$S_t = \Big(\underbrace{N_t}_{R_t}, \underbrace{\left(c_{t,N_t,j}\right)_j}_{I_t}, \underbrace{\quad}_{K_t} \Big)$$

Modeling dynamic problems

- A proposed definition of a state variable:
  - » *The state variable is the minimally dimensioned function of history that is necessary and sufficient to calculate the decision function, the cost/reward function, and the transition function.*
  - » *From:*

    Powell, W.B., *Approximate Dynamic Programming: Solving the Curses of Dimensionality*, Chapter 5, downloadable from

    http://adp.princeton.edu/

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.

To make a decision, we define

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

I prefer:

Let $A^\pi(s)$ (or $X^\pi(s)$ or $U^\pi(s)$), where π specifies the class of policy, and any tunable parameters (which we represent using θ).

Modeling dynamic problems

- Exogenous information:

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

$\hat{R}_t$ = 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

Note: Any variable indexed by t is known at time t. This convention, which is not standard in control theory, dramatically simplifies the modeling of information.

Modeling dynamic problems

- The transition function

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

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

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

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

Also known as the:

“System model”

“State transition model”

“Plant model”

“Model”

Stochastic optimization models

- The objective function

$$\min_{\pi} E^{\pi} \left\{ \sum_{t=0}^{T} \gamma^{t} C\left(S_t, X_t^{\pi}(S_t)\right) \right\}$$

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.

Objective functions

- There are different objectives that we can:
  - » Expectations

$$\min_x \mathbb{E}F(x,W)$$

  - » Risk measures

$$\min_x \mathbb{E}F(x,W)+\theta\mathbb{E}\left[F(x,W)-f_\alpha\right]_+^2$$

$$\min_x \rho\big(F(x,W)\big) \qquad \rho \in \text{Convex/coherent risk measures}$$

  - » Worst case ("robust optimization")

$$\min_x \max_w F(x,w)$$

Stochastic optimization models

- Definition:
  - *A policy is a mapping from a state to an action.*
  - *… any mapping.*
- Observation:
  - *From my experience, there are four fundamental classes of policies (for sequential decision problems).*

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 ***cost function approximation:***

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

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

Policies

2) Lookahead policies - Plan over the next T periods, but implement only the action it tells you to do now.

» Deterministic forecast

$$X_t^{LA-D}(S_t) = \arg\min_{\tilde{x}_{tt}, \tilde{x}_{t,t+1}, \ldots, \tilde{x}_{t,t+T}} C(\tilde{S}_{tt}, \tilde{x}_{tt}) + \sum_{t'=t+1}^{T} \gamma^{t'-t} C(\tilde{S}_{tt'}, \tilde{x}_{tt'})$$

» Stochastic programming (e.g. two-stage)

$$X_t^{LA-S}(S_t) = \arg\min_{\tilde{x}_{tt}, \left(\tilde{x}_{t,t+1}(\omega), \ldots, \tilde{x}_{t,t+T}(\omega)\right), \omega \in \tilde{\Omega}_t} C(\tilde{S}_{tt}, \tilde{x}_{tt}) + \sum_{\omega \in \tilde{\Omega}_t} p(\omega) \sum_{t'=t+1}^{T} \gamma^{t'-t} C(\tilde{S}_{tt'}(\omega), \tilde{x}_{tt'}(\omega))$$

» Rolling/receding horizon procedures

» Model predictive control

» Rollout heuristics

» Tree search (decision trees), Monte Carlo tree search

Four classes of policies

3) Policy function approximations

» Lookup table

- Recharge the battery between 2am and 6am each morning, and discharge as needed.

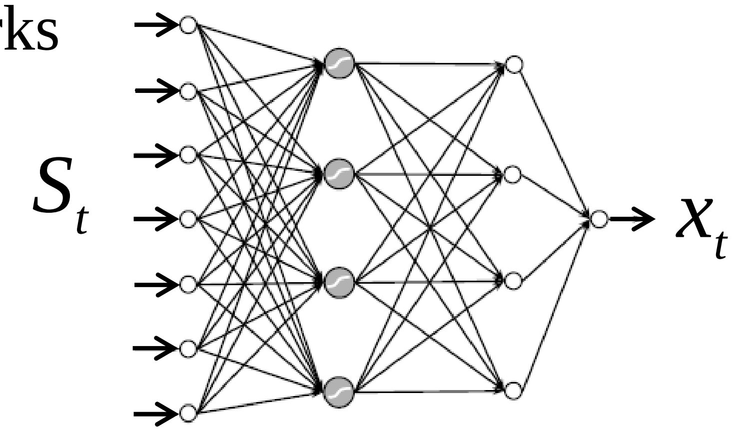
» Parameterized functions

- 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

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

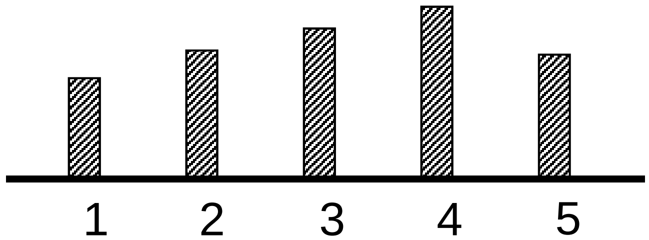
» Or the post-decision state:

$$X_t^{VFA}(S_t) = \arg\max_{x_t} \left(C(S_t, x_t) + \gamma \bar{V}_t^x \left(S_t^x(S_t, x_t) \right) \right)$$

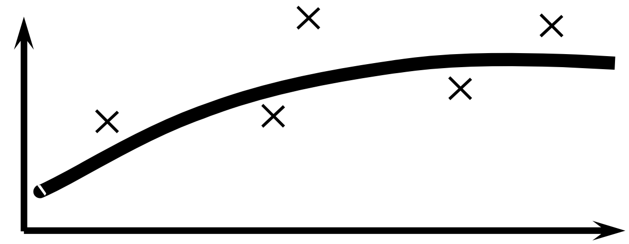
» This is what most people associate with "approximate dynamic programming" or "reinforcement learning"

Four classes of policies

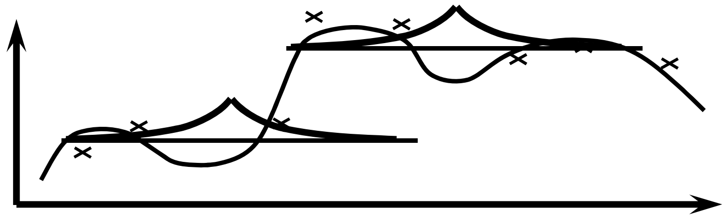
- ❑ There are three classes of approximation strategies
  - » Lookup table
    - • Given a discrete state, return a discrete action or value



  - » Parametric models
    - • Linear models ("basis functions")
    - • Nonlinear models
    - • Neural networks



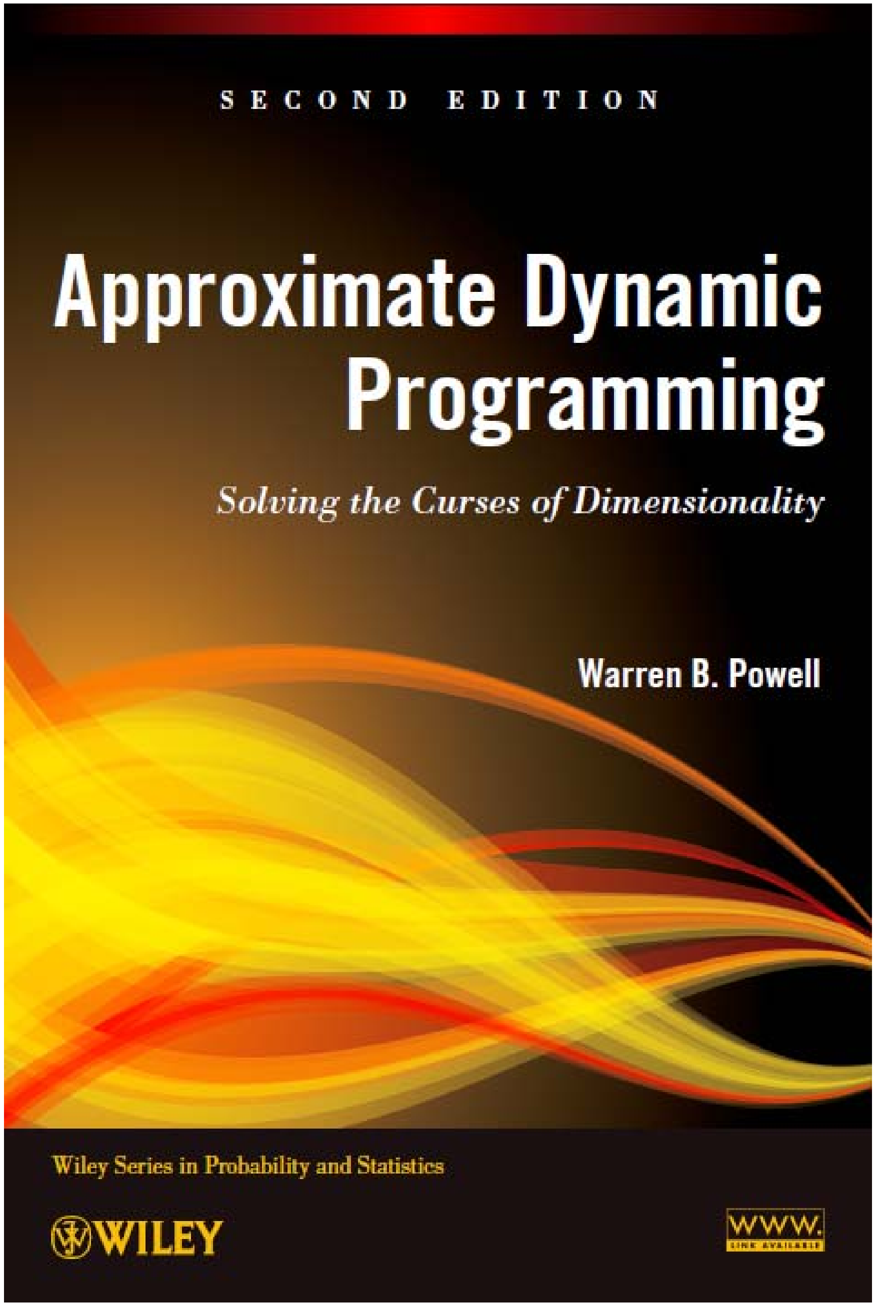
  - » Nonparametric/local parametric models
    - • Kernel regression
    - • Local polynomial



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/

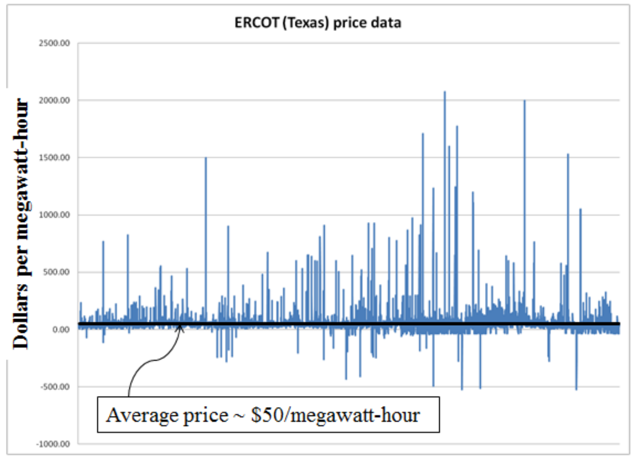


Lecture outline

- Types of uncertainty
- Modeling stochastic, dynamic systems
- Optimizing energy storage
  - Using Bellman error minimization
  - Using policy search and optimal learning
- SMART-ISO – Robust unit commitment using a lookahead policy
- Observations

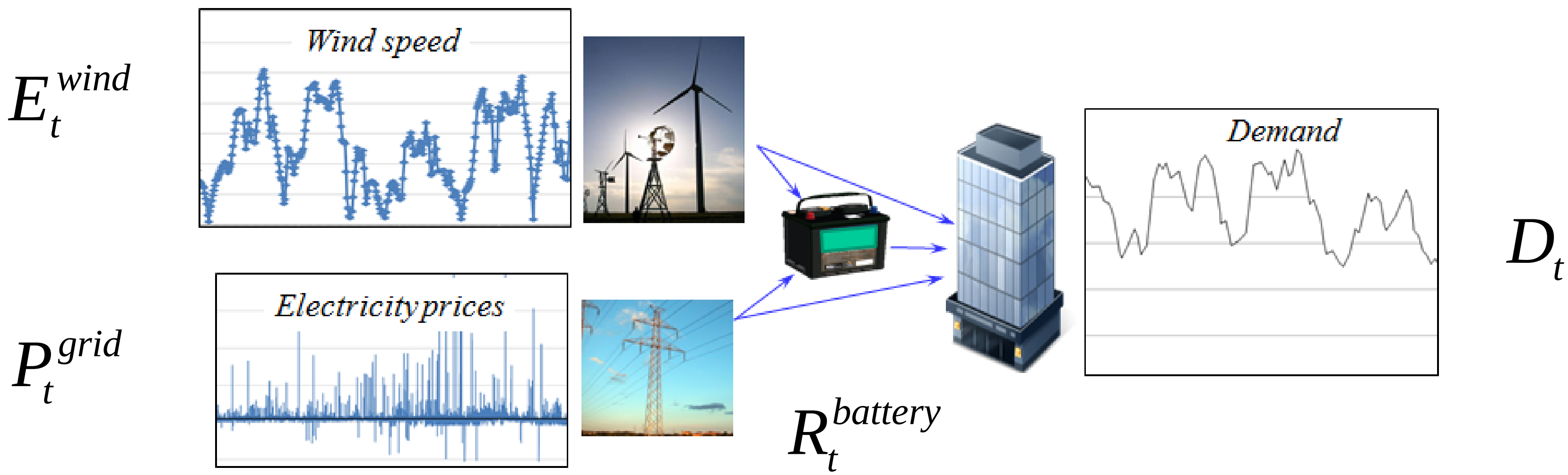
Optimizing energy storage

- ❑ Take advantage of price variations



Optimizing a renewable system

- Energy storage with stochastic prices, supplies and demands.



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

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

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

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

W_{t+1} = Exogenous inputs

S_t = State variable

x_t = Controllable inputs

Optimizing a renewable system

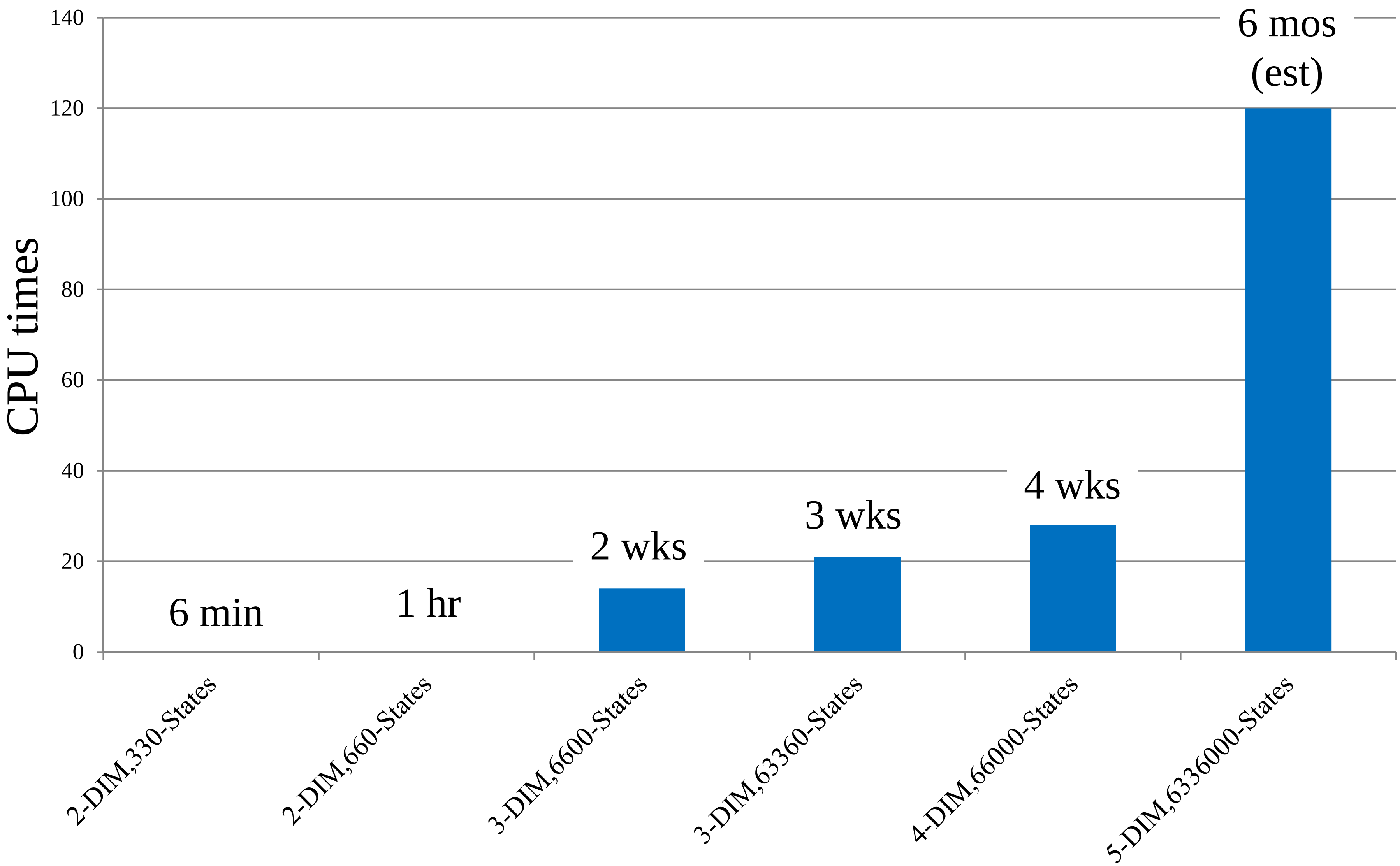
- Bellman's optimality equation

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

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

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

Value function approximation

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.

New information ("innovation")

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.

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:

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

to obtain x^m.

Step 2b: Update the value function approximation

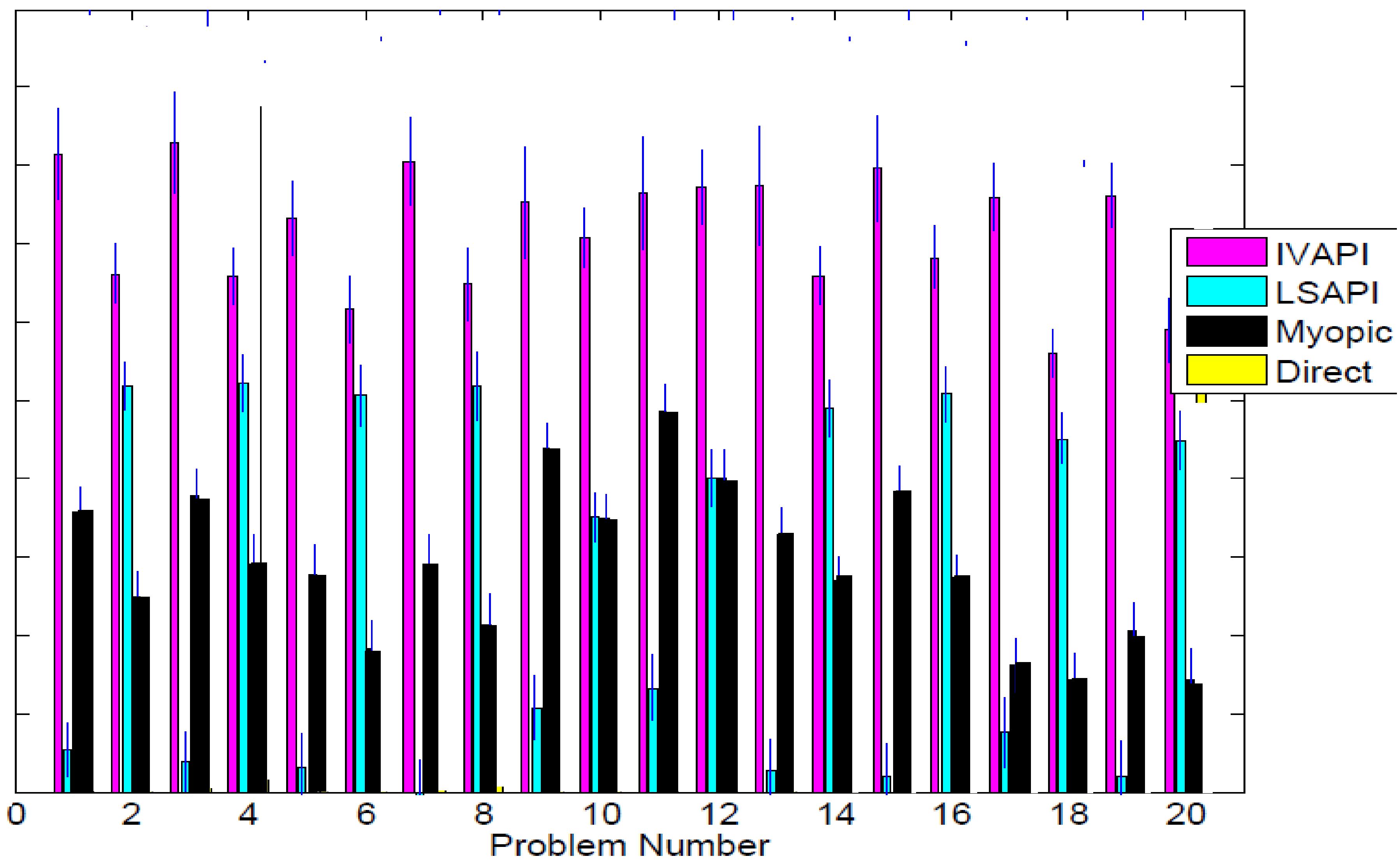
$$\bar{V}^{n-1,m}(S^{x,m}) = (1-\alpha_{m-1})\bar{V}^{n-1,m-1}(S^{x,m}) + \alpha_{m-1}\hat{v}^m$$

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

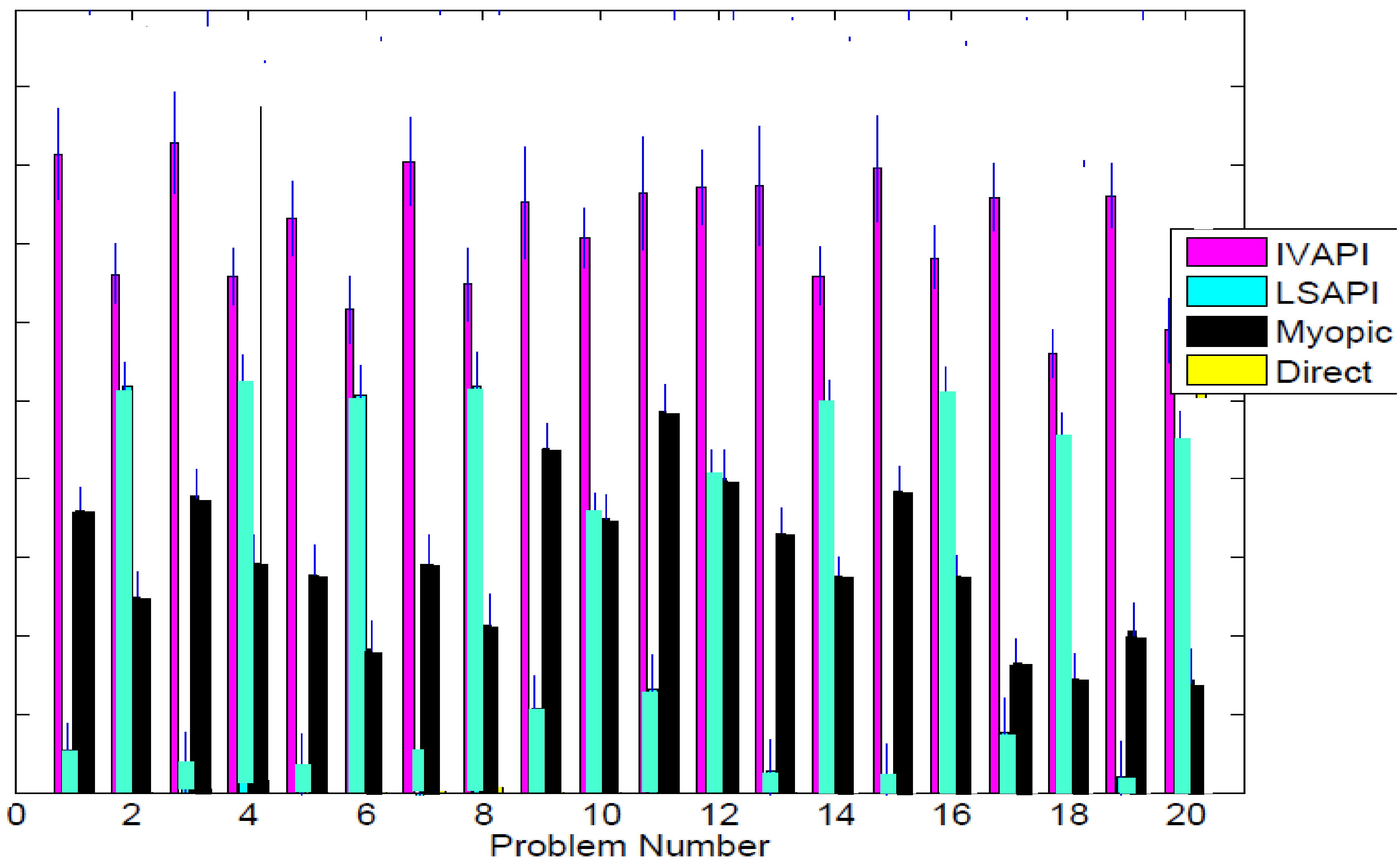
$$S^{m+1} = S^M(S^m, x^m, W(\omega^m))$$

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

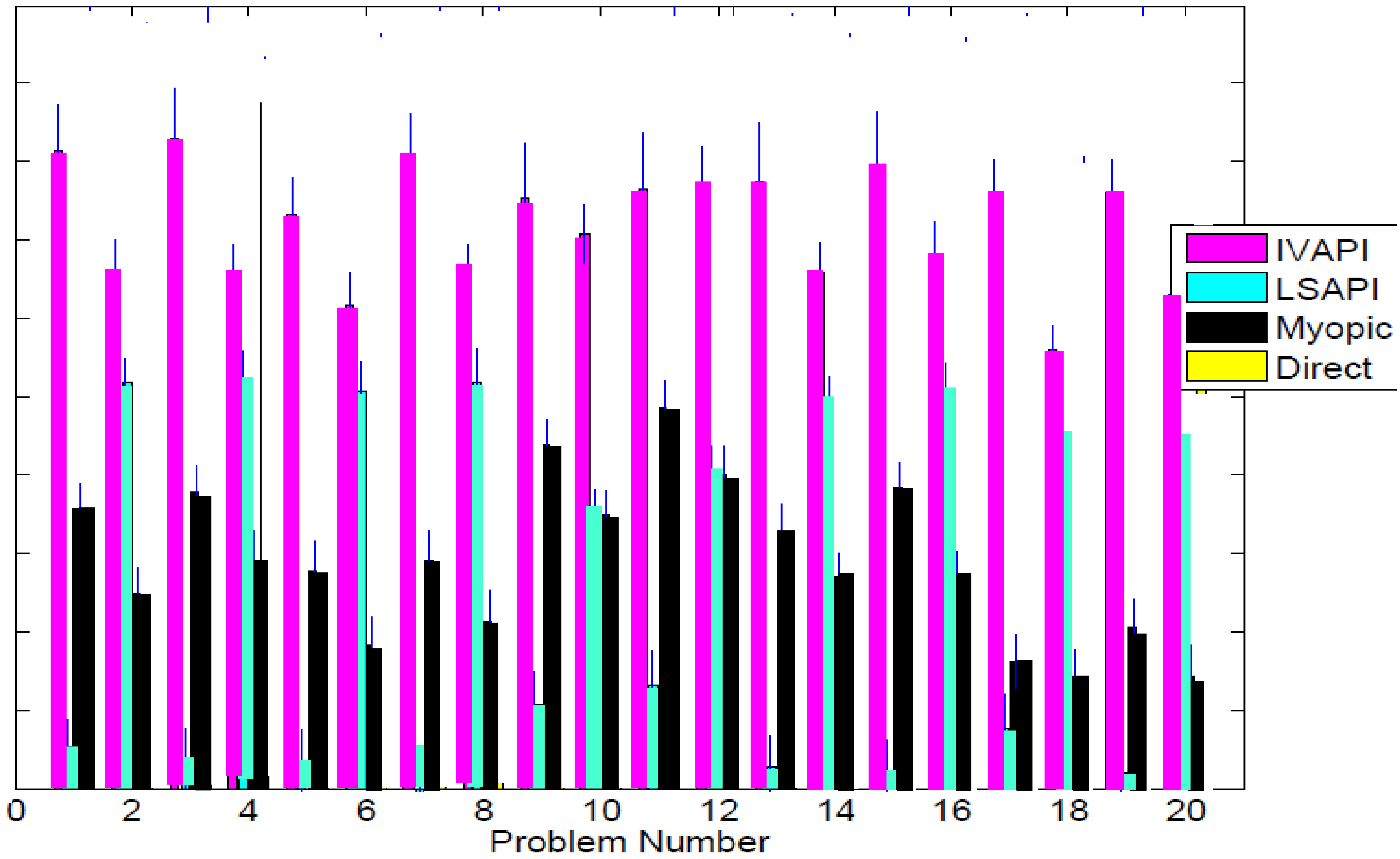
Optimizing storage policy



Optimizing storage policy



Optimizing storage policy

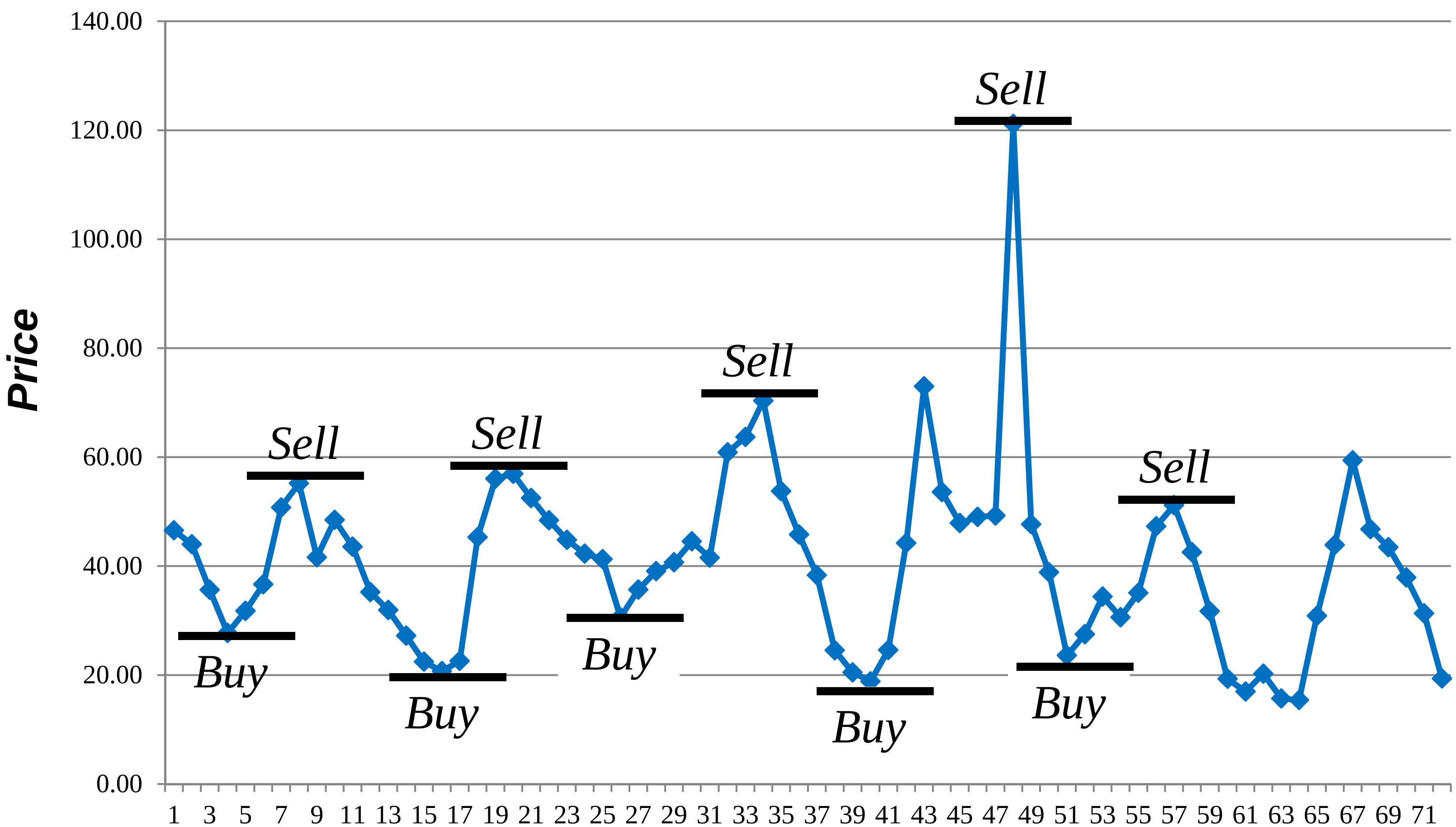


Lecture outline

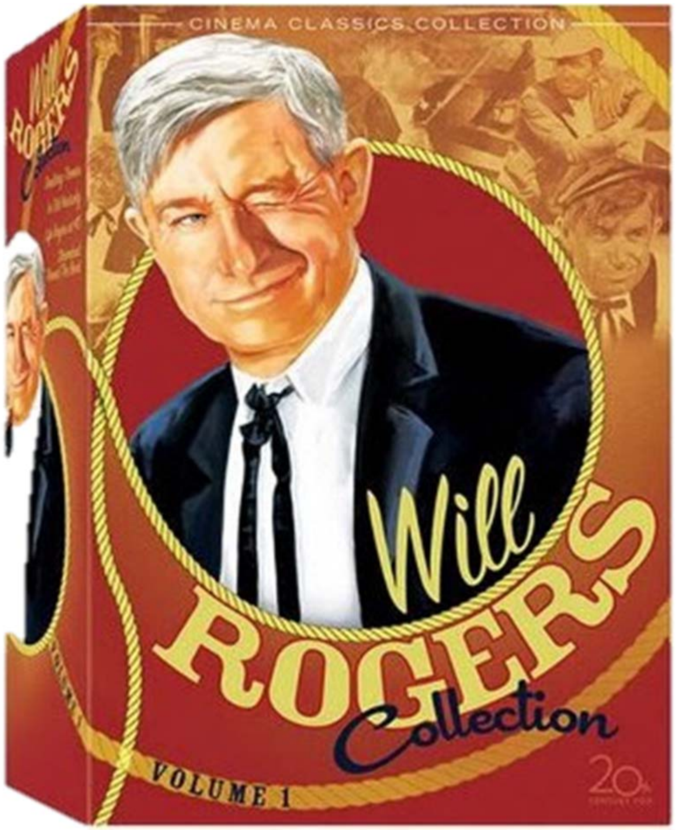
- Types of uncertainty
- Modeling stochastic, dynamic systems
- Optimizing energy storage
  - Using Bellman error minimization
  - Using policy search and optimal learning
- SMART-ISO – Robust unit commitment using a lookahead policy
- Observations

Optimizing energy storage

- The "buy low, sell high" policy



Optimizing energy storage



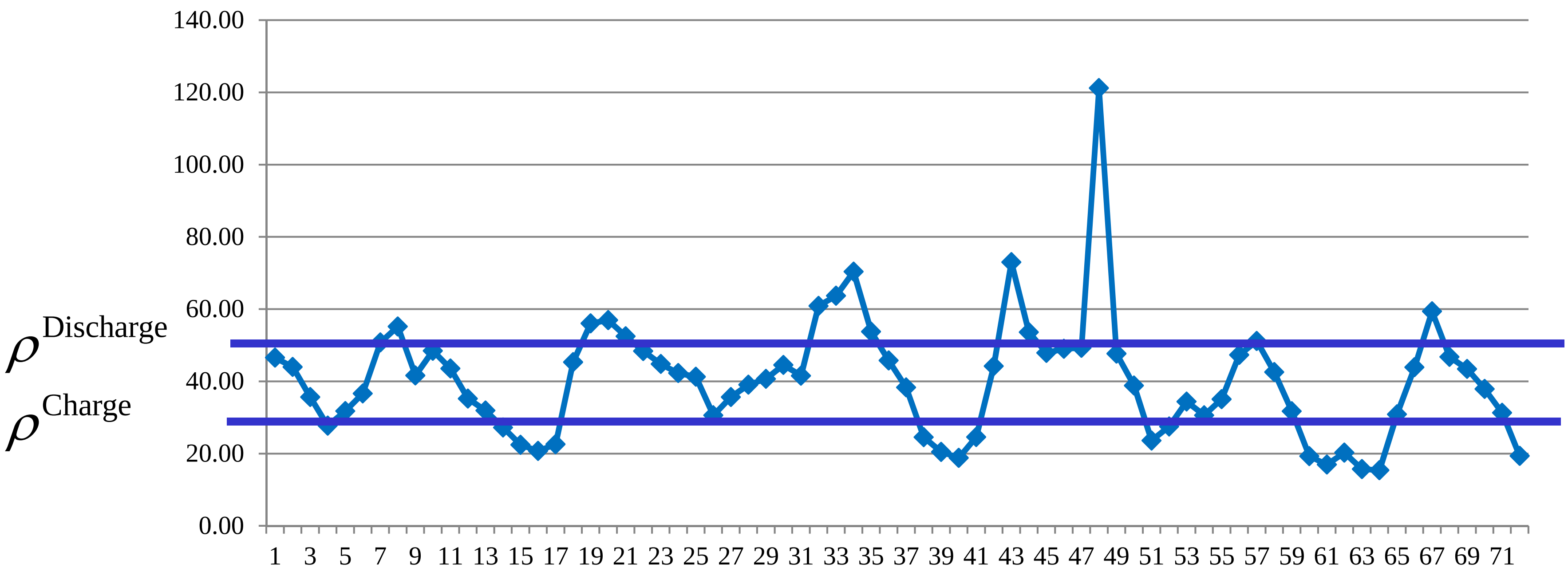
Don't gamble; take all your savings and buy some good stock and hold it till it goes up, then sell it. If it don't go up, don't buy it.

Will Rogers

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

Optimizing energy storage

- Grid operators require that batteries bid charge and discharge prices, an hour in advance.

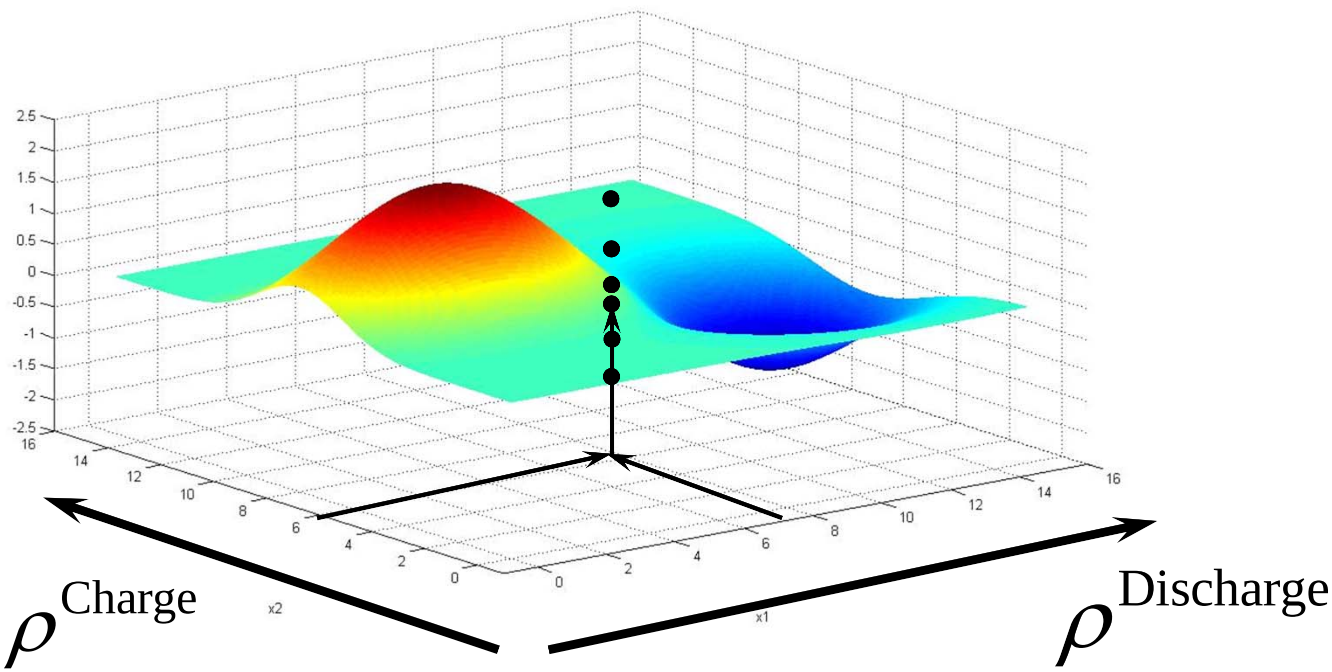


- We have to search for the best values for the policy parameters ρ^{Charge} and $\rho^{Discharge}$.

Optimizing energy storage

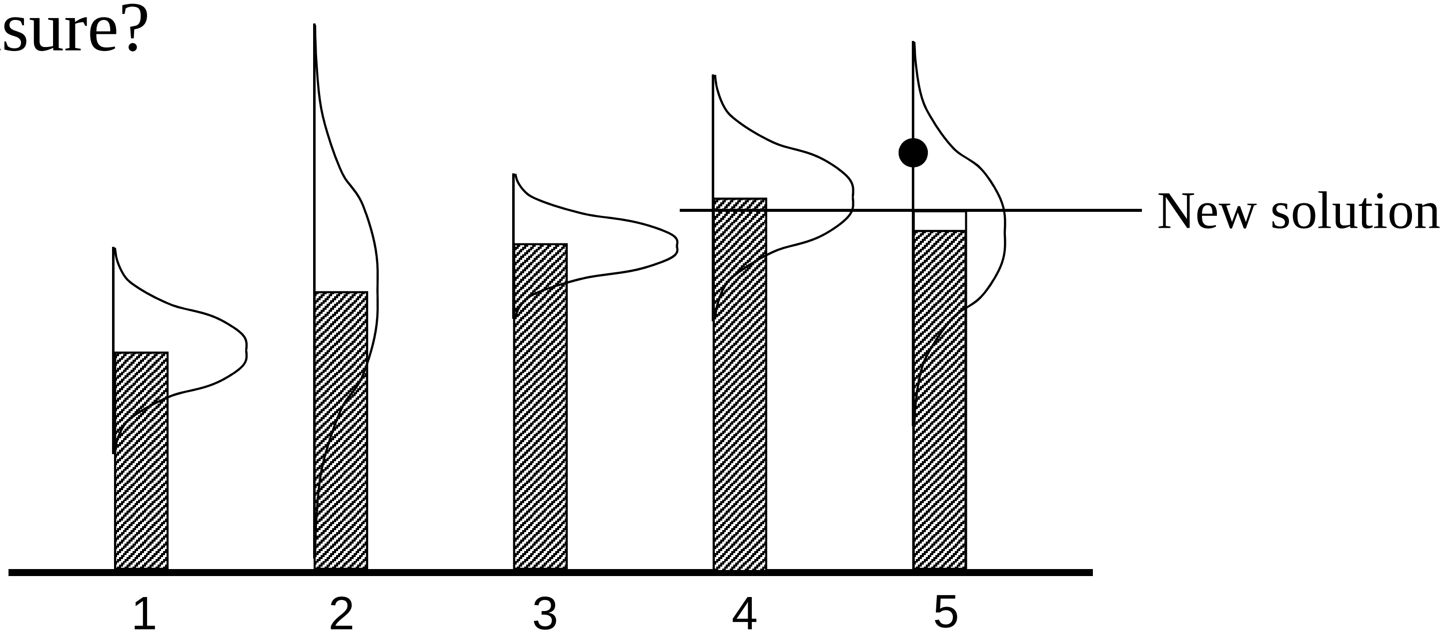
- We have to find the best policy
  - Let $X^{\pi}(S_t \mid \rho^{store}, \rho^{withdraw})$ be the "policy" that chooses the actions.
  - We wish to maximize the function

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



Optimal learning

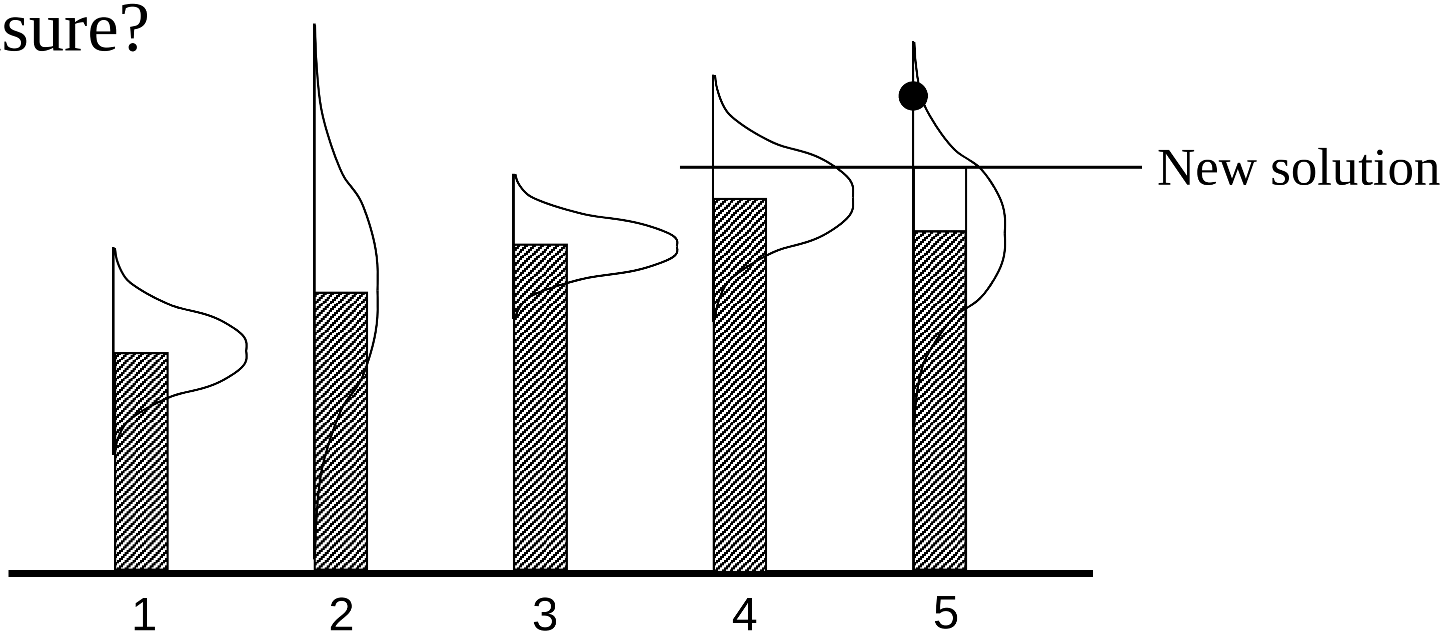
- Assume we have five choices of experiments, with uncertainty in our belief about how well each one will perform.
- If you can perform one experiment, which would you measure?



- The value of information is the expected improvement in a design as a result of an experiment.

Optimal learning

- Assume we have five choices of experiments, with uncertainty in our belief about how well each one will perform.
- If you can perform one experiment, which would you measure?



- The value of information is the expected improvement in a design as a result of an experiment.

Optimal learning

- The knowledge gradient for policy search
  - » We need to solve the classical stochastic search problem

$$\max_x \mathbb{E} F(x,W)$$

  - » We assume that $F(x,W)$ can only be simulated, and observations may be expensive.
  - » The knowledge gradient is the expected value of a single measurement *x*, given by

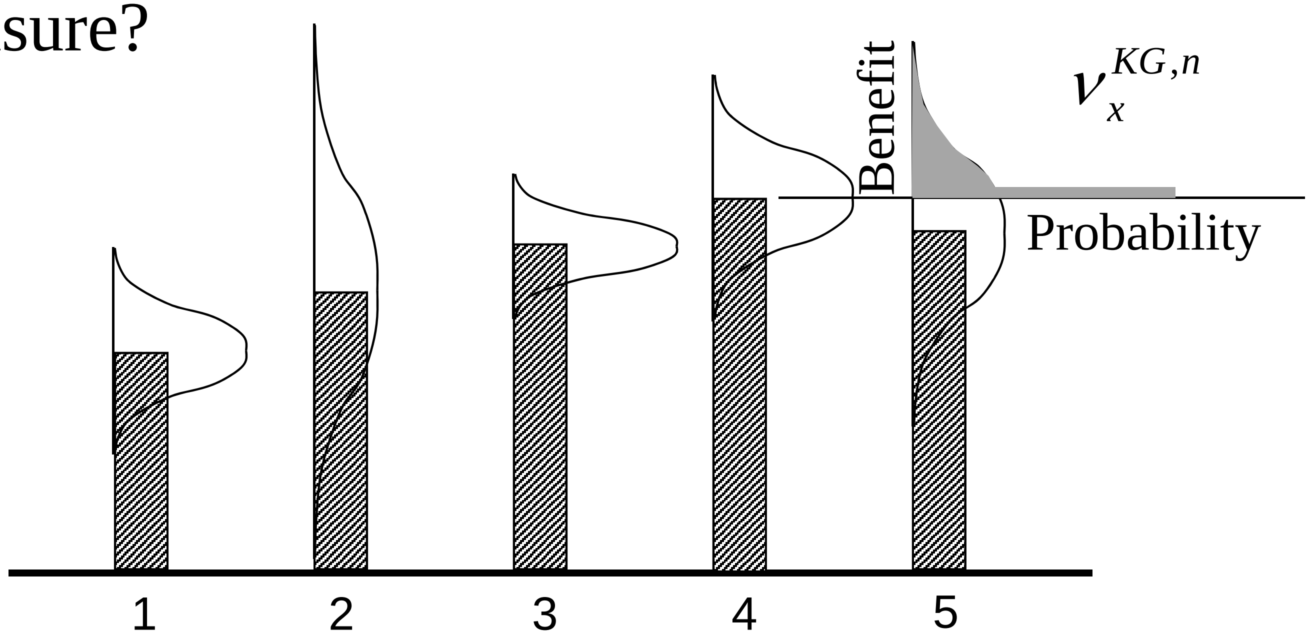
$$\nu_x^{KG,n} = E\left\{\max_y F(y,K^{n+1}(x))\right\} - \max_y F(y,K^n)$$

Marginal … Expectation … New optimization problem … Optimization problem given what we know … Knowledge state

  - » The knowledge gradient policy evaluates $F(x,W)$ at *x* with the largest value of $\nu_x^{KG,n}$.

Optimal learning

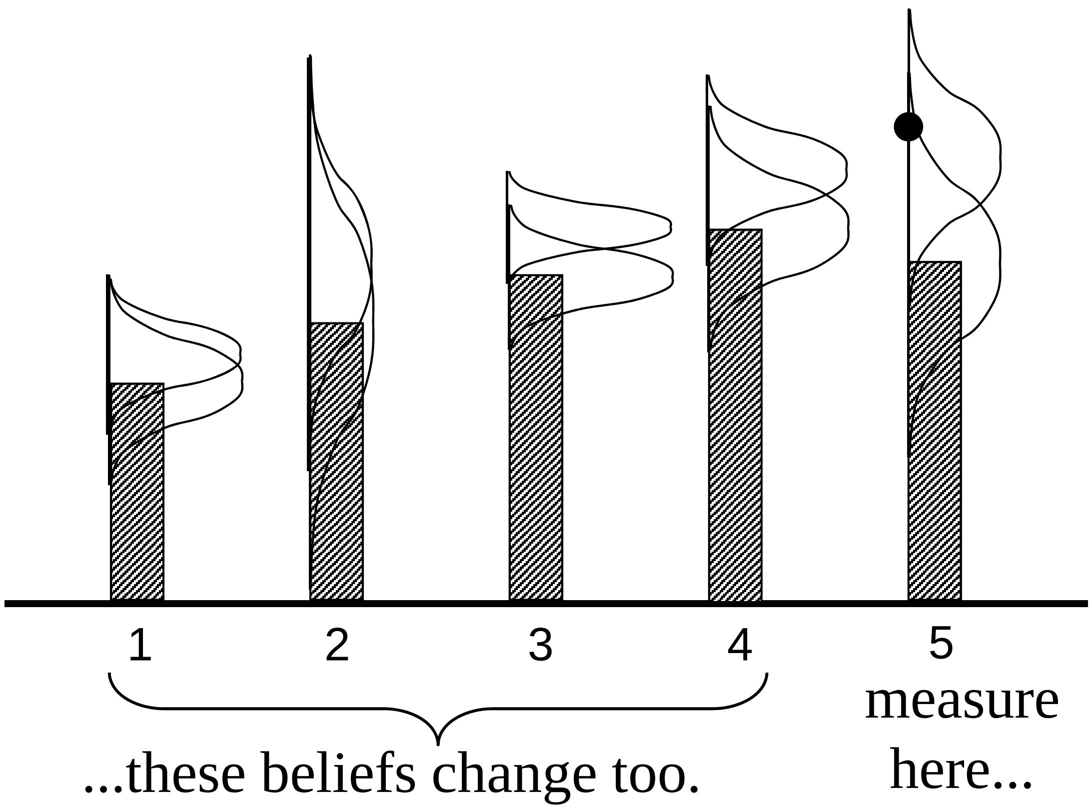
- Assume we have five choices of experiments, with uncertainty in our belief about how well each one will perform.
- If you can perform one experiment, which would you measure?



- The value of information is the expected improvement in a design as a result of an experiment, which requires striking a balance between potential performance and uncertainty.

Optimal learning

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

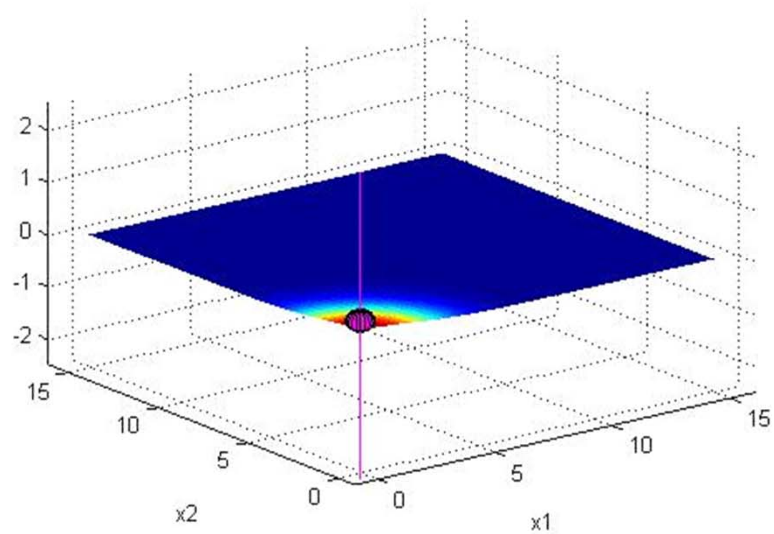


- Correlated beliefs allow us to dramatically reduce the number of experiments that need to be run.

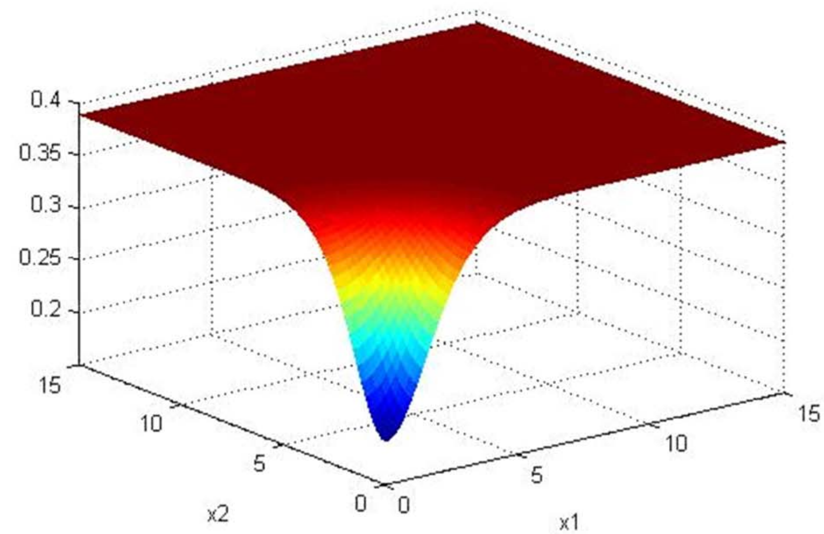
Optimizing storage policy

- Initially we think the concentration is the same everywhere:

Estimated profit



Knowledge gradient

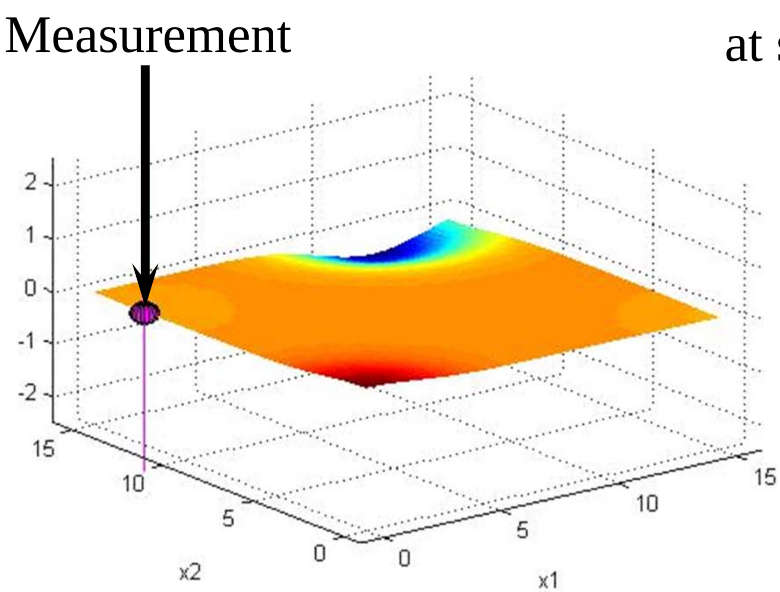


- We want to measure the value where the knowledge gradient is the highest. This is the measurement that teaches us the most.

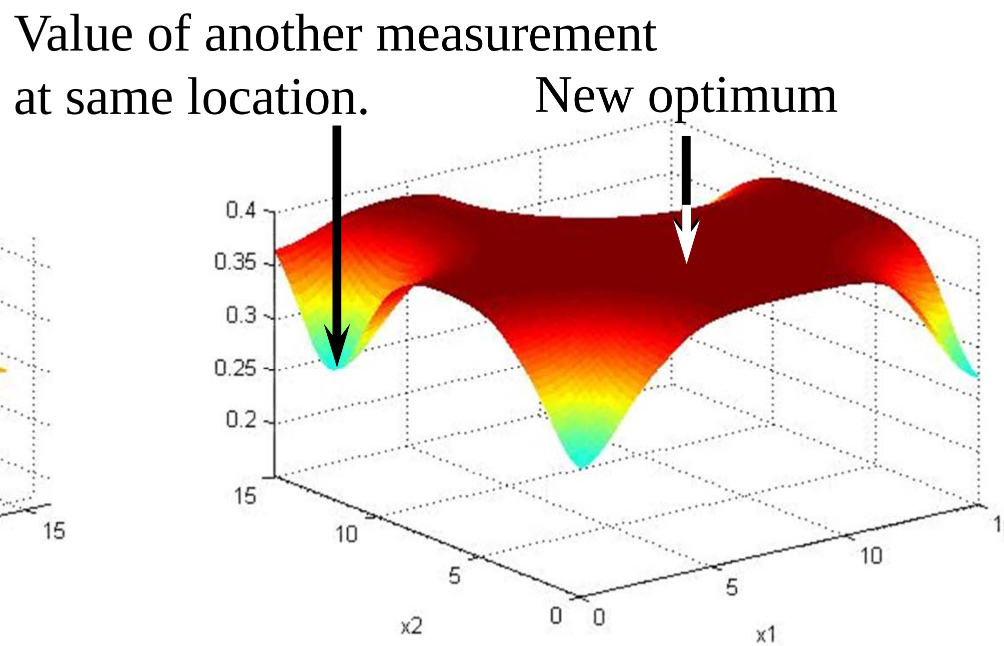
Optimizing storage policy

- After four measurements:

Estimated value



Knowledge gradient



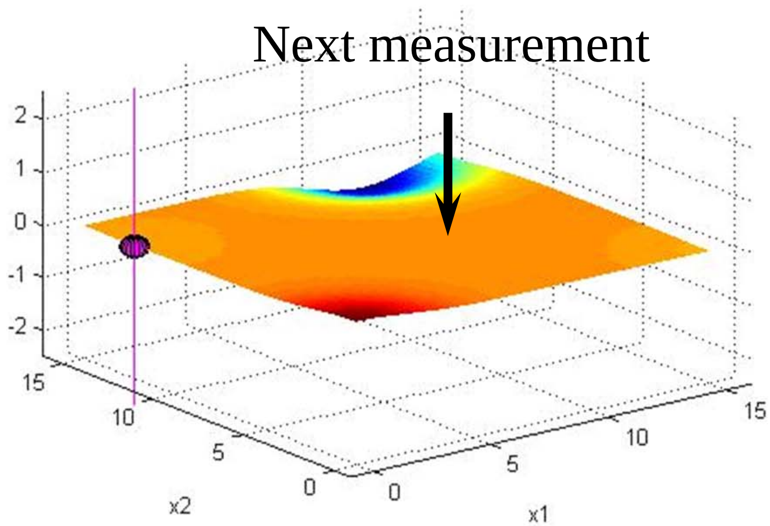
» Whenever we measure at a point, the value of another measurement at the same point goes down. The knowledge gradient guides us to measuring areas of high uncertainty.

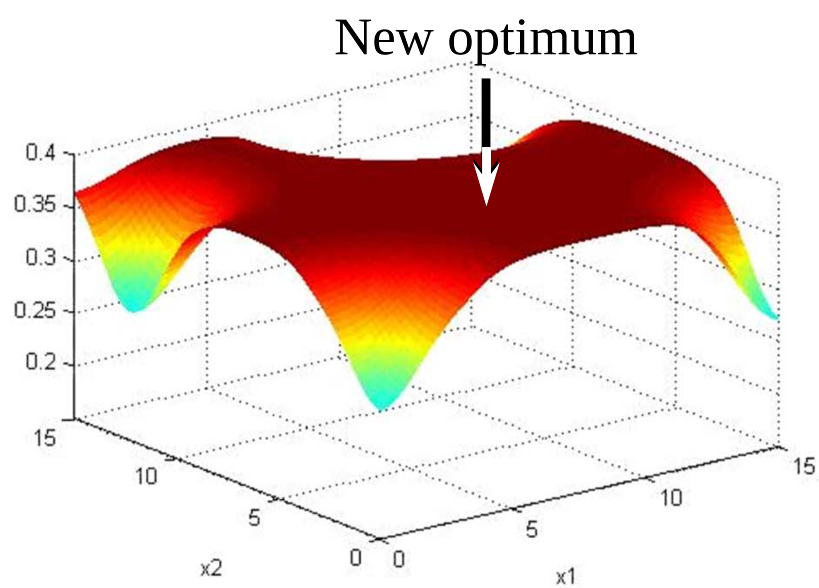
Optimizing storage policy

- After four measurements:

Estimated value

Knowledge gradient



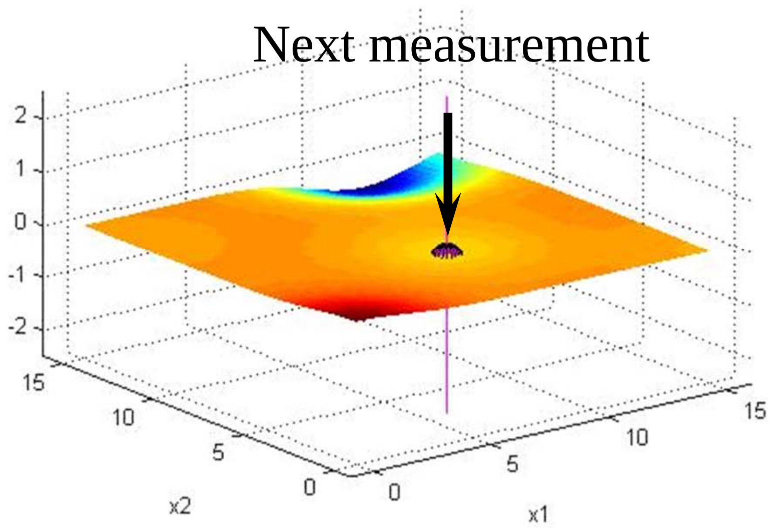


» Whenever we measure at a point, the value of another measurement at the same point goes down. The knowledge gradient guides us to measuring areas of high uncertainty.

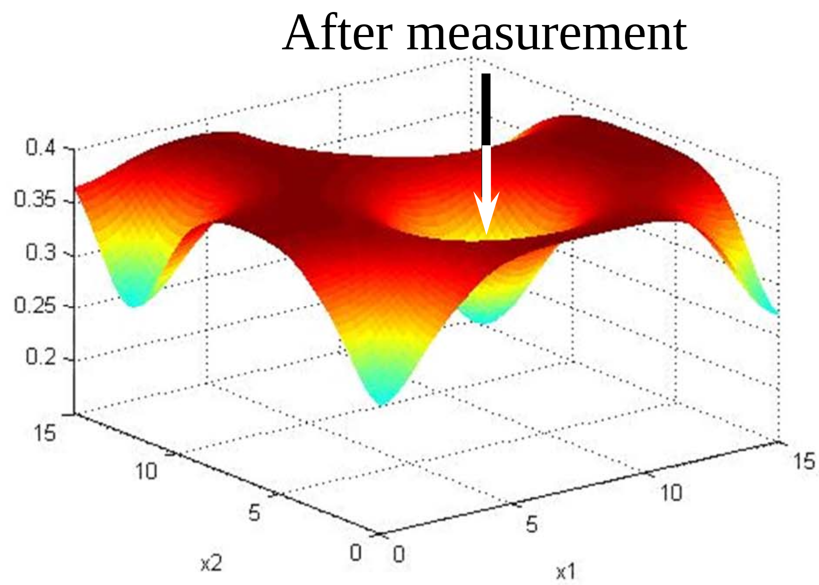
Optimizing storage policy

- After five measurements:

Estimated value



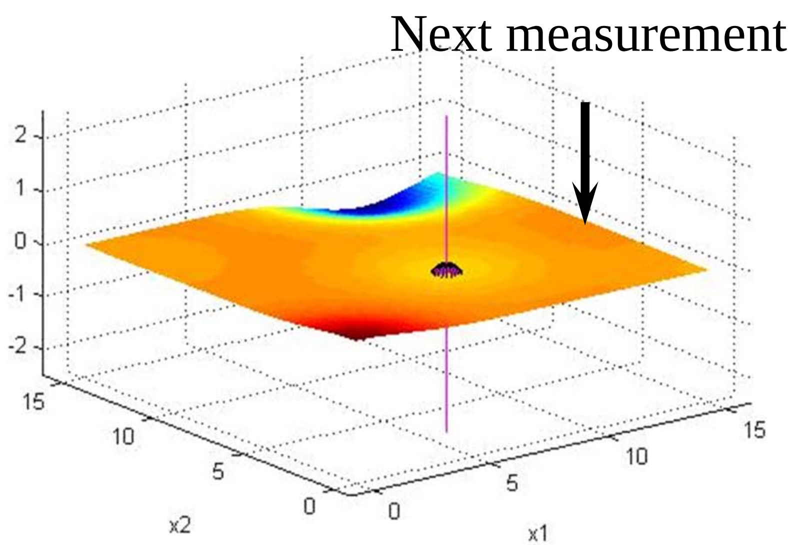
Knowledge gradient



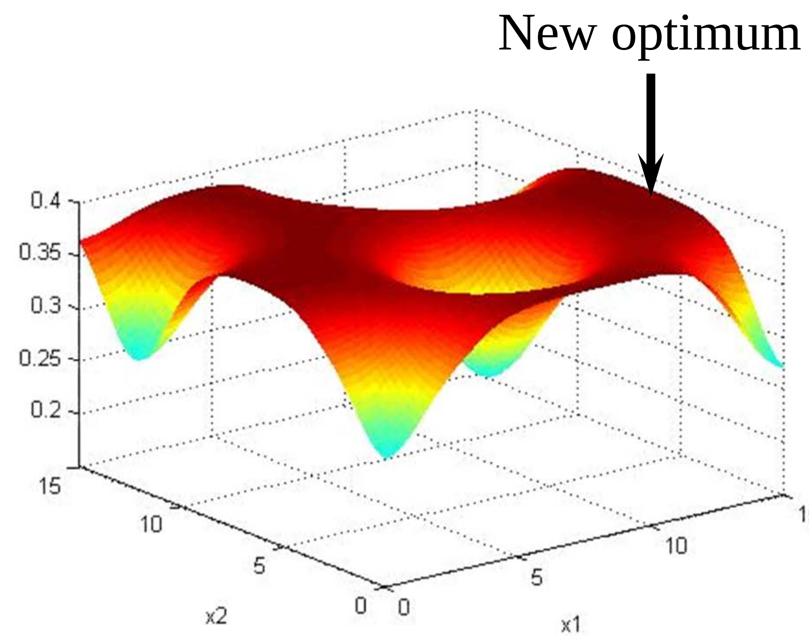
Optimizing storage policy

- After five measurements:

Estimated value



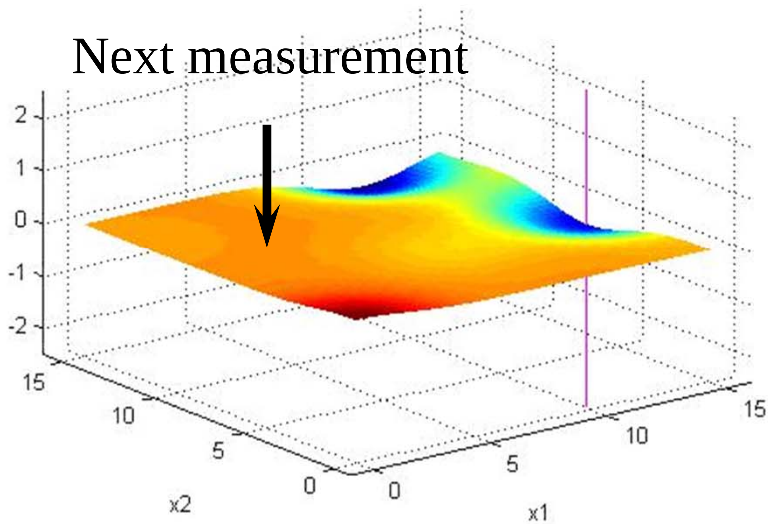
Knowledge gradient



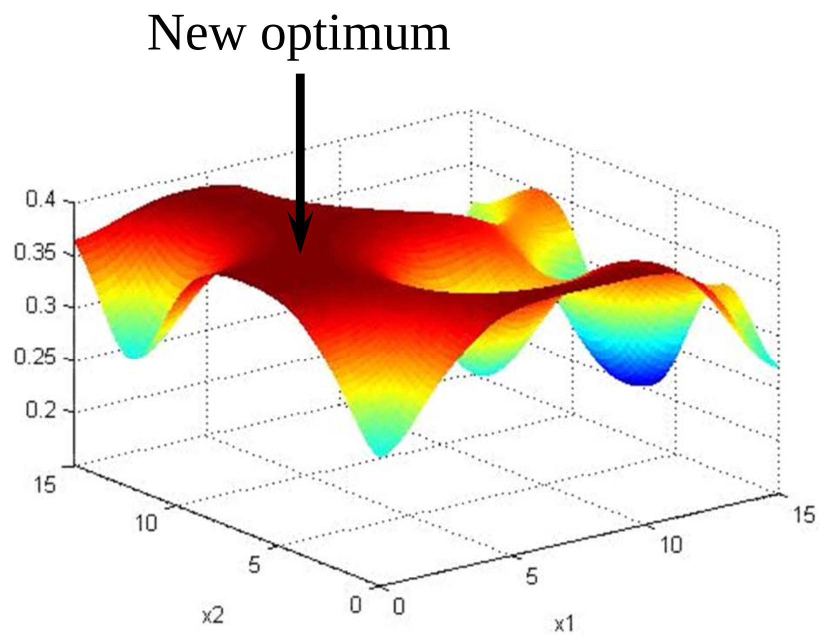
Optimizing storage policy

- After six samples

Estimated value



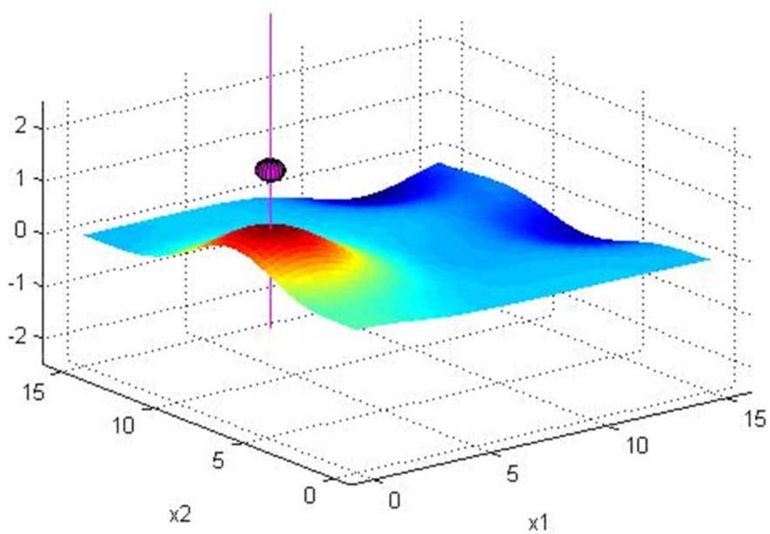
Knowledge gradient



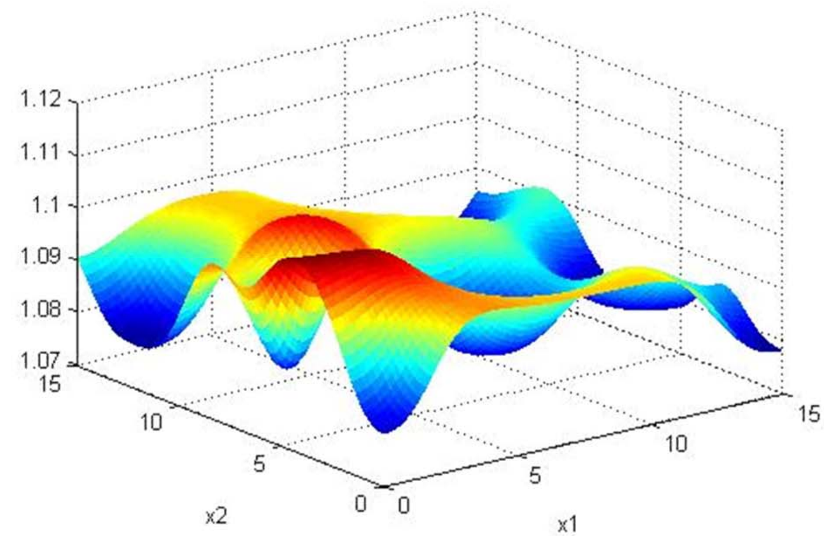
Optimizing storage policy

- After seven samples

Estimated value



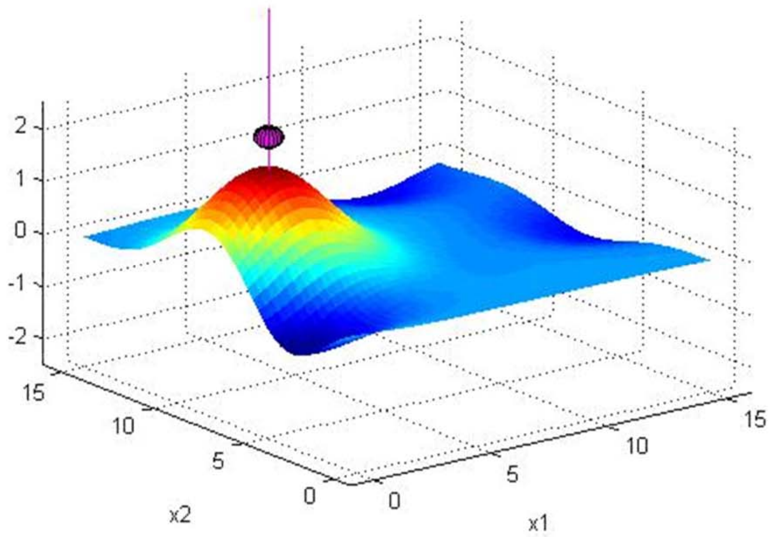
Knowledge gradient



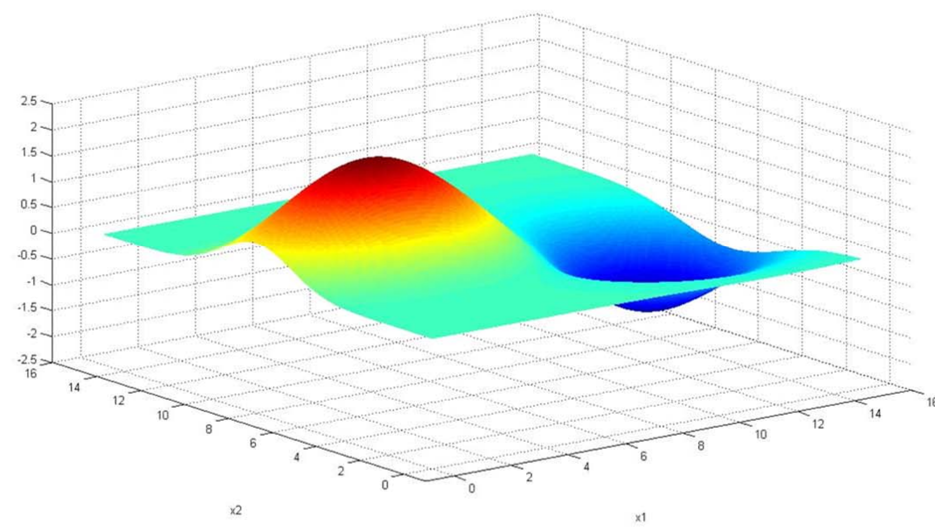
Optimizing storage policy

- After ten samples, our estimate of the surface:

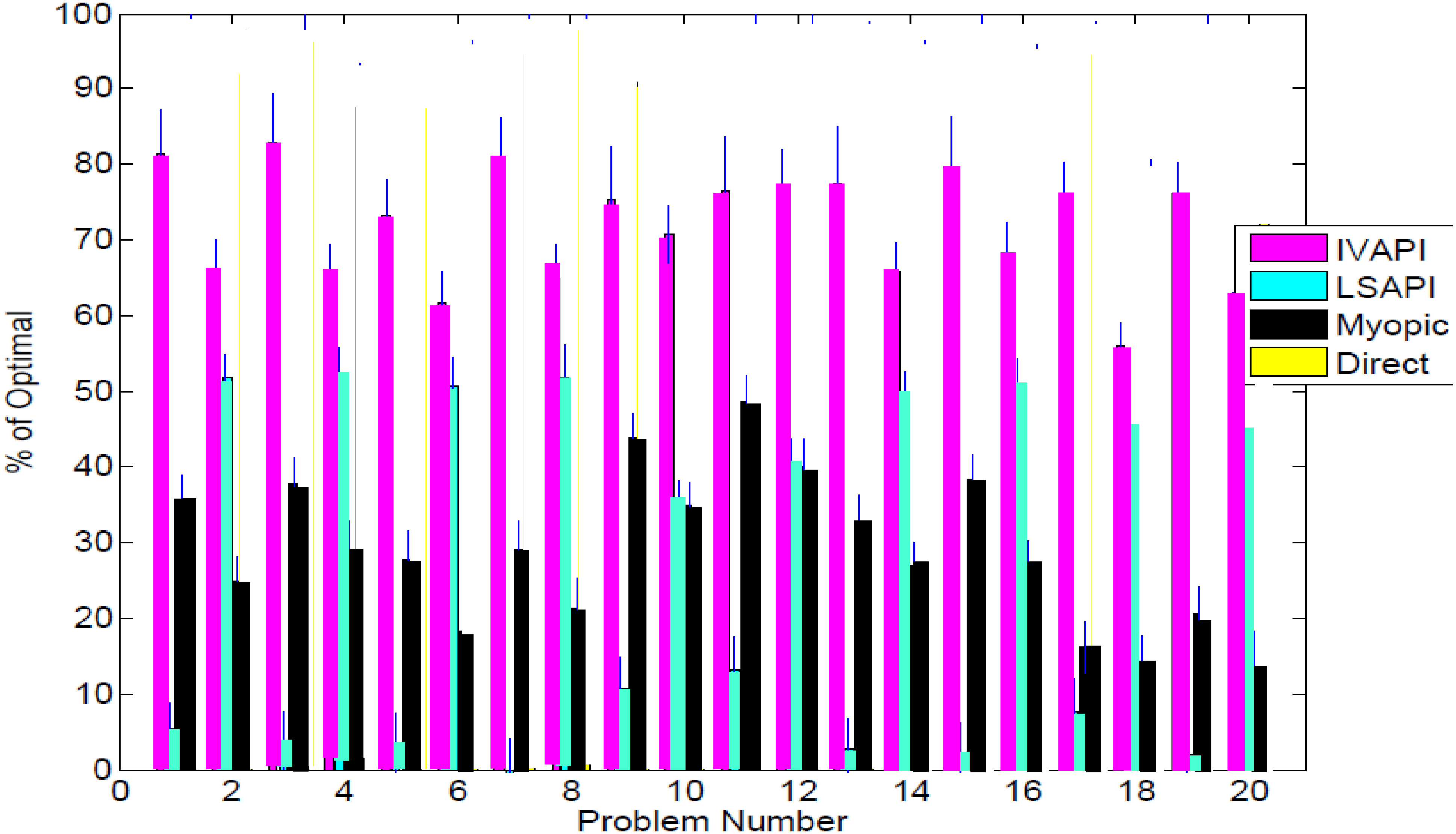
Estimated value



True value

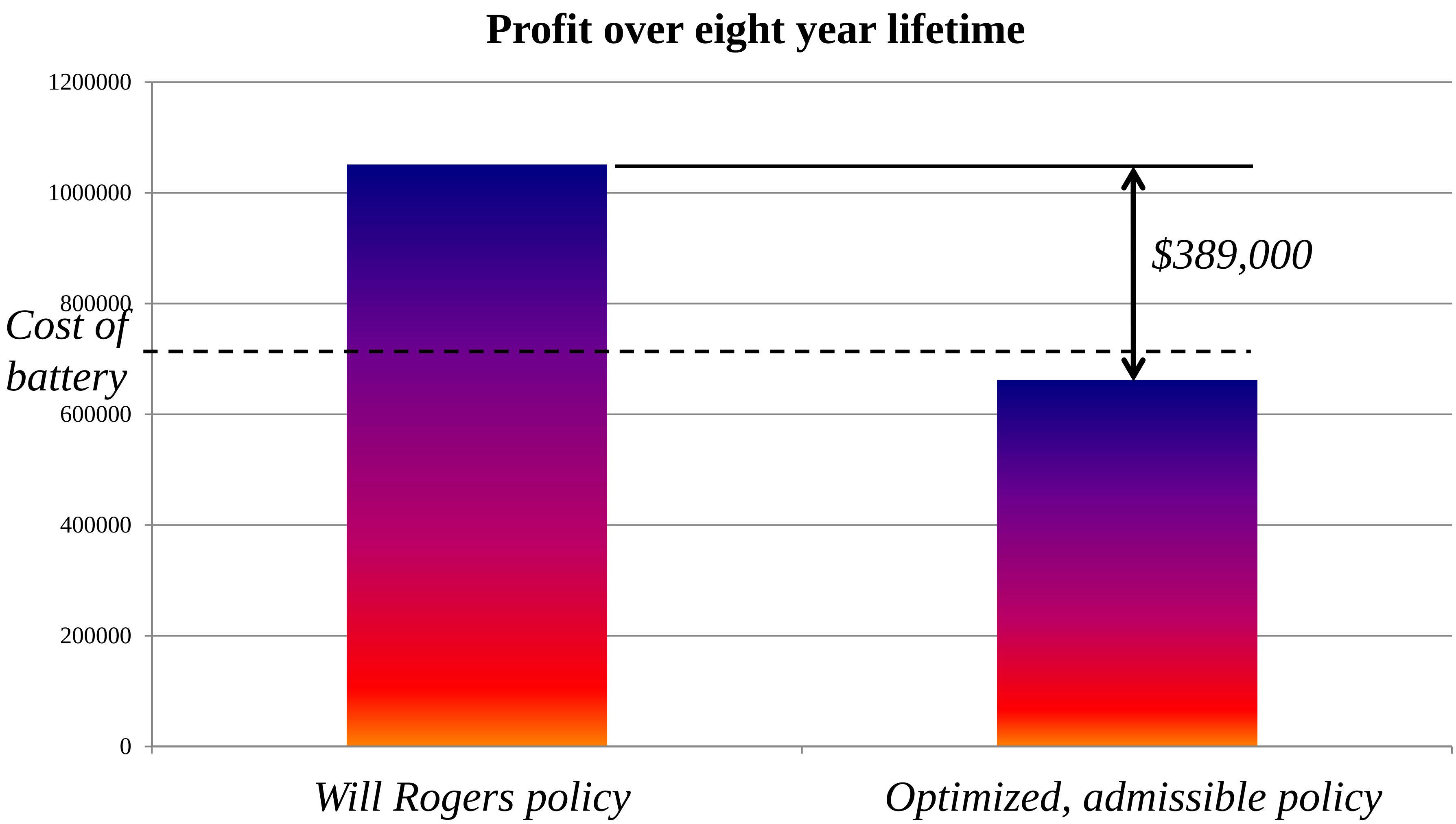


Optimizing storage policy



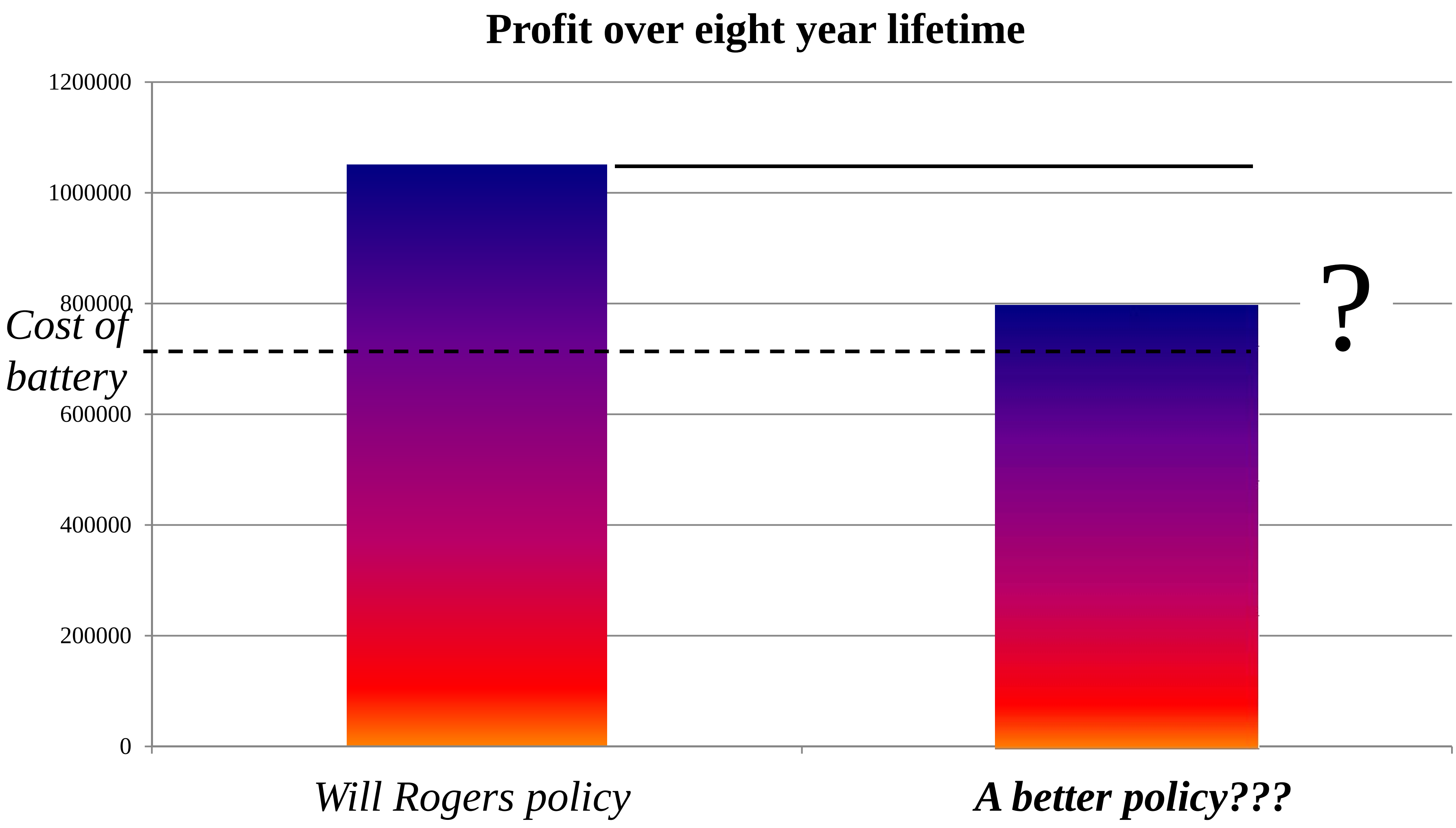
Optimizing storage policy

- The value of perfect information



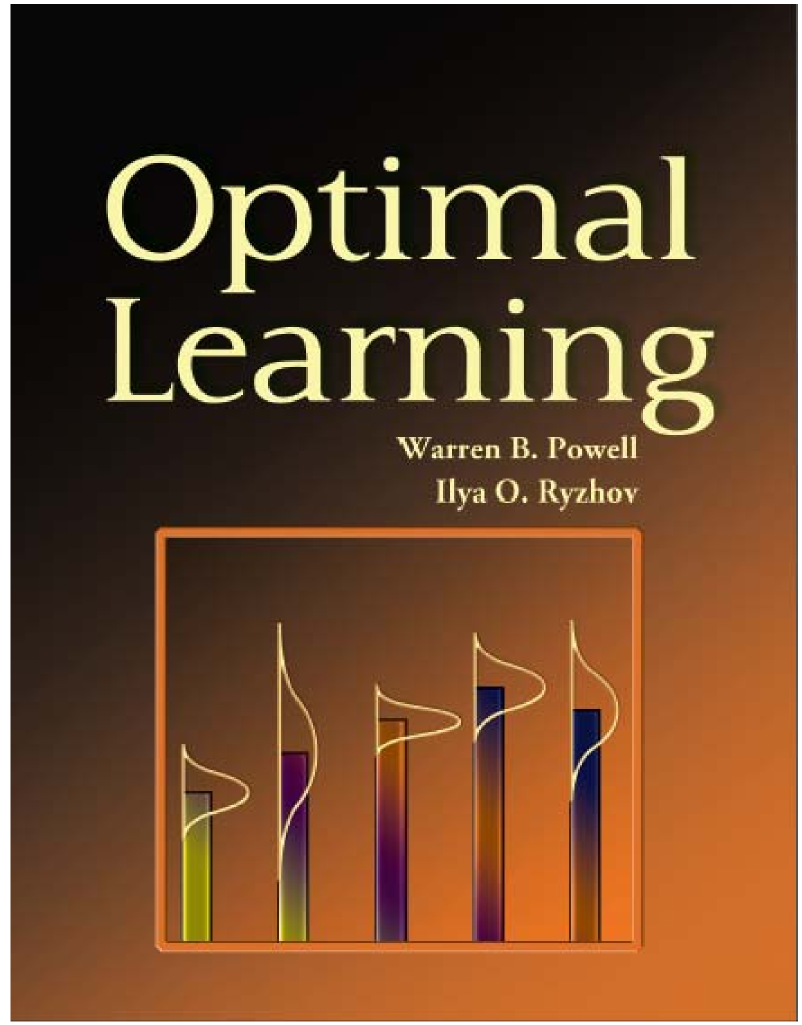
Optimizing storage policy

- The value of perfect information



New book!

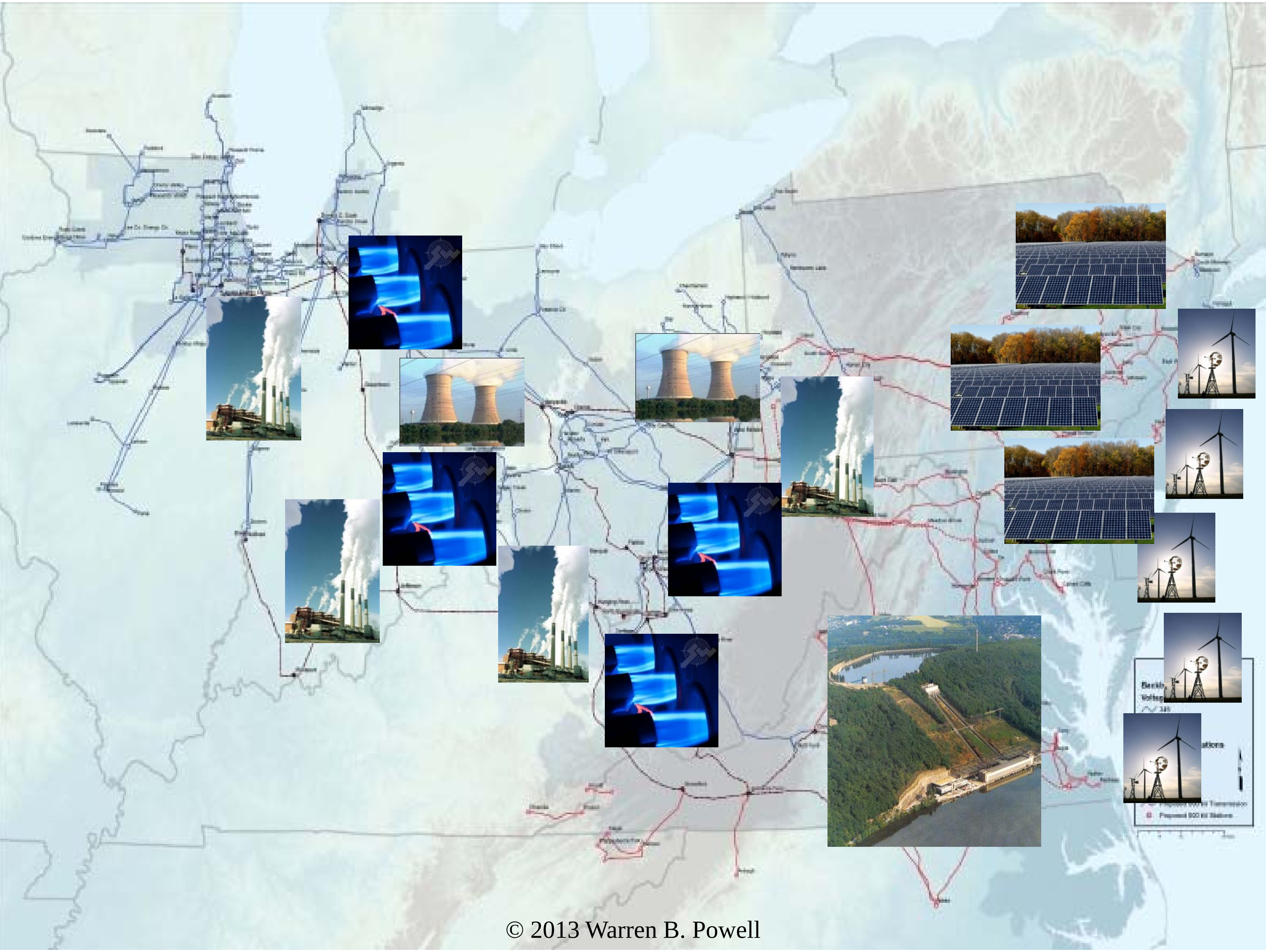
- New book on *Optimal Learning*
  - » Published by John Wiley in 2012.
  - » First 12 chapters are at an advanced undergraduate level.
  - » Funded by AFOSR
- Synthesizes communities:
  - » Ranking and selection
  - » Multiarmed bandits
  - » Stochastic search
  - » Simulation optimization
  - » Global optimization
  - » Experimental design



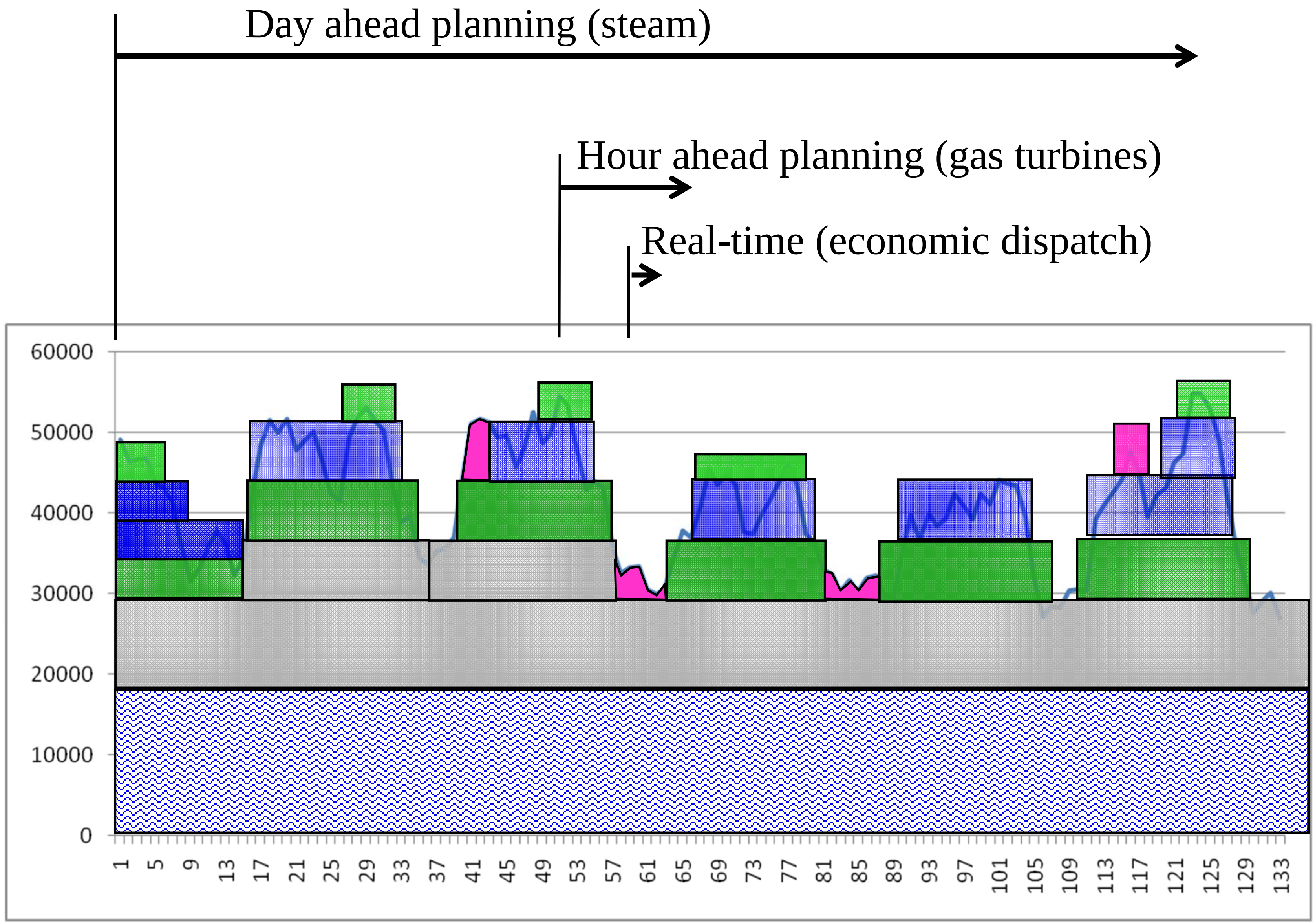
http://optimallearning.princeton.edu/

Lecture outline

- Types of uncertainty
- Modeling stochastic, dynamic systems
- Optimizing energy storage
  - Using Bellman error minimization
  - Using policy search and optimal learning
- SMART-ISO – Robust unit commitment using a lookahead policy
- Observations

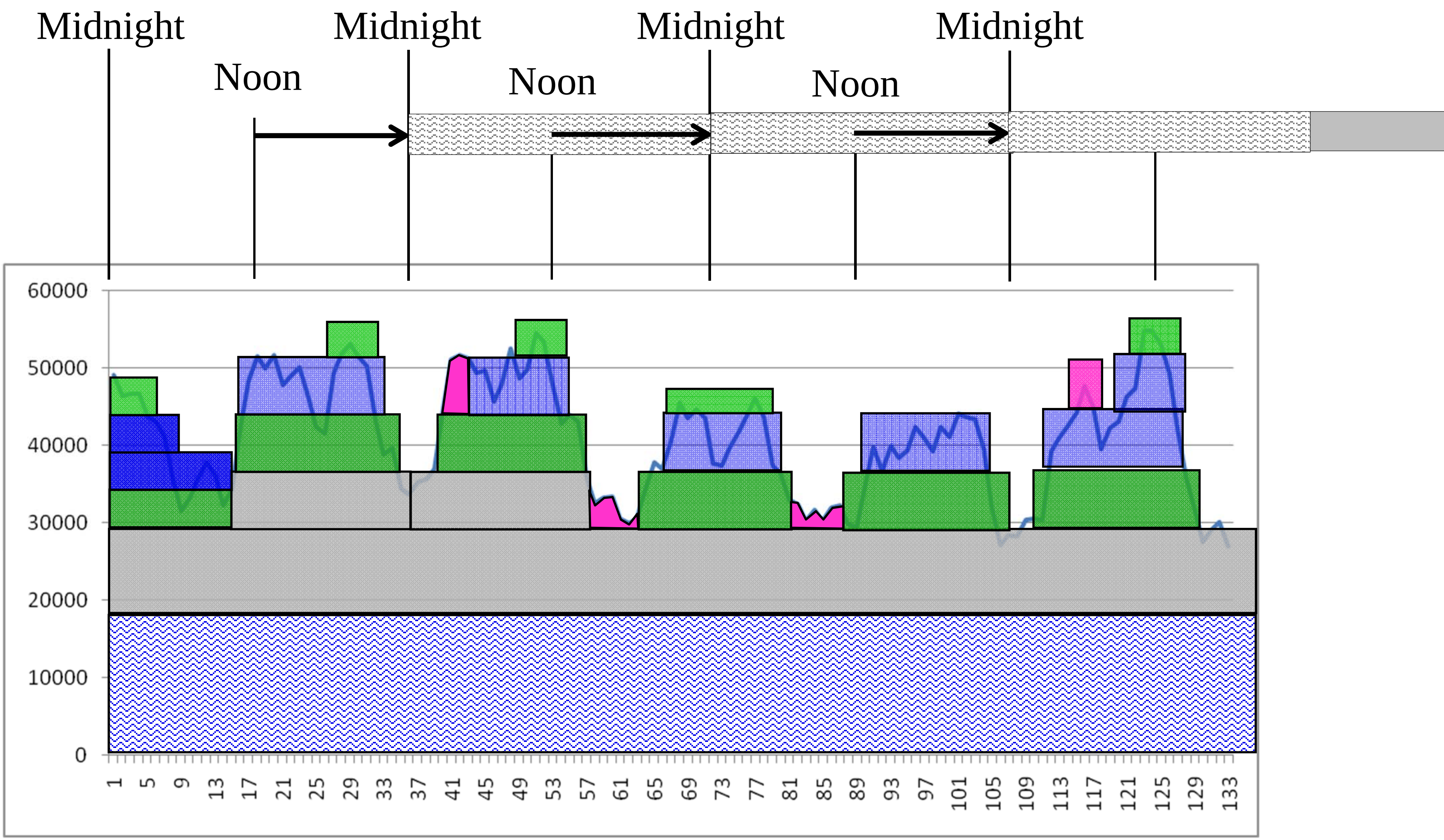


The stochastic unit commitment problem



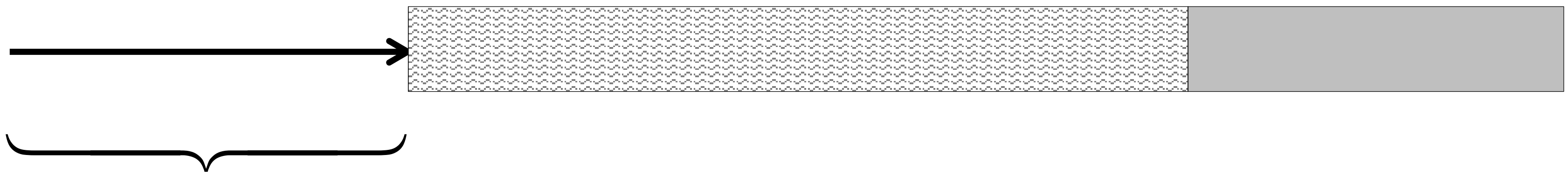
The timing of decisions

- The day-ahead problem



The timing of decisions

- The day-ahead problem

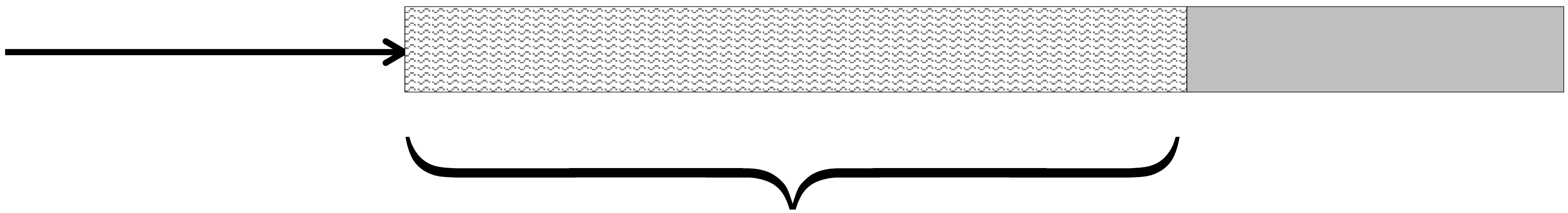


Noon to midnight:

Steam on/off decisions determined the day before
Optimize within spinning reserve margins
Optimize on/off operation of gas turbines

The timing of decisions

- The day-ahead problem



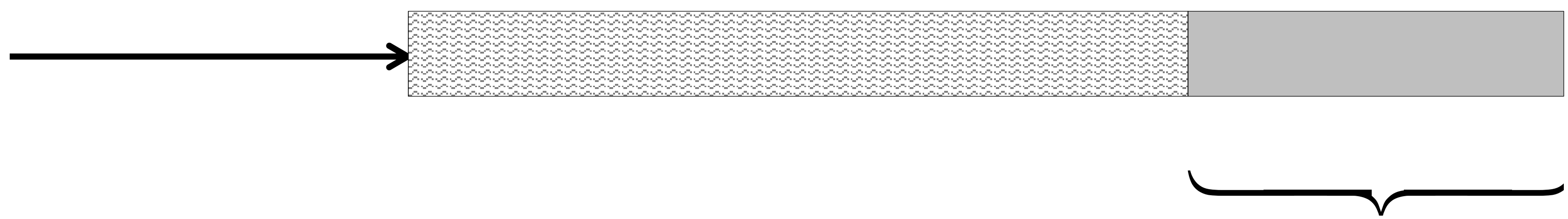
Midnight to midnight:

Optimize steam on/off decisions
Optimize within spinning reserve margins
Optimize on/off operation of gas turbines
Constrained by aggregate DC power flow

Steam on/off decisions are stored and implemented

The timing of decisions

- The day-ahead problem



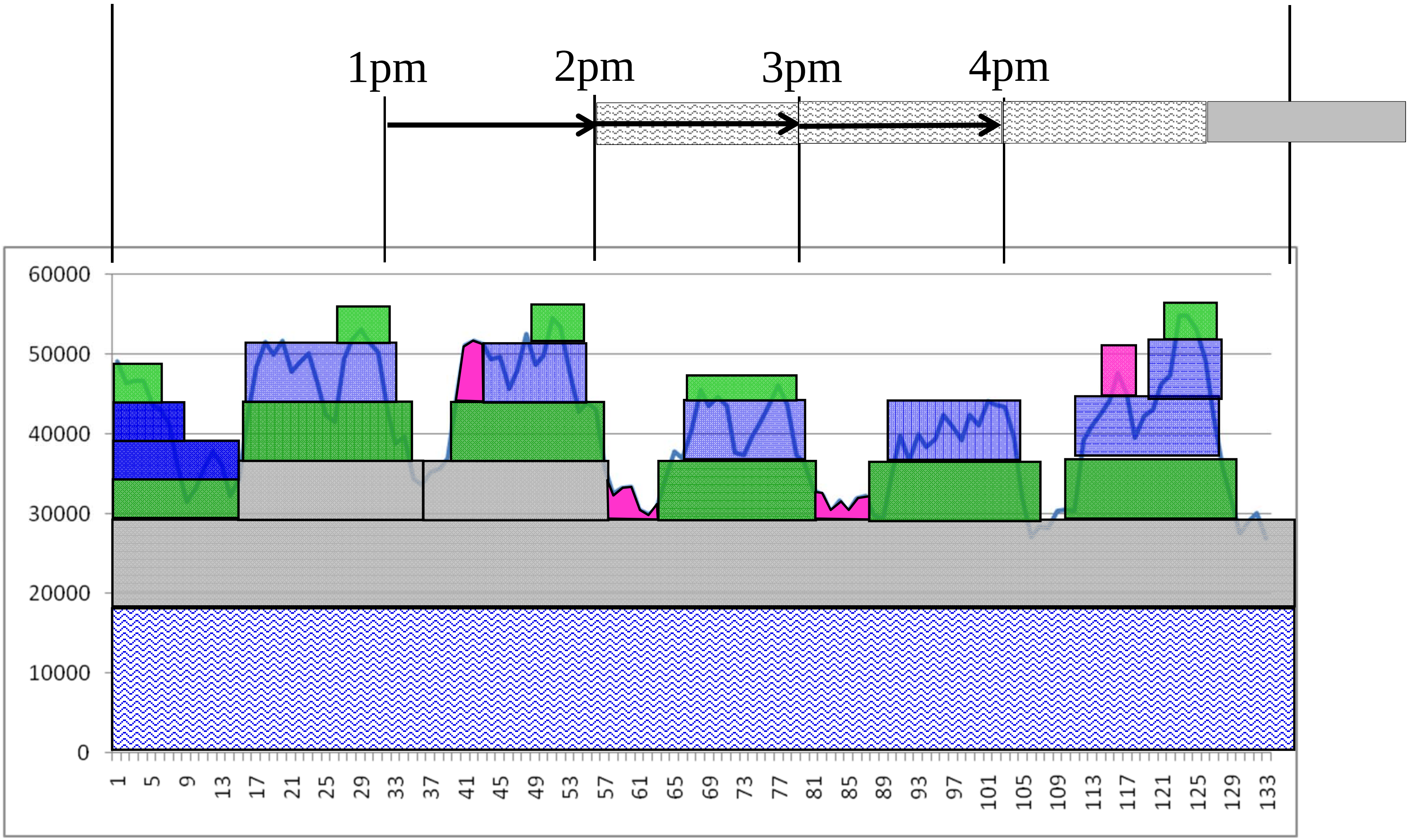
Midnight to noon the next day

Optimize steam on/off decisions
Optimize within spinning reserve margins
Optimize on/off operation of gas turbines
Constrained by aggregate DC power flow

No decisions are implemented. These are solved only to minimize end-of-day truncation error.

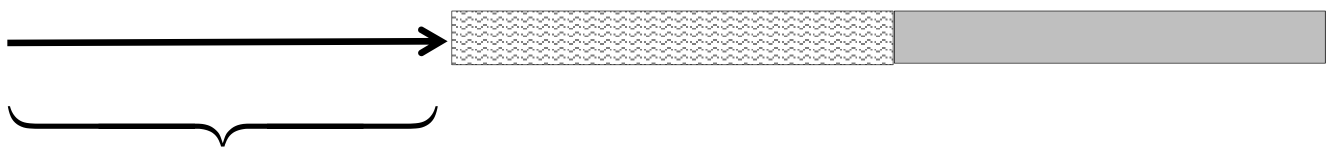
The timing of decisions

- The hour-ahead problem



The timing of decisions

- The hour-ahead problem



1pm to 2pm:

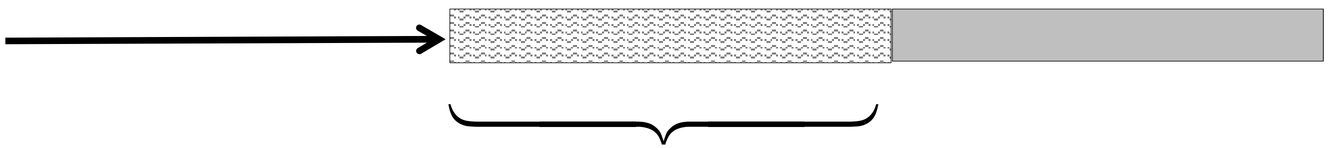
Steam on/off decisions determined the day before

Optimize within spinning reserve margins

On/off operation of gas turbines determined the hour before

The timing of decisions

- The hour-ahead problem



2pm to 3pm:

Steam on/off decisions determined the day before

Optimize within spinning reserve margins

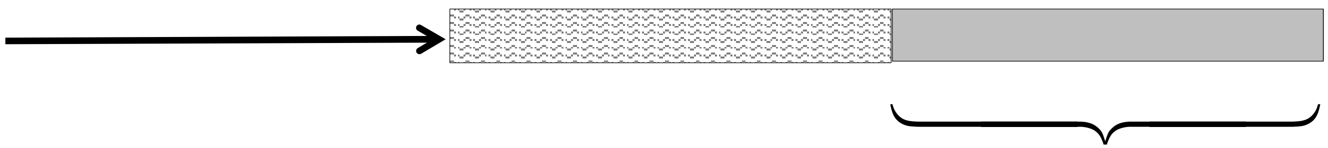
Optimize on/off operation of gas turbines

Constrained by aggregate DC power flow

On/off decisions for gas turbines are stored and implemented

The timing of decisions

- The hour-ahead problem



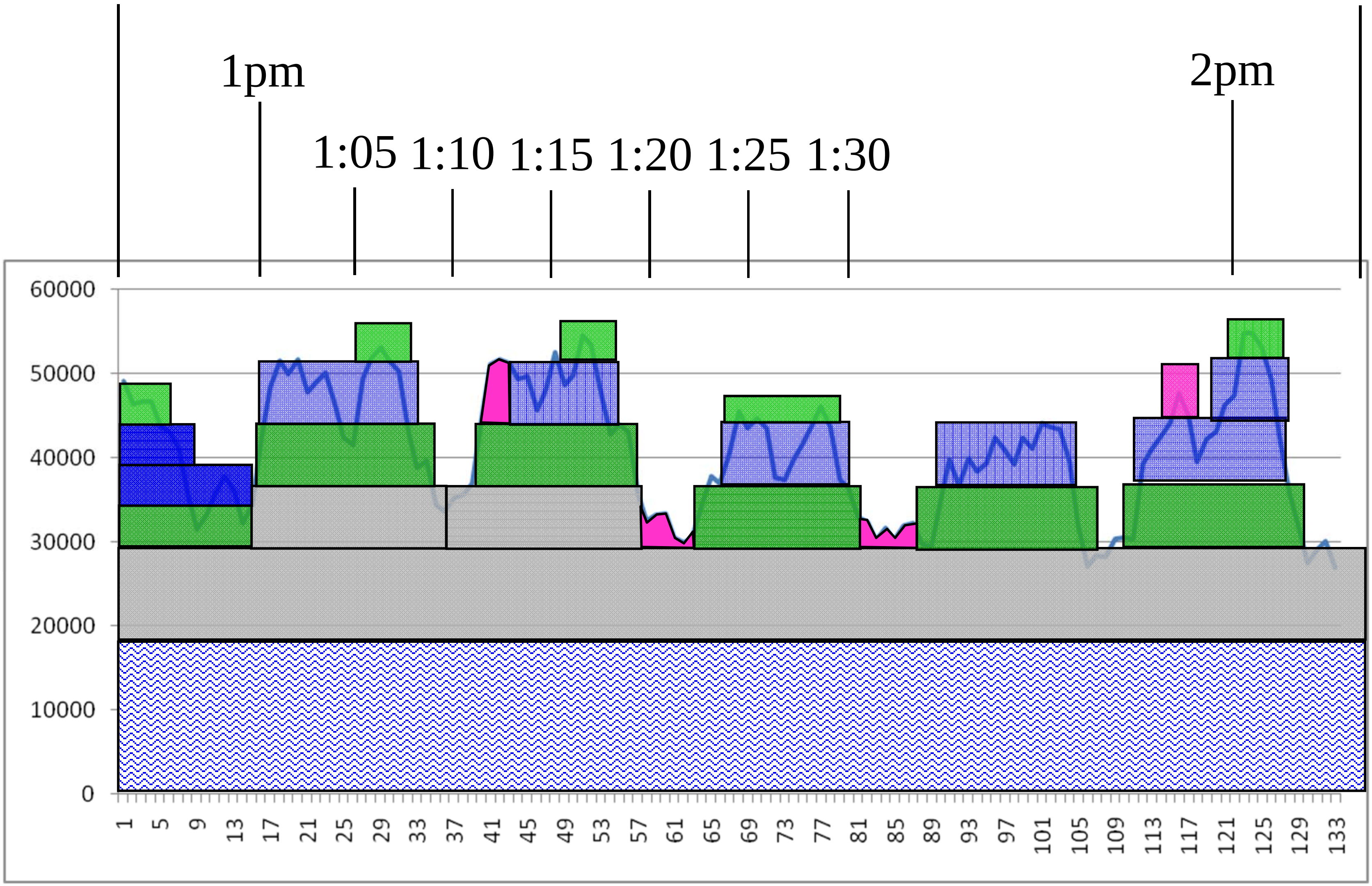
3pm to 4pm:

Steam on/off decisions determined the day before
Optimize within spinning reserve margins
Optimize on/off operation of gas turbines
Constrained by aggregate DC power flow

No decisions are stored or implemented.

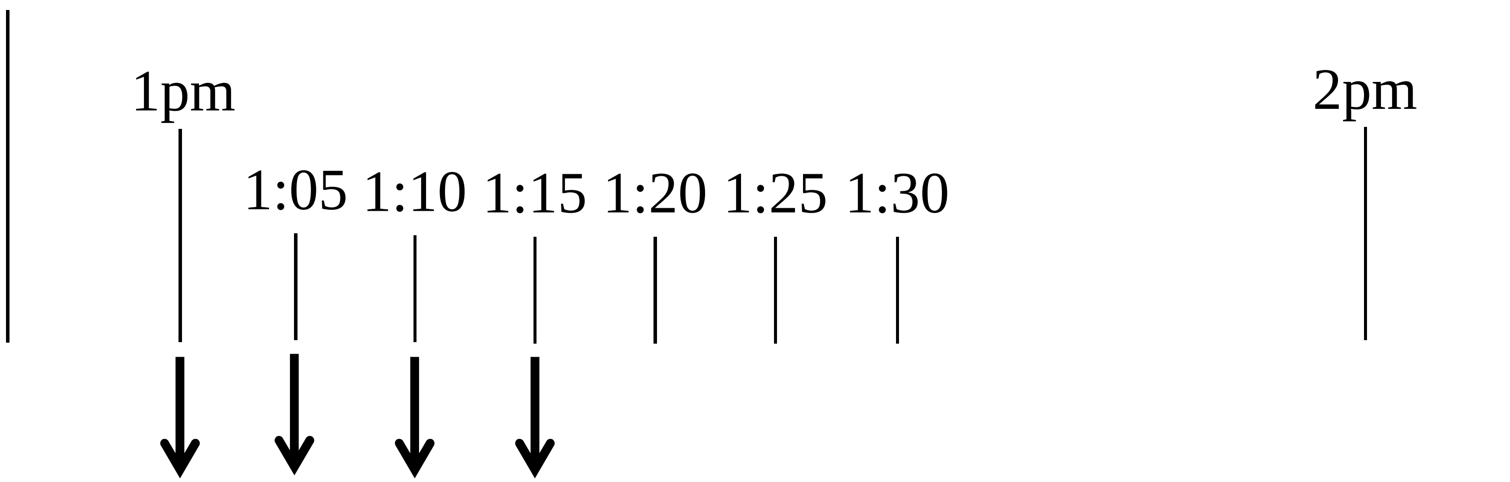
The timing of decisions

- ❑ Economic dispatch



The timing of decisions

- Economic dispatch



Steam on/off decisions determined the day before

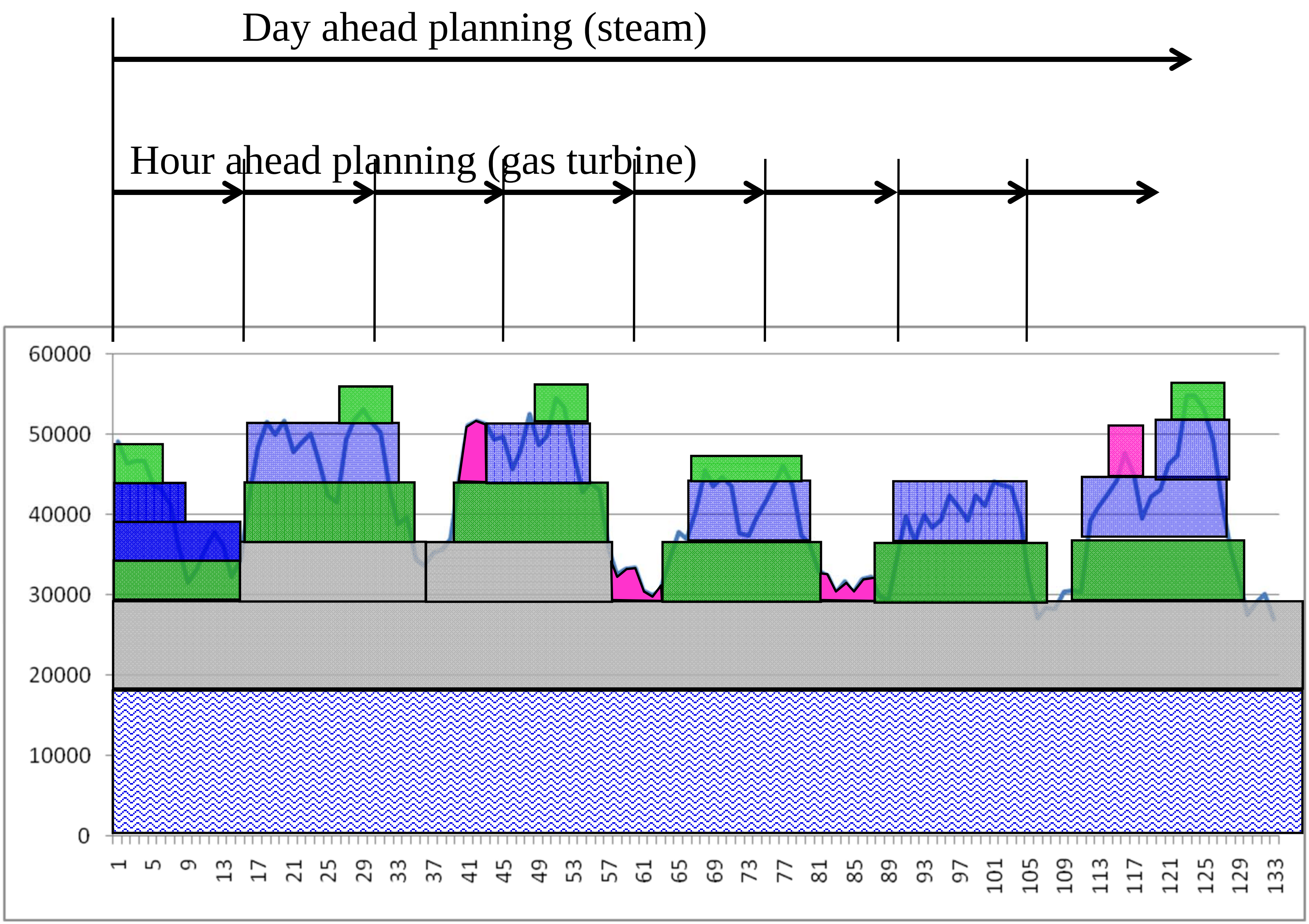
Optimize within spinning reserve margins

On/off of gas turbines determined the hour before

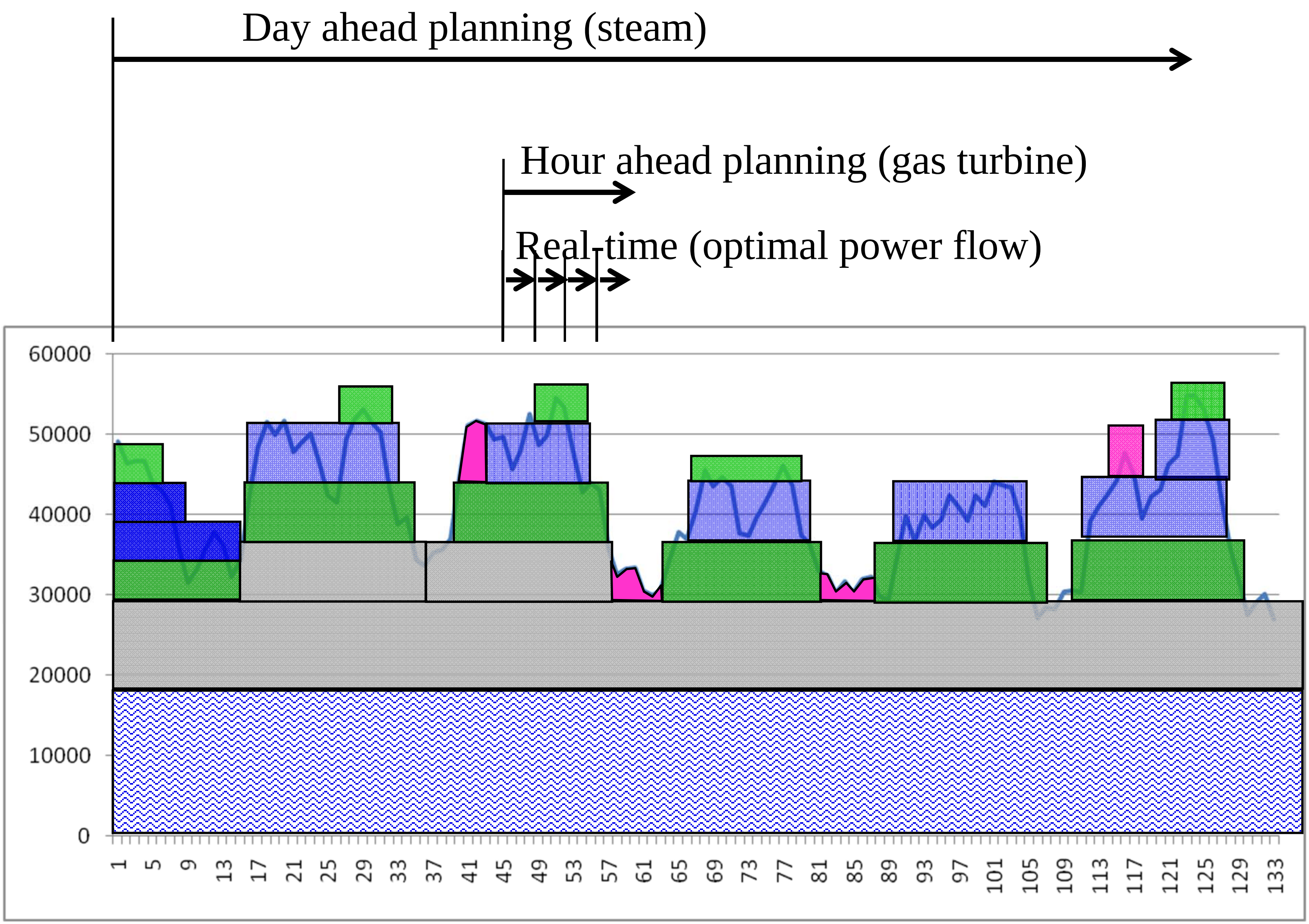
Optimize within adjustment margins

Disaggregate (DC) power flow model

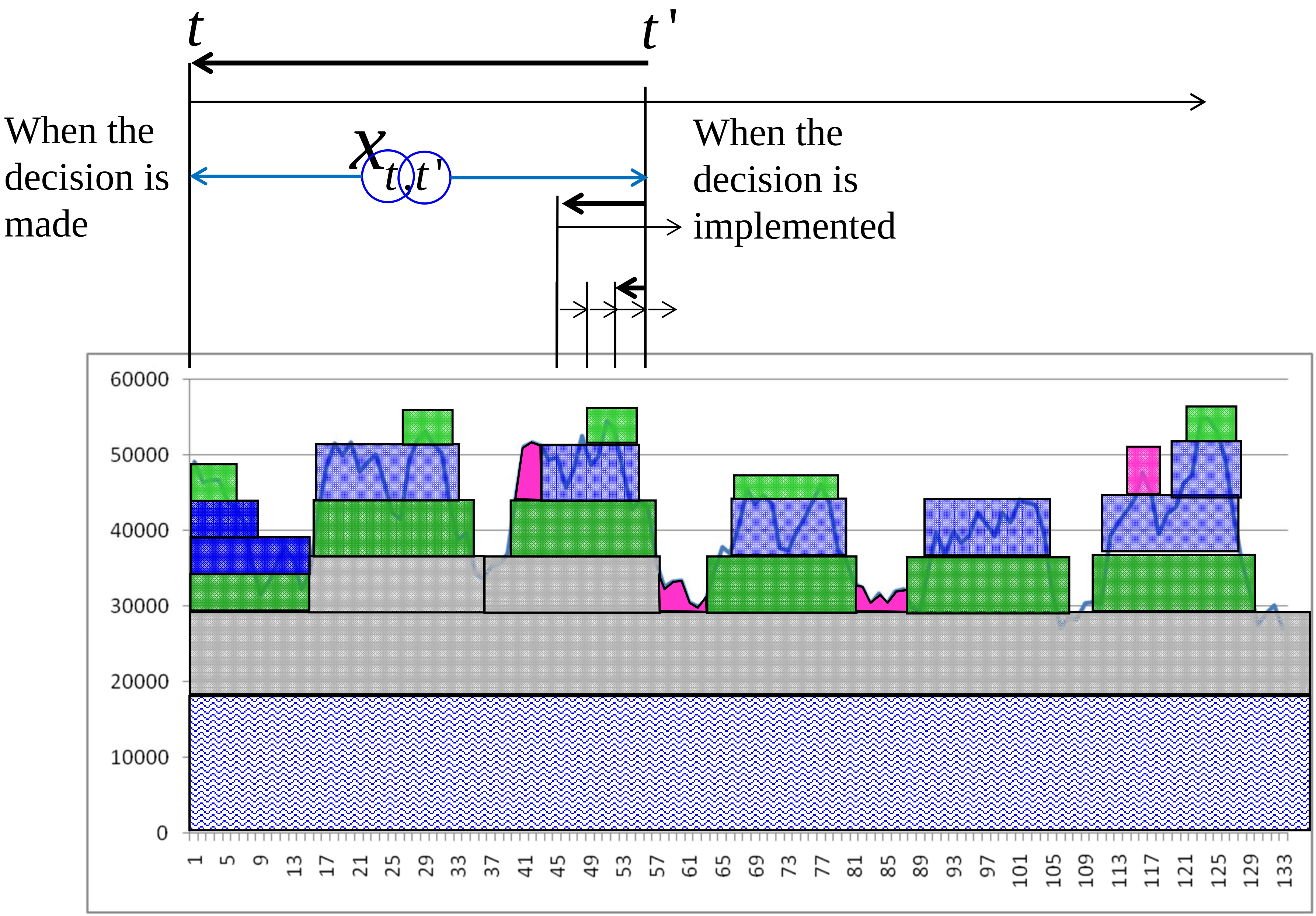
The nesting of decisions



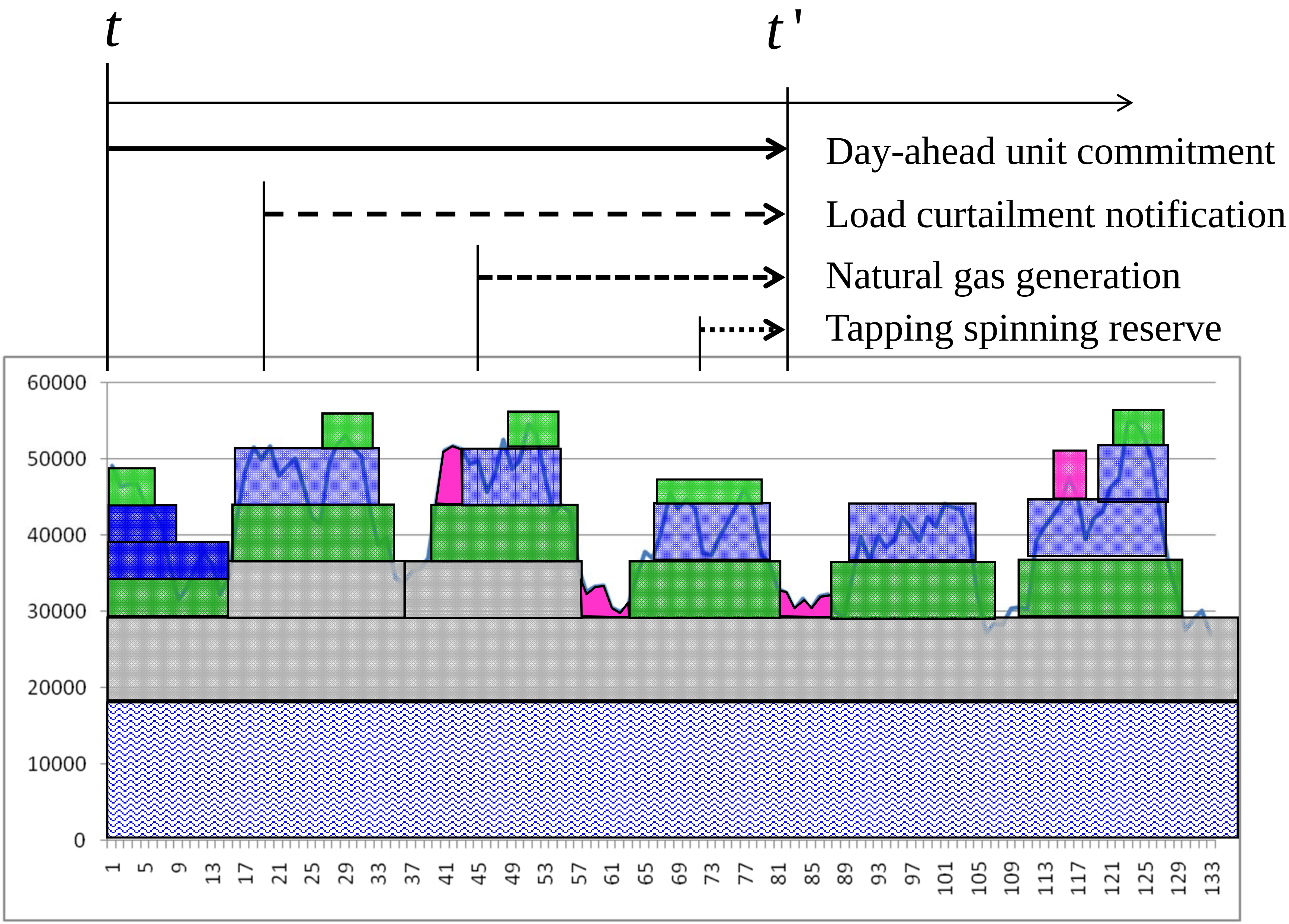
The nesting of decisions



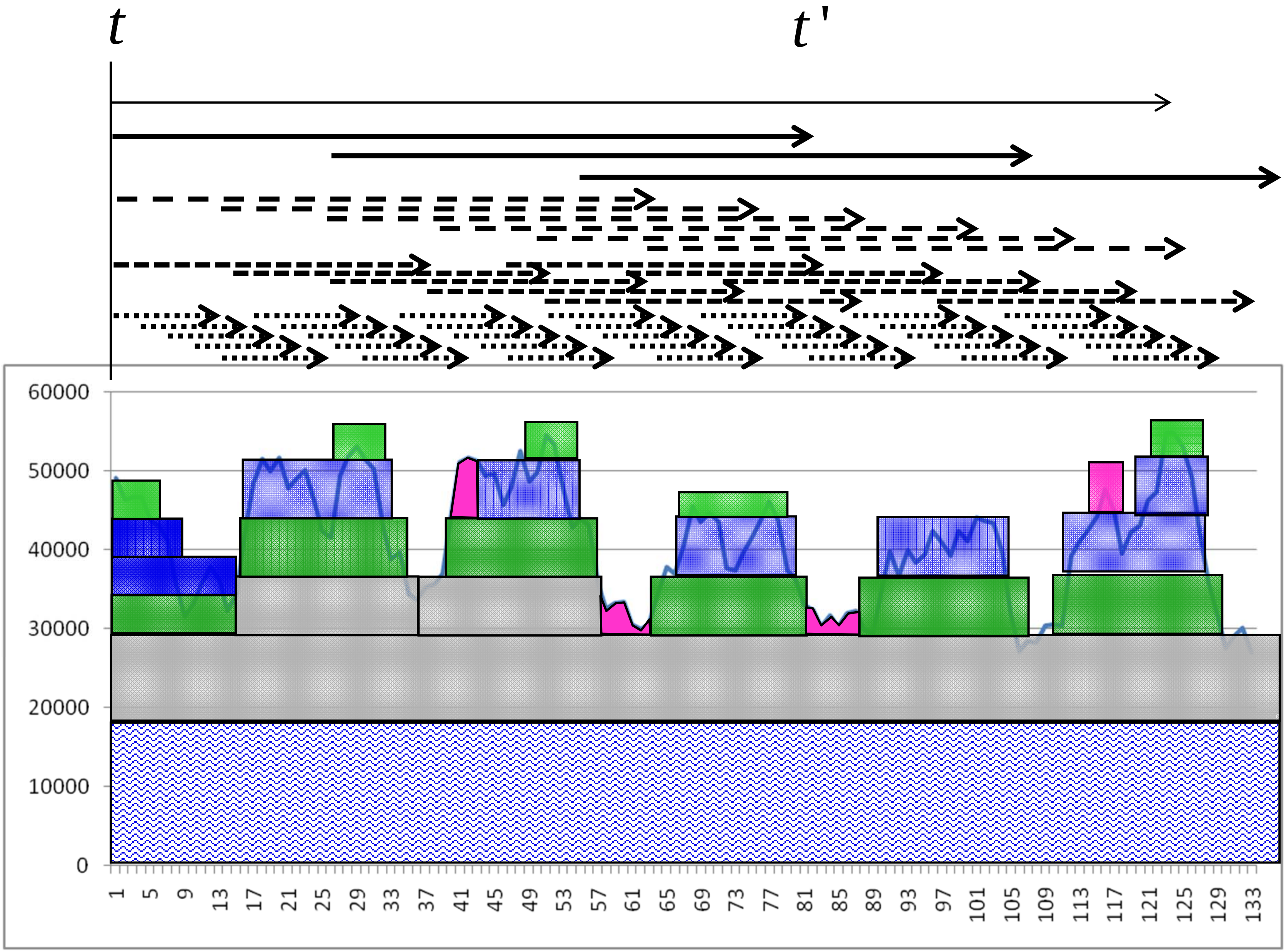
The nesting of decisions



The nesting of decisions



The nesting of decisions



Four classes of policies

1) Cost function approximation

» $X^{CFA}(S_t \mid \theta) = \arg\min_{x_t} \bar{C}^{\pi}(S_t, x_t \mid \theta)$

2) Lookahead policies

» ***Deterministic lookahead:***

$$X_t^{LA-D}(S_t) = \arg\min_{\tilde{x}_{tt}, \tilde{x}_{t,t+1}, \ldots, \tilde{x}_{t,t+T}} C(\tilde{S}_{tt}, \tilde{x}_{tt}) + \sum_{t'=t+1}^{T} \gamma^{t'-t} C(\tilde{S}_{tt'}, \tilde{x}_{tt'})$$

» ***Stochastic lookahead (“stochastic programming”)***

$$X_t^{LA-S}(S_t) = \arg\min C(\tilde{S}_{tt}, \tilde{x}_{tt}) + \sum_{\omega \in \tilde{\Omega}_t} p(\omega) \sum_{t'=t+1}^{T} \gamma^{t'-t} C(\tilde{S}_{tt'}(\omega), \tilde{x}_{tt'}(\omega))$$

3) Policy function approximations

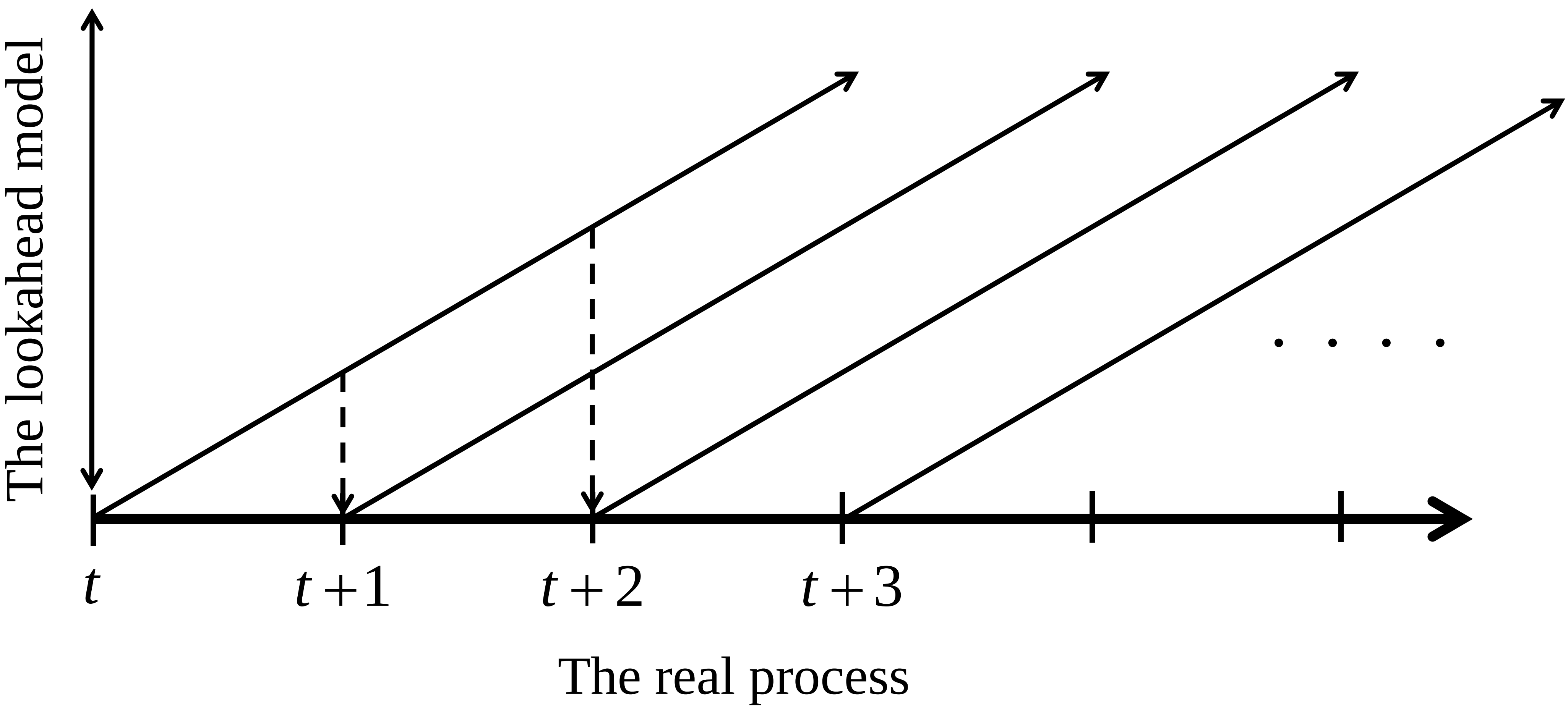
» Lookup tables, rules, parametric functions

4) Policies based on value function approximations

» $X_t^{VFA}(S_t) = \arg\min_{x_t} \left(C(S_t, x_t) + \gamma \bar{V}_t^{x} \left(S_t^{x}(S_t, x_t) \right) \right)$

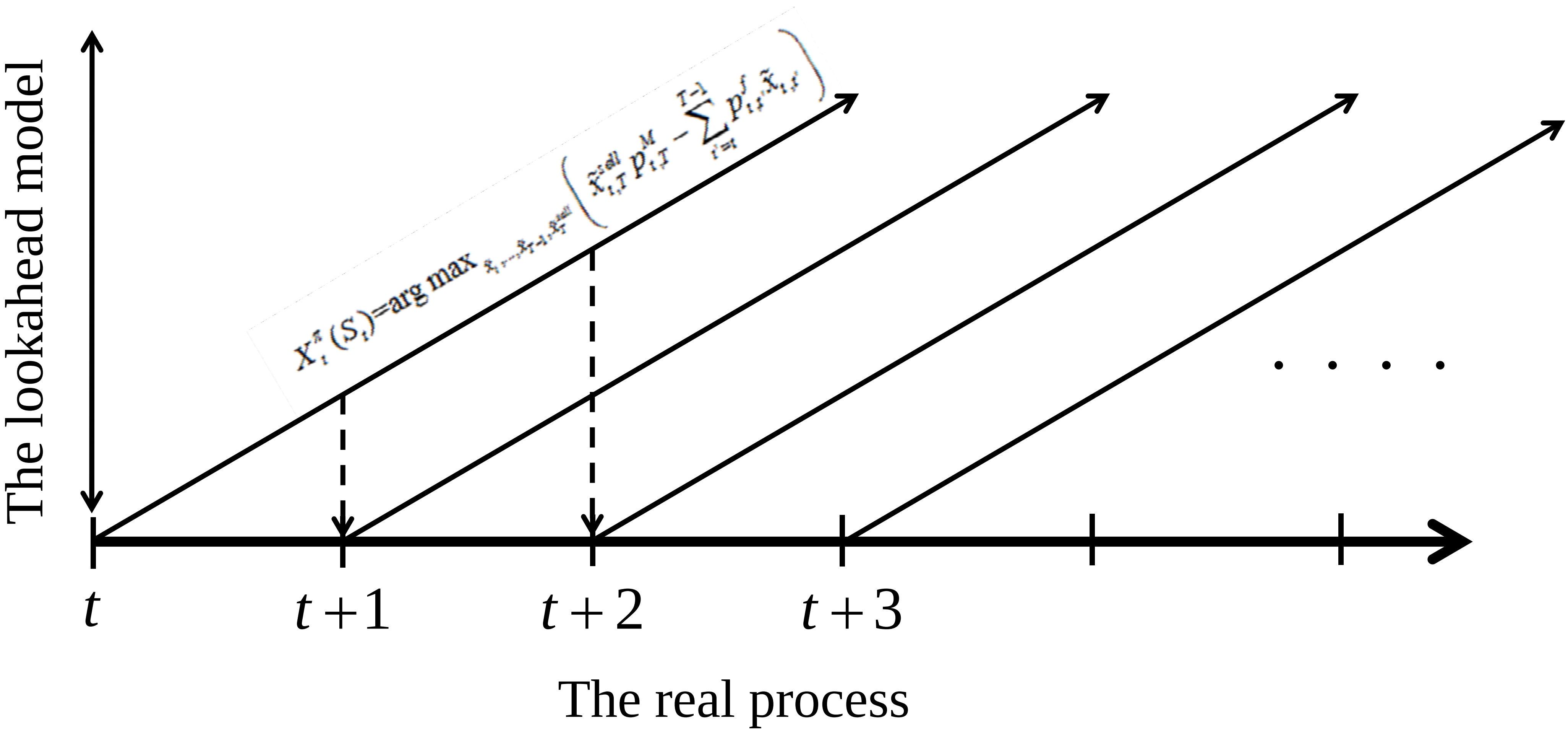
Lookahead policies

- Lookahead policies peek into the future
  - » Optimize over deterministic lookahead model



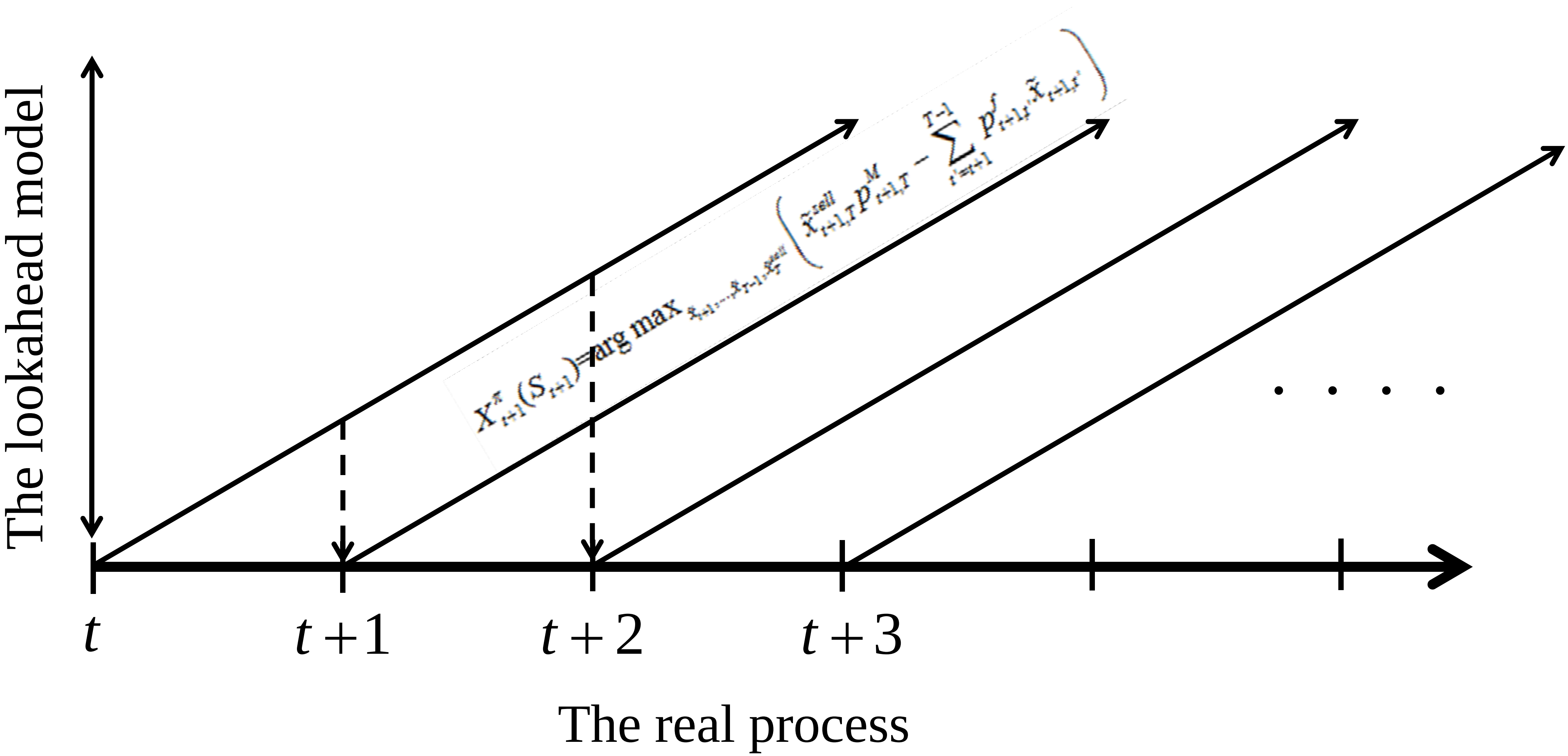
Lookahead policies

- Lookahead policies peek into the future
  - » Optimize over deterministic lookahead model



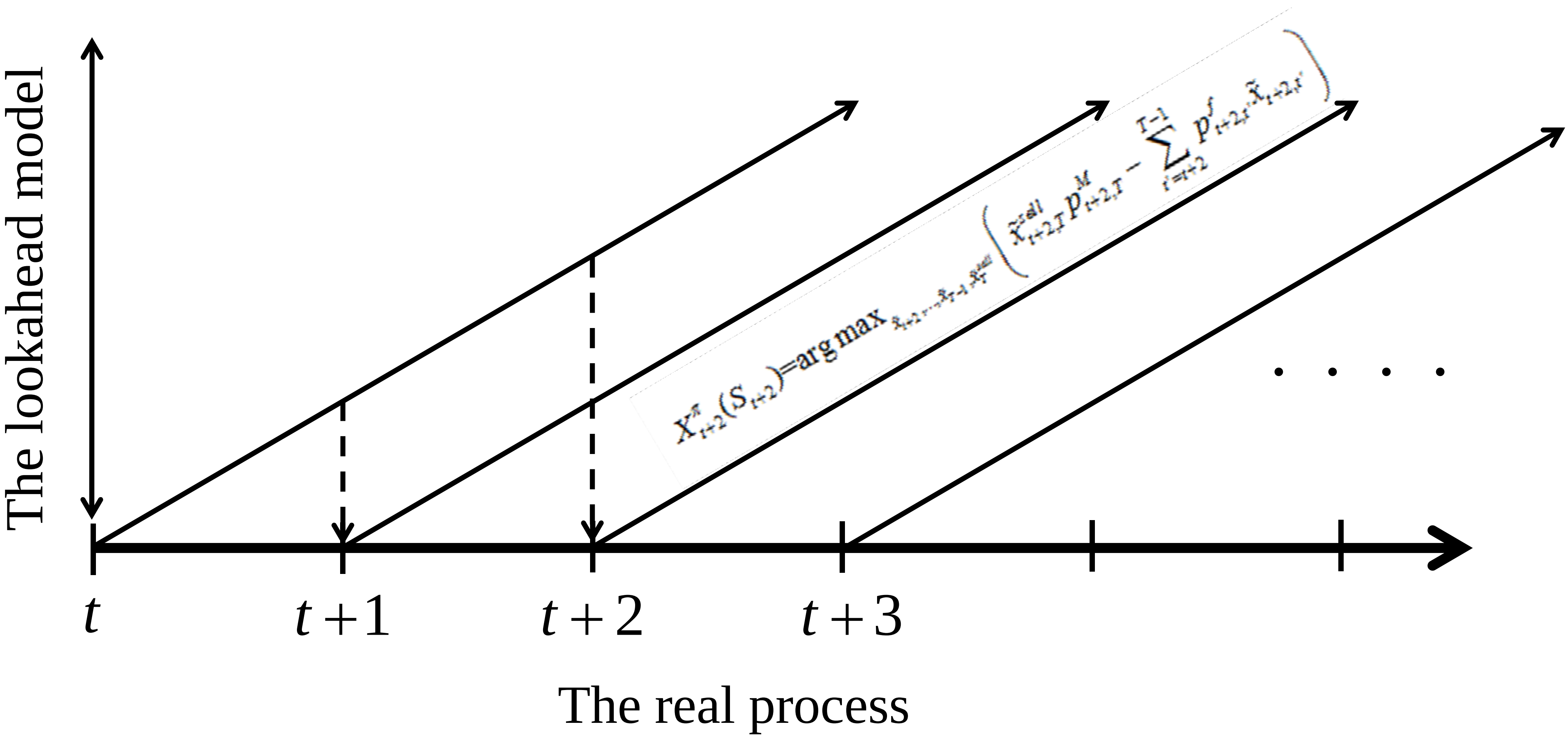
Lookahead policies

- Lookahead policies peek into the future
  - » Optimize over deterministic lookahead model



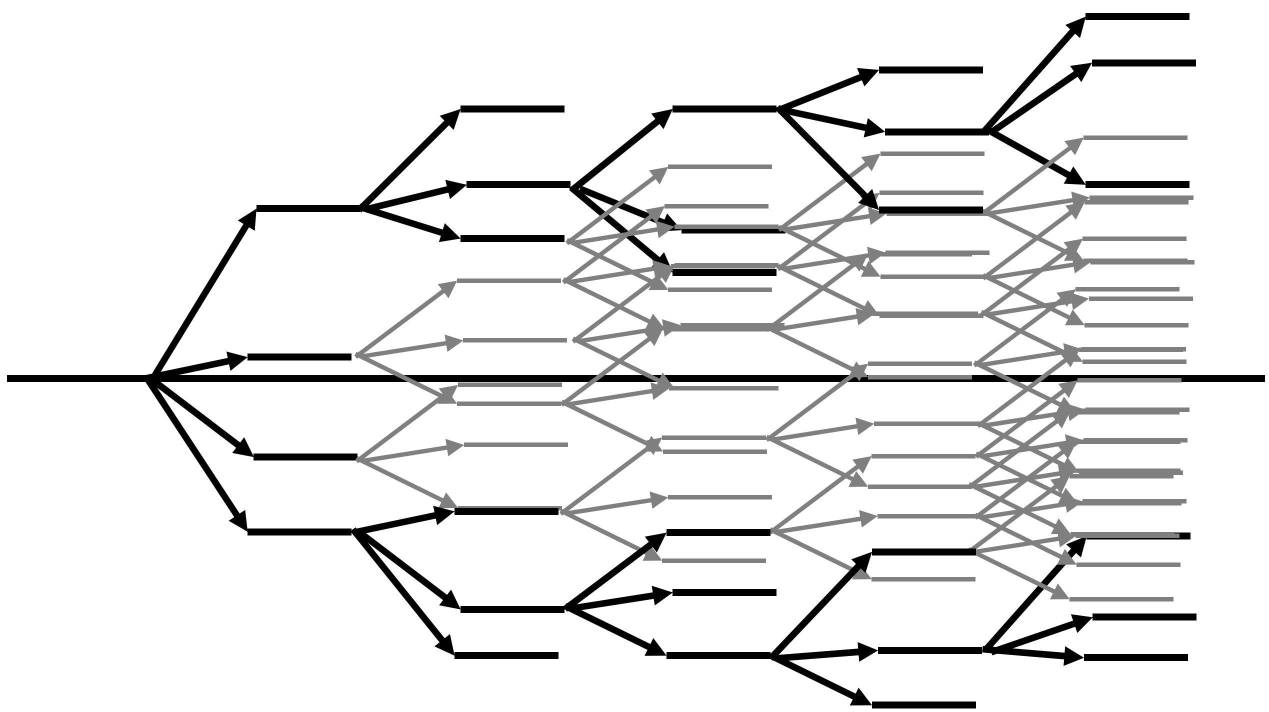
Lookahead policies

- ❑ Lookahead policies peek into the future
  - » Optimize over deterministic lookahead model



Lookahead policies

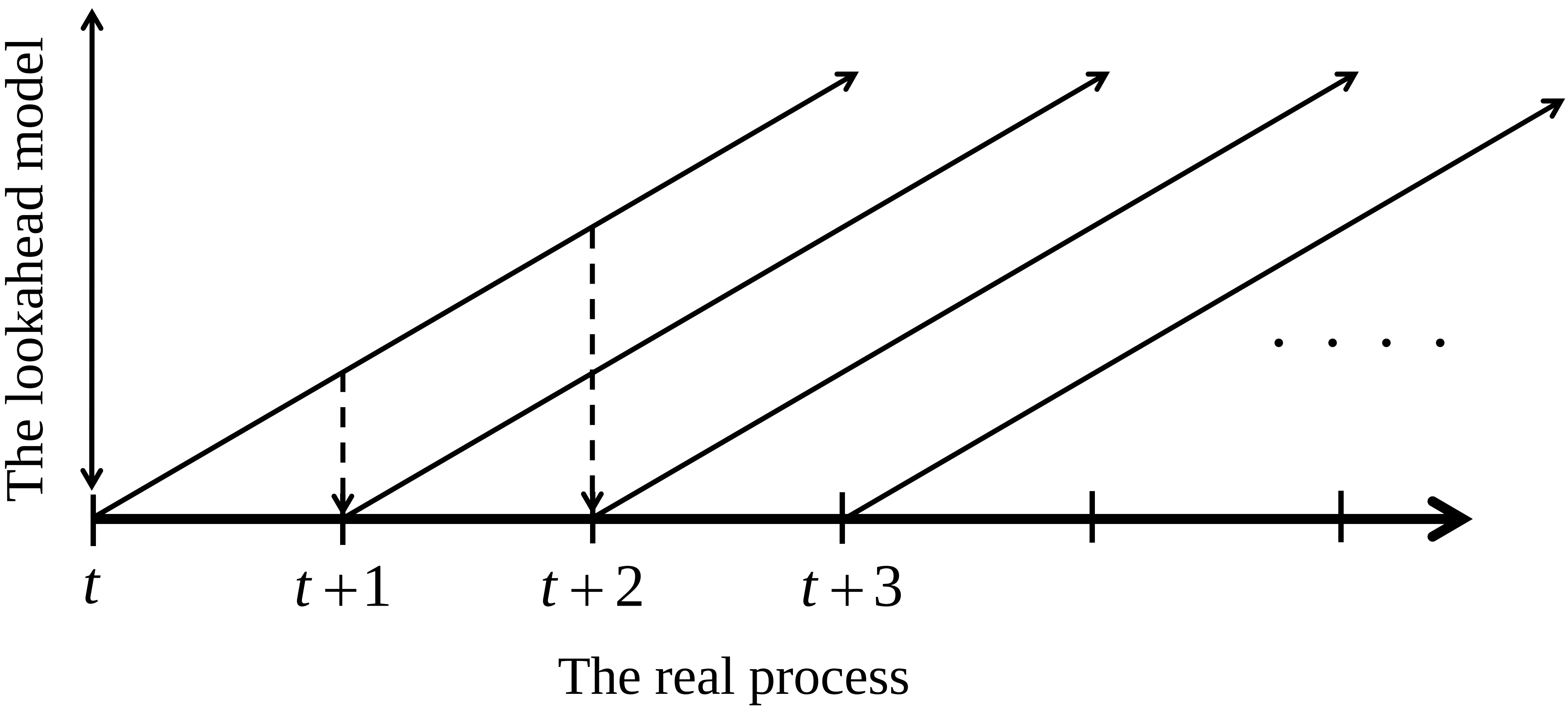
- Probabilistic lookahead
  - » Here, we approximate the information model by using a Monte Carlo sample to create a scenario tree:



  - » We can try to solve this as a single "deterministic" optimization problem. This is a *direct lookahead policy*.

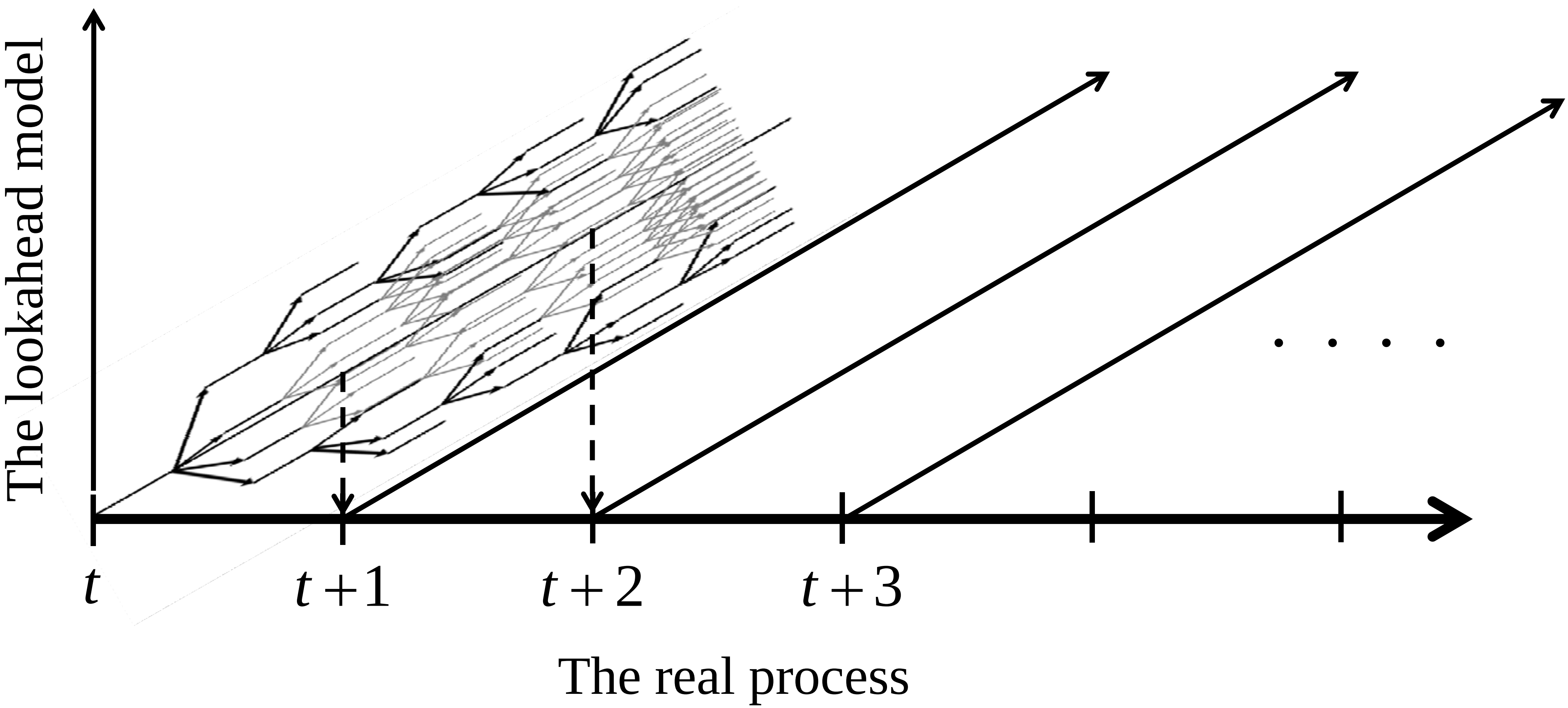
Lookahead policies

- We can then simulate this *lookahead policy* over time:



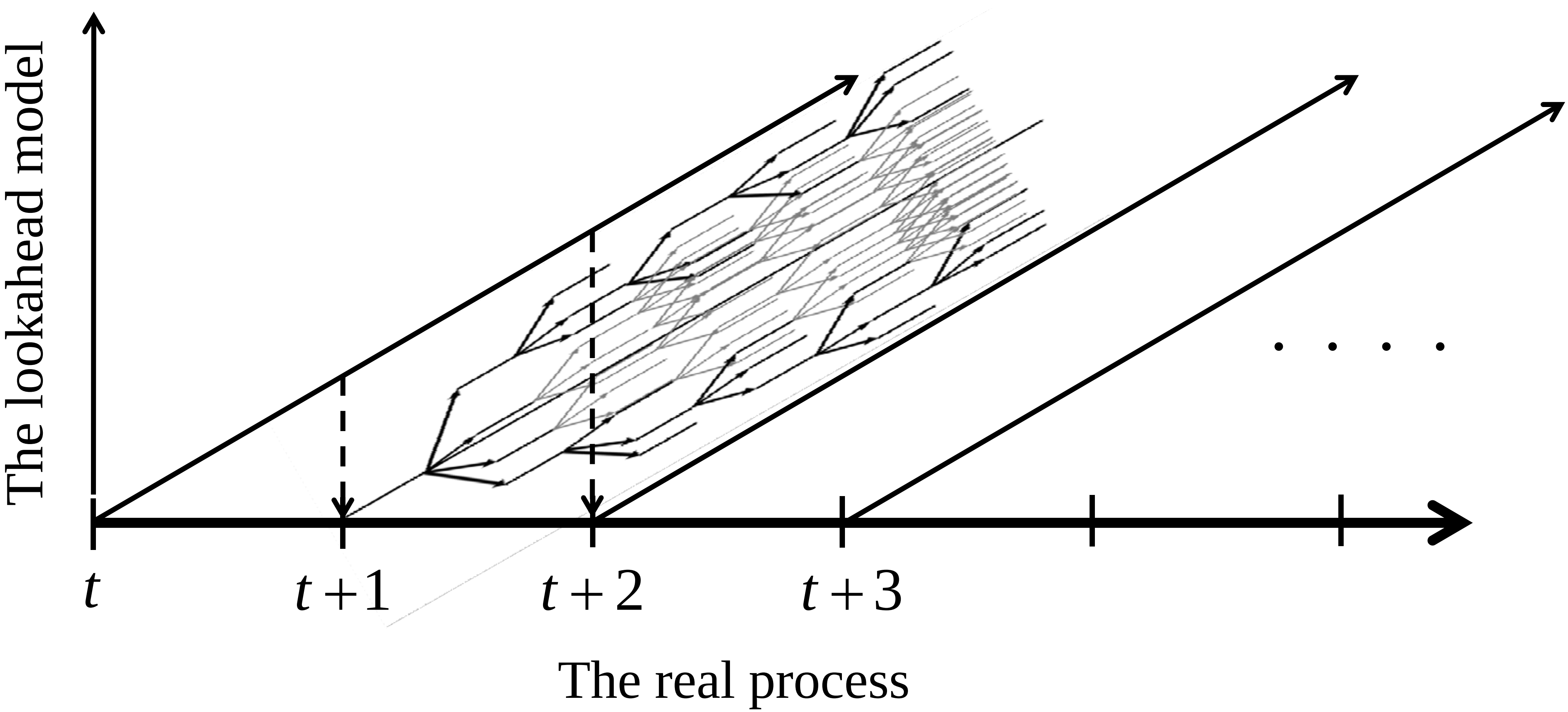
Lookahead policies

- We can then simulate this *lookahead policy* over time:



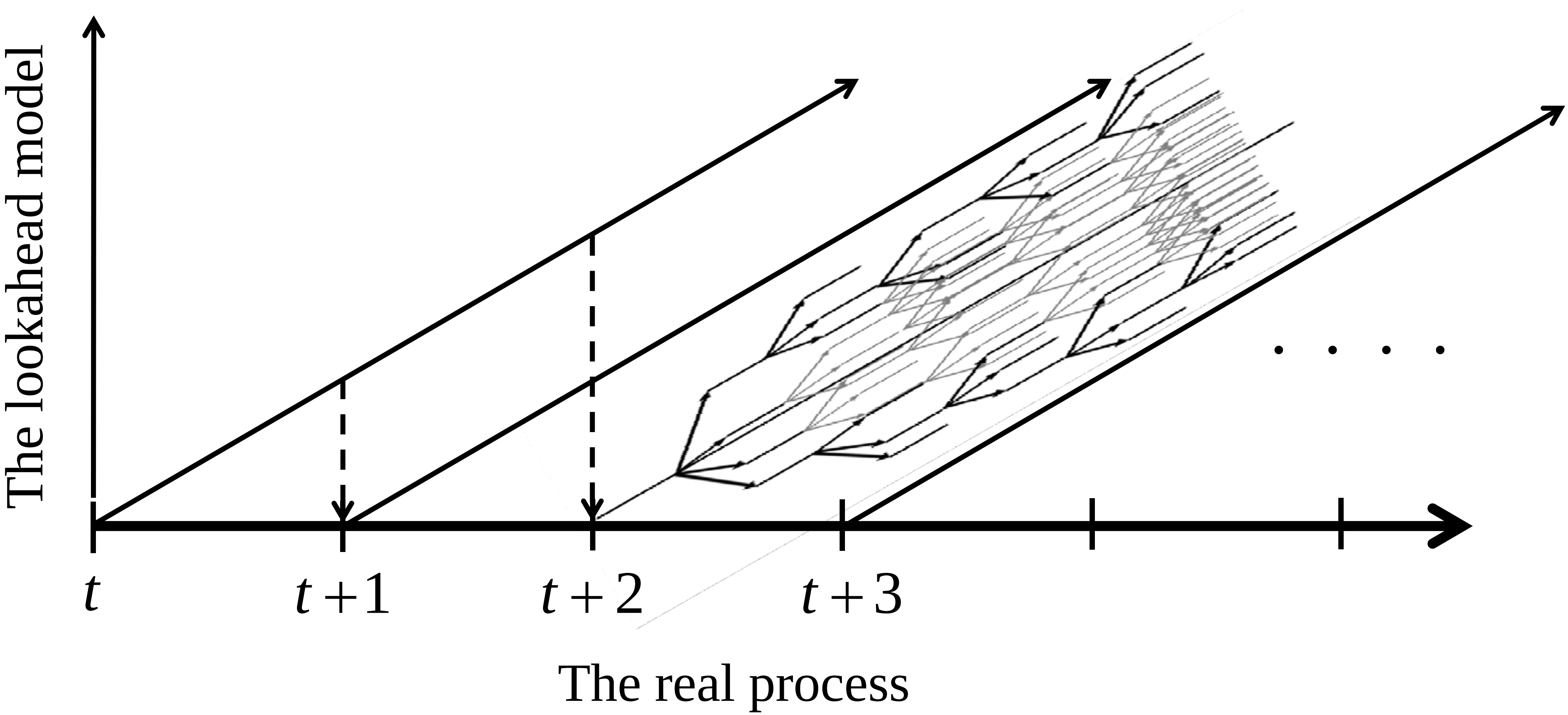
Lookahead policies

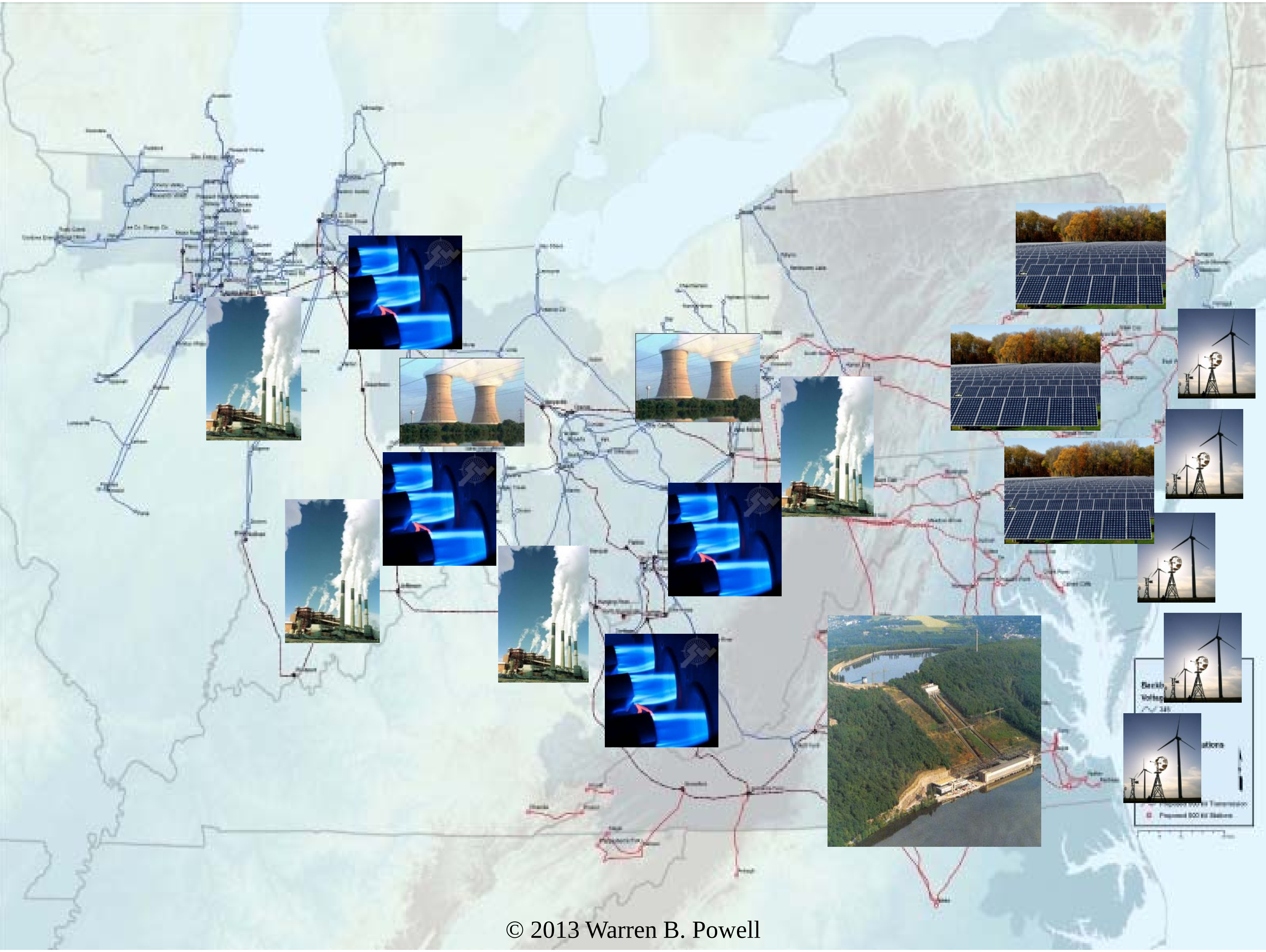
- We can then simulate this *lookahead policy* over time:

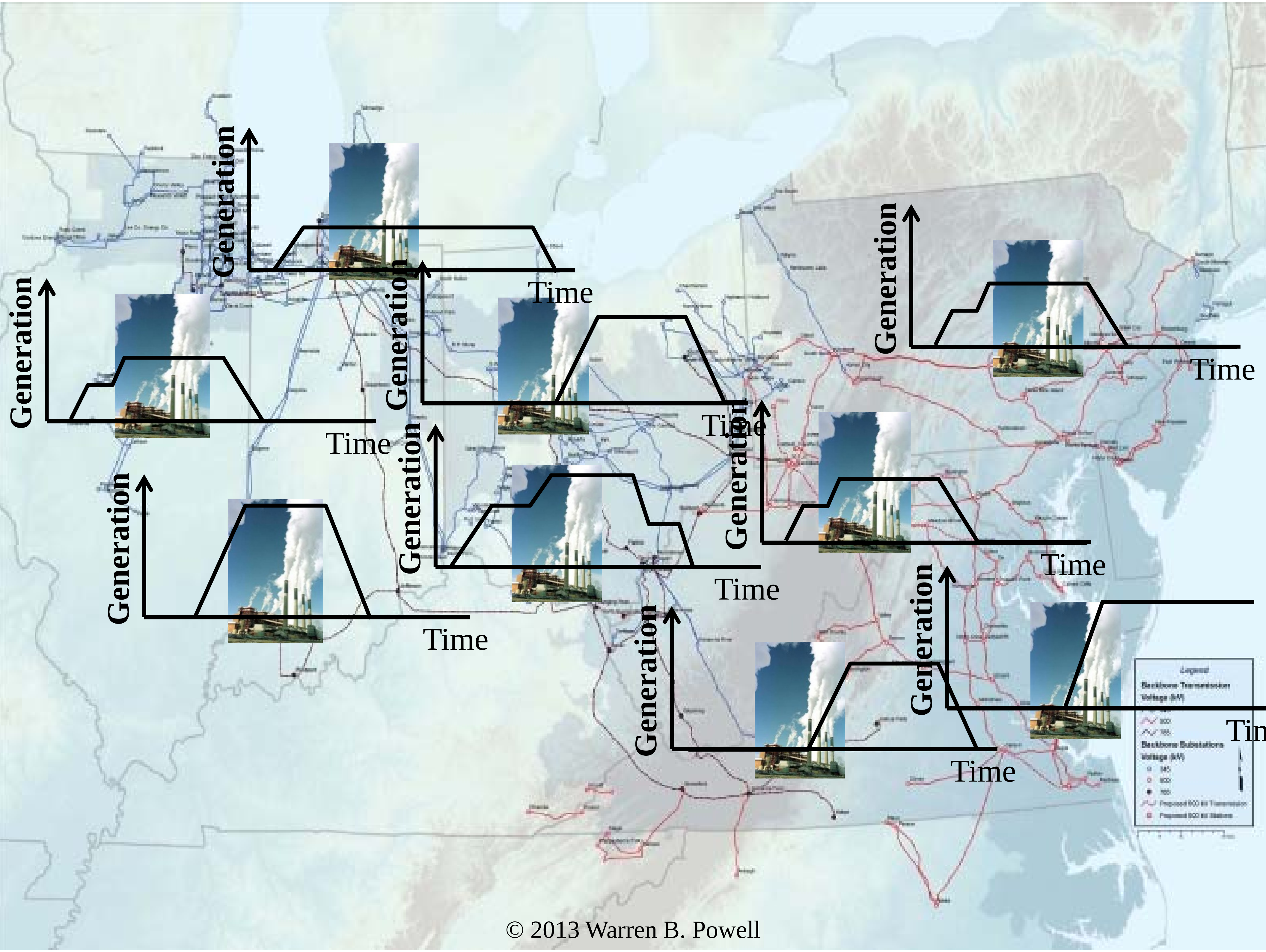


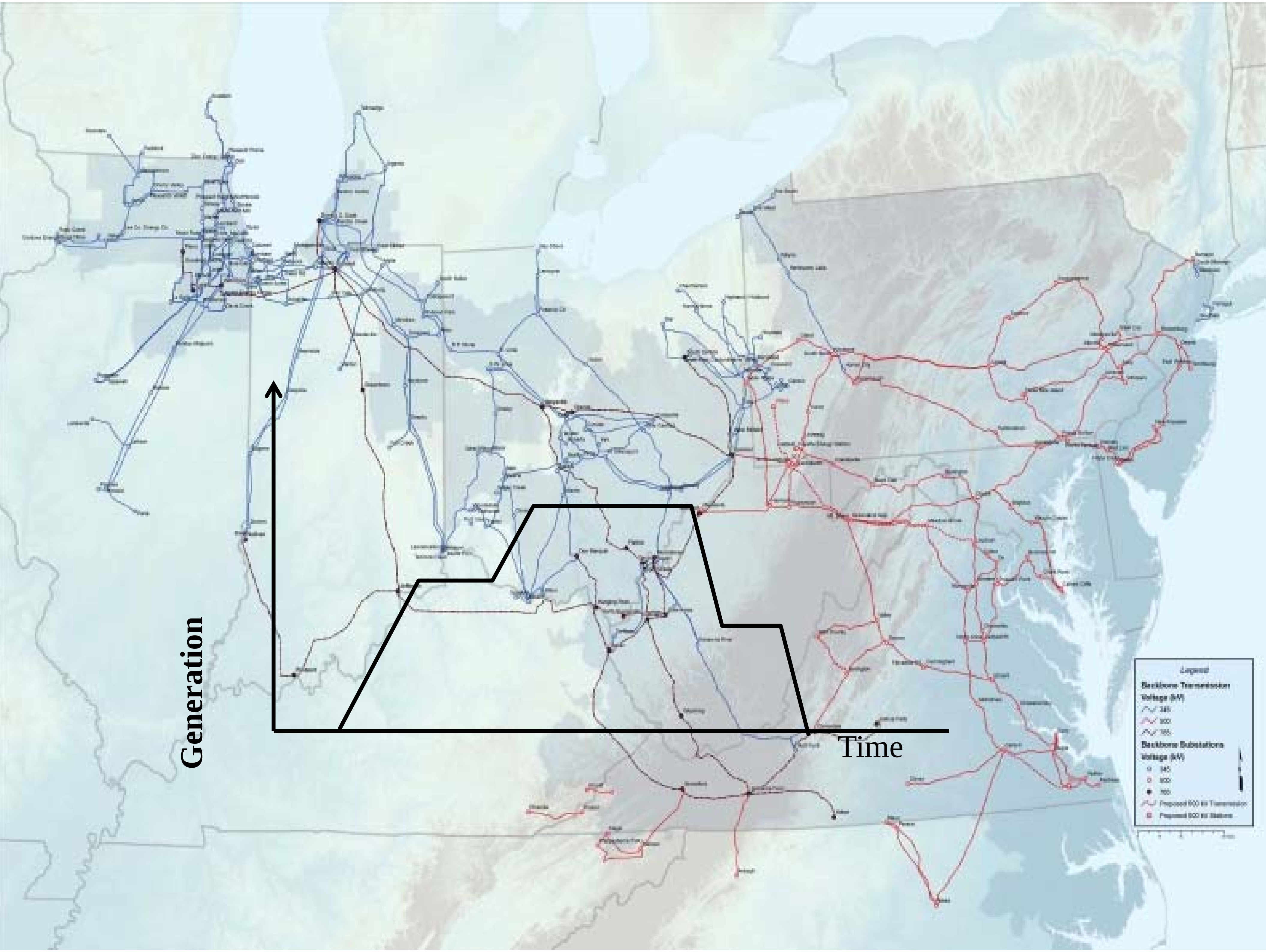
Lookahead policies

- We can then simulate this *lookahead policy* over time:



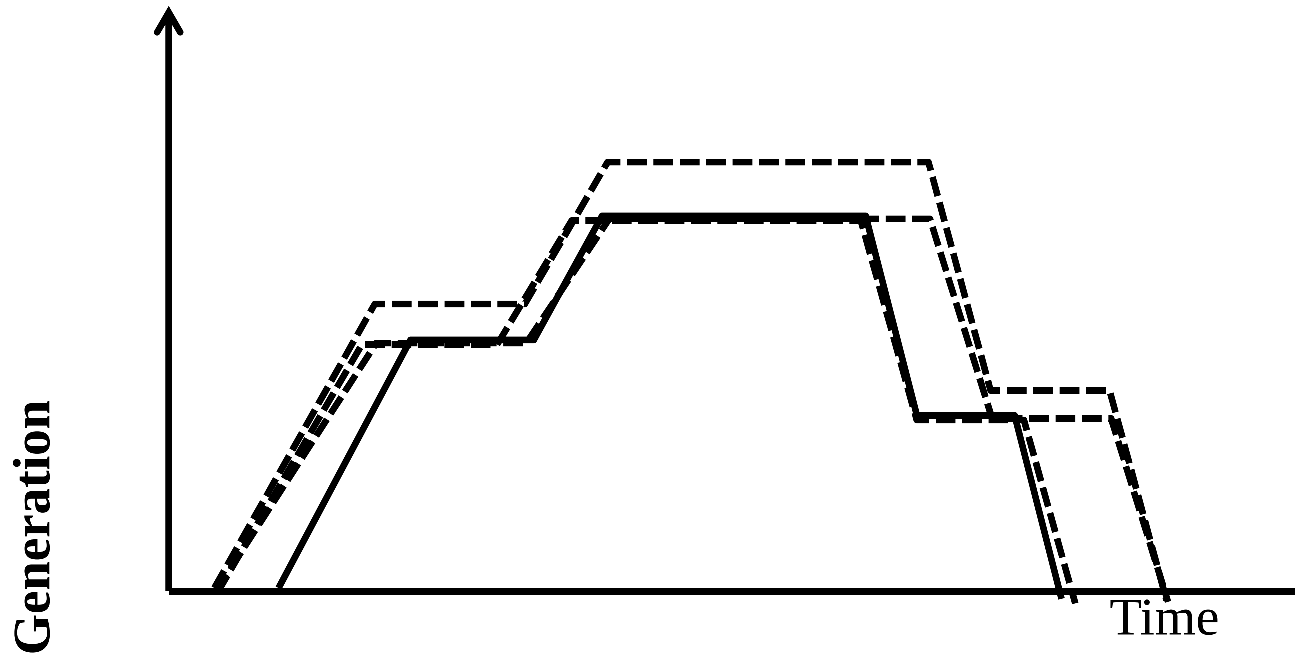



Generation
Time
Generation
Time
Generation
Time
Generation
Time
Generation
Time
Generation
Time
Generation
Time
Generation
Time
Generation
Tin


Generation
Time

Robust unit commitment

- We need to achieve a robust schedule:



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 ($x_{t'}$) Gas turbines ($y_{t'}$)

  - » 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
  - » We capture the information content of decisions

$$F_t(S_t) = \min_{\substack{(x_{t,t'})_{t'=1,\dots,24} \\ (y_{t',t'})_{t'=1,\dots,24}}} \mathbb{E} \sum_{t'=t}^{t+24} C(x_{t,t'}, y_{t',t'})$$

$x_{t,t'}$

$y_{t',t'}$

- $x_{t,t'}$ 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'+1*

  - » 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 lookahead model
  - » We capture the information content of decisions

$$F_t(S_t|\theta) = \min_{\substack{(x_{tt'})_{t'=1,...,24} \\ \pi}} \mathbb{E} \sum_{t'=1}^{24} C(x_{tt'}, Y_{t'}^{\pi}(S_{tt'}))$$

Policy ← π

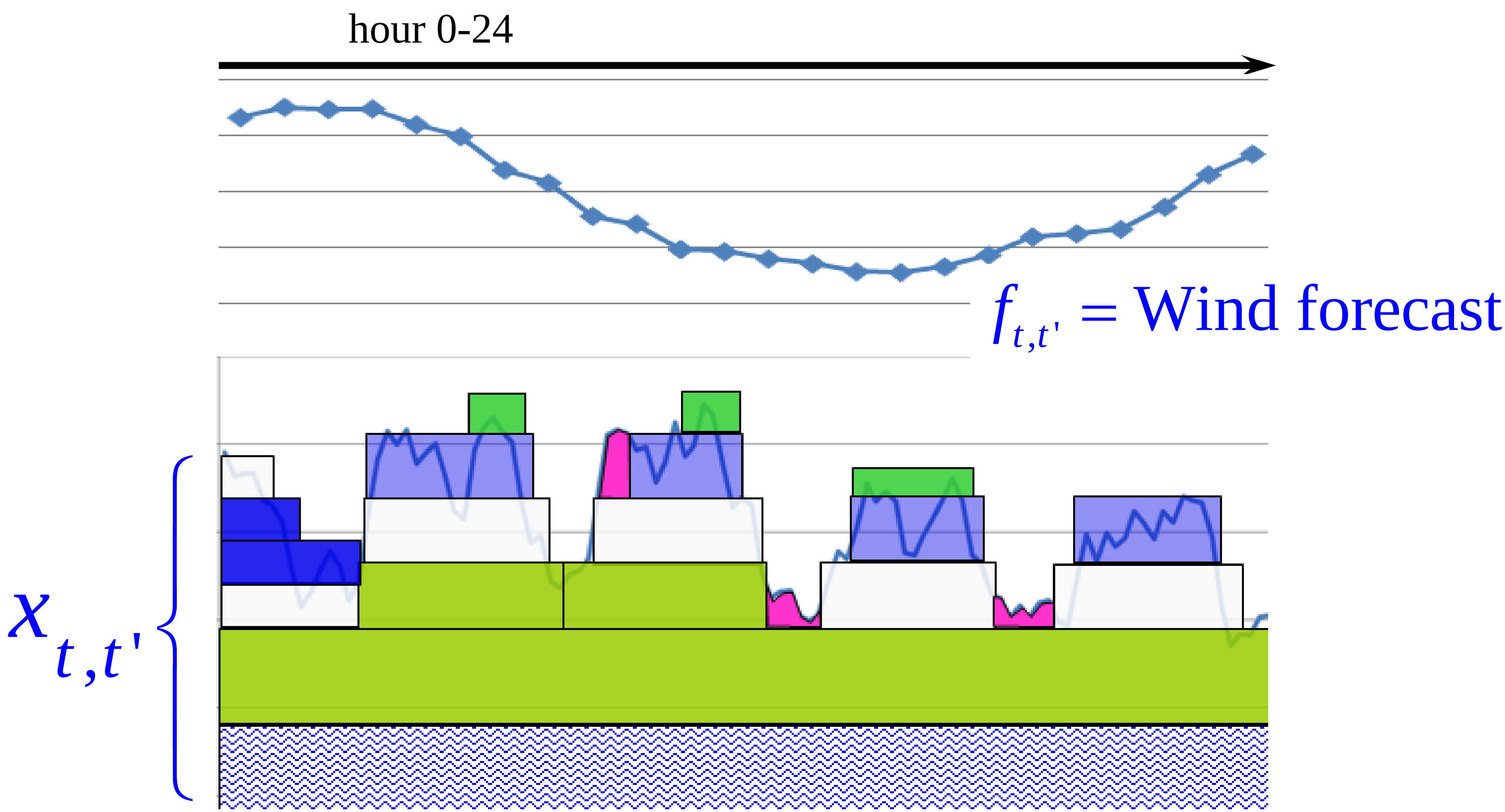
$$x_{t,t'}^{\max} - x_{t,t'} \geq \theta L_{tt'}$$ Reserve must be a fraction of the load

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

  - » The challenge now is to adaptively estimate the ramping constraints $\theta_{tt'}$, and the policies $Y^{\pi}(S_{tt'})$.

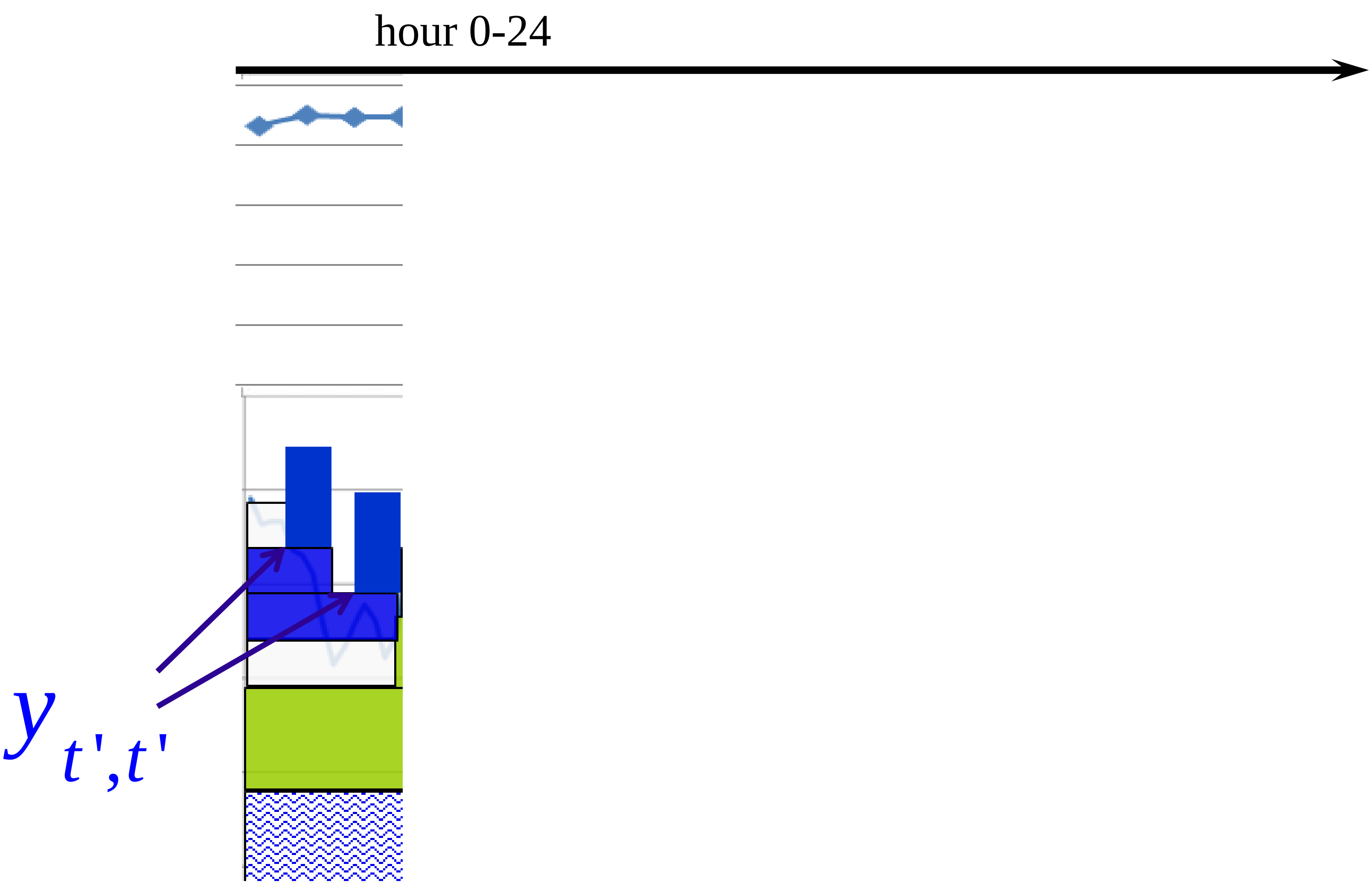
The stochastic unit commitment problem

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



The stochastic unit commitment problem

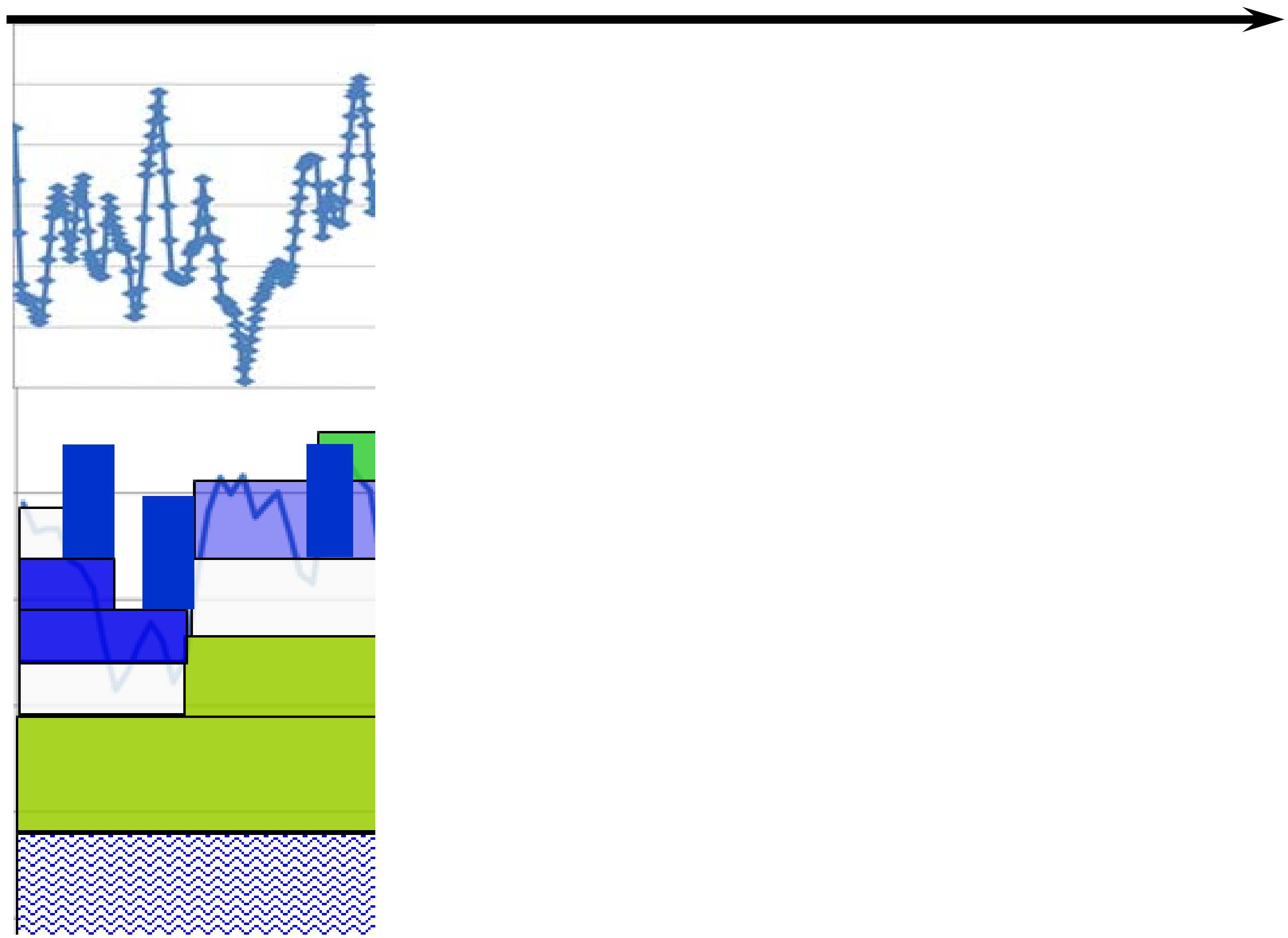
- The unit commitment problem
  - » Stepping forward observing actual wind, making small adjustments



The stochastic unit commitment problem

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

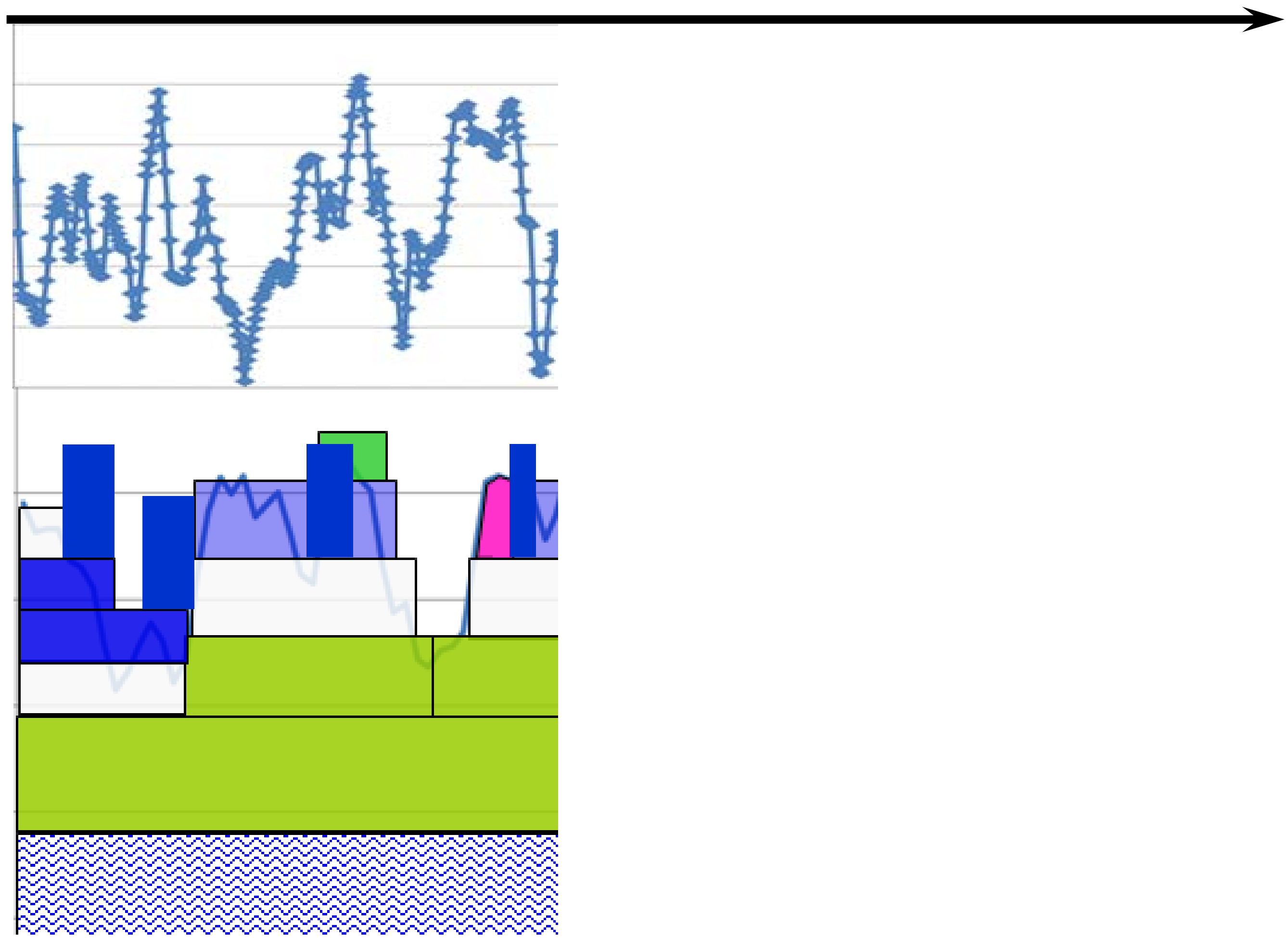
hour 0-24



The stochastic unit commitment problem

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

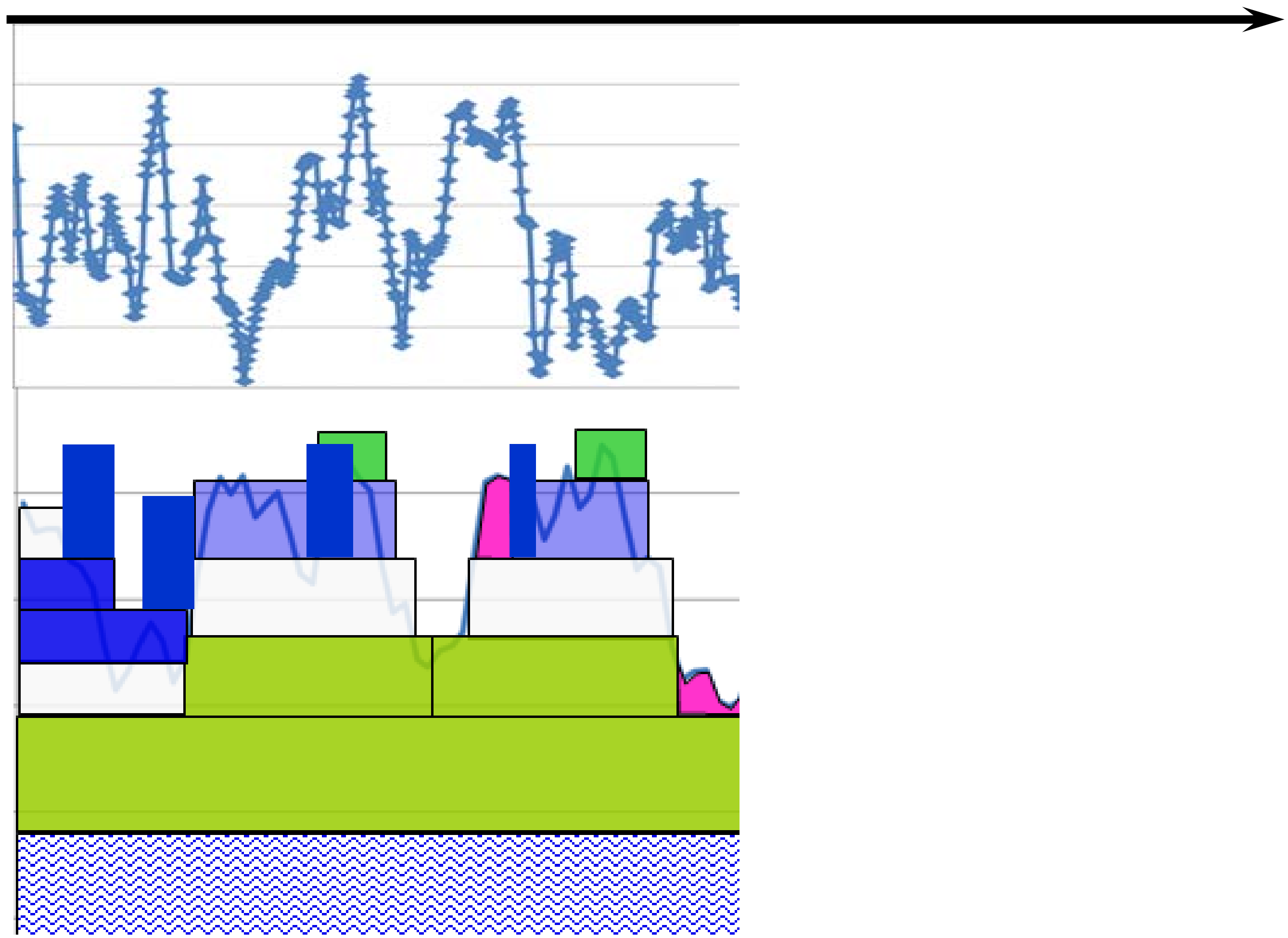
hour 0-24



The stochastic unit commitment problem

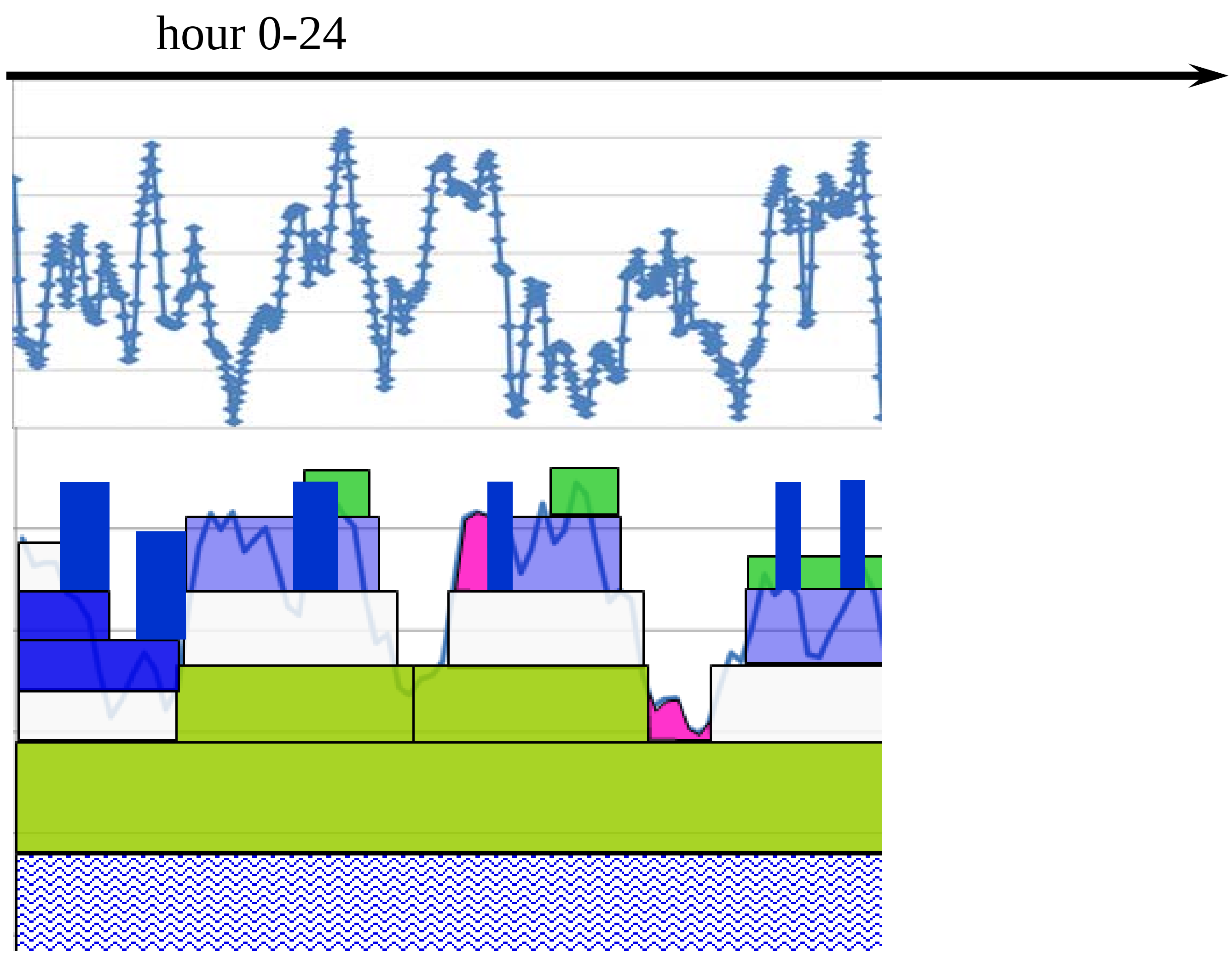
- The unit commitment problem
  - » Stepping forward observing actual wind, making small adjustments

hour 0-24



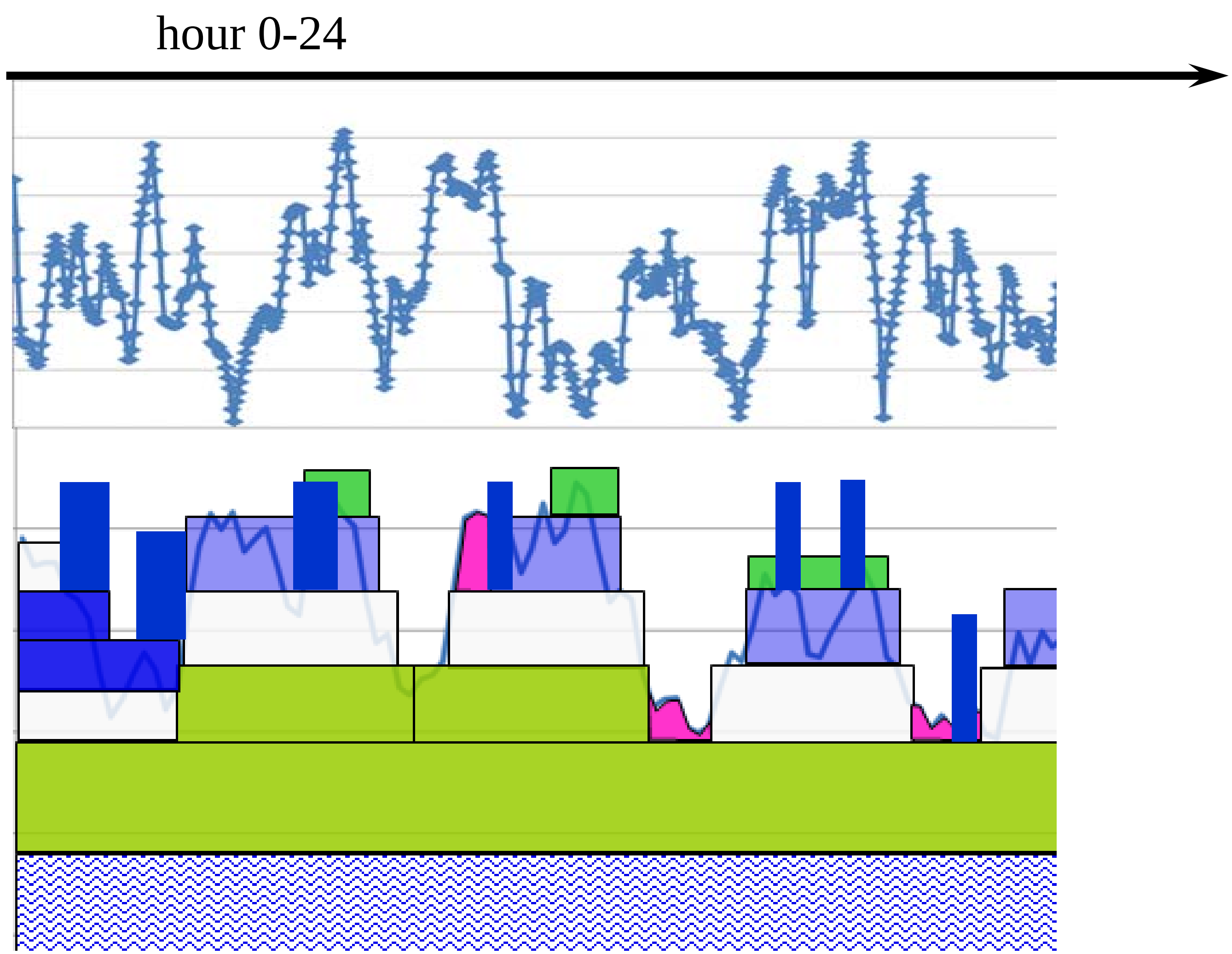
The stochastic unit commitment problem

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



The stochastic unit commitment problem

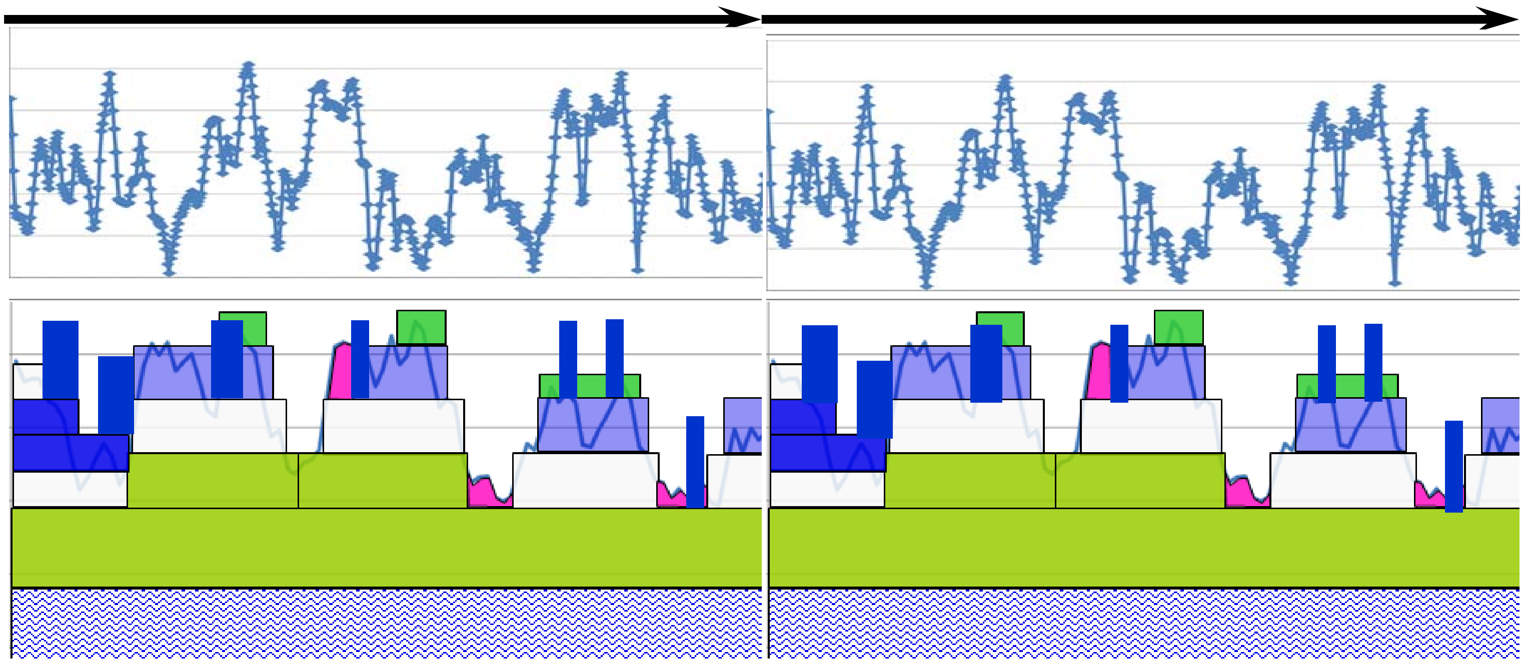
- The unit commitment problem
  - » Stepping forward observing actual wind, making small adjustments



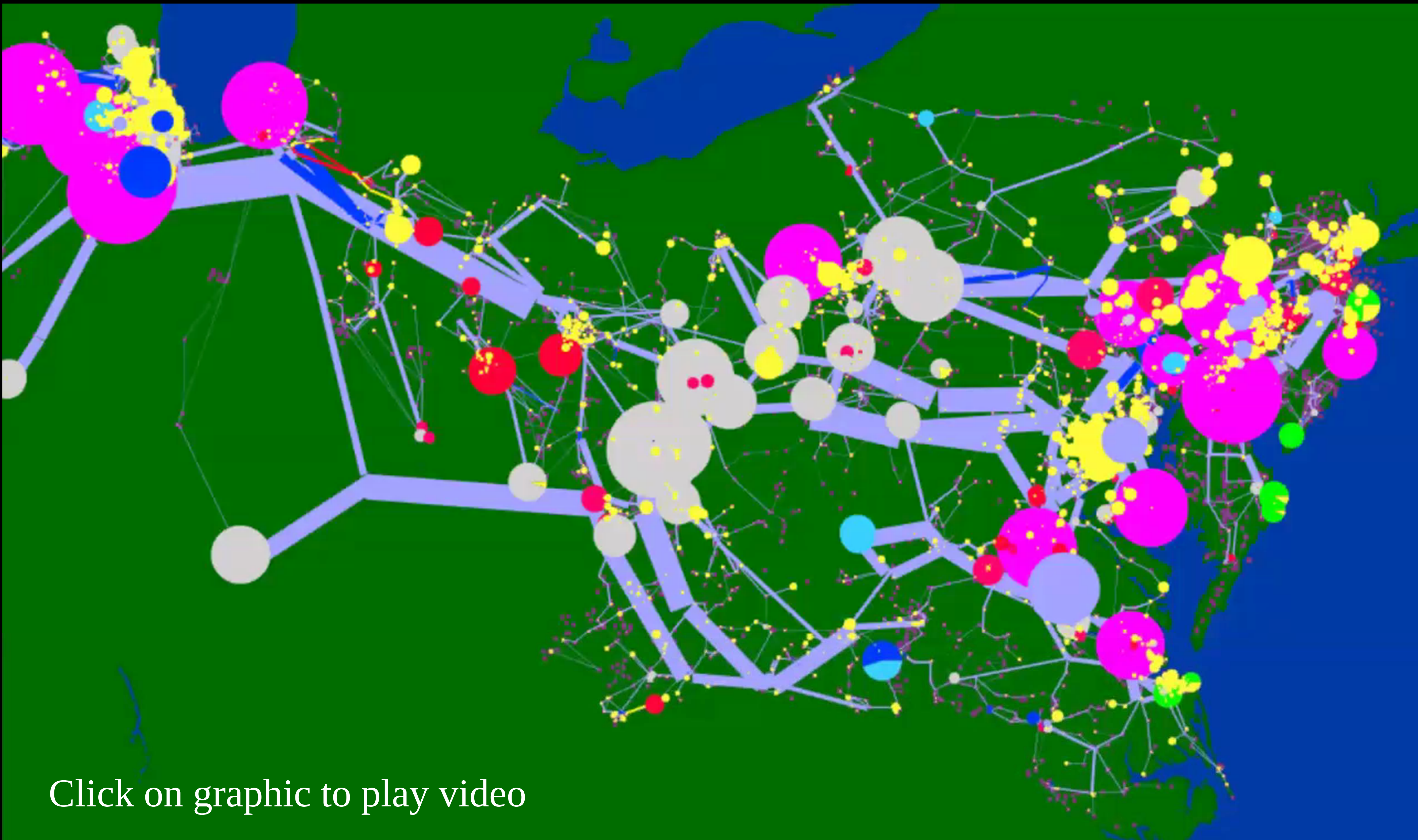
The stochastic unit commitment problem

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

Hours 0-24 Hours 25-48

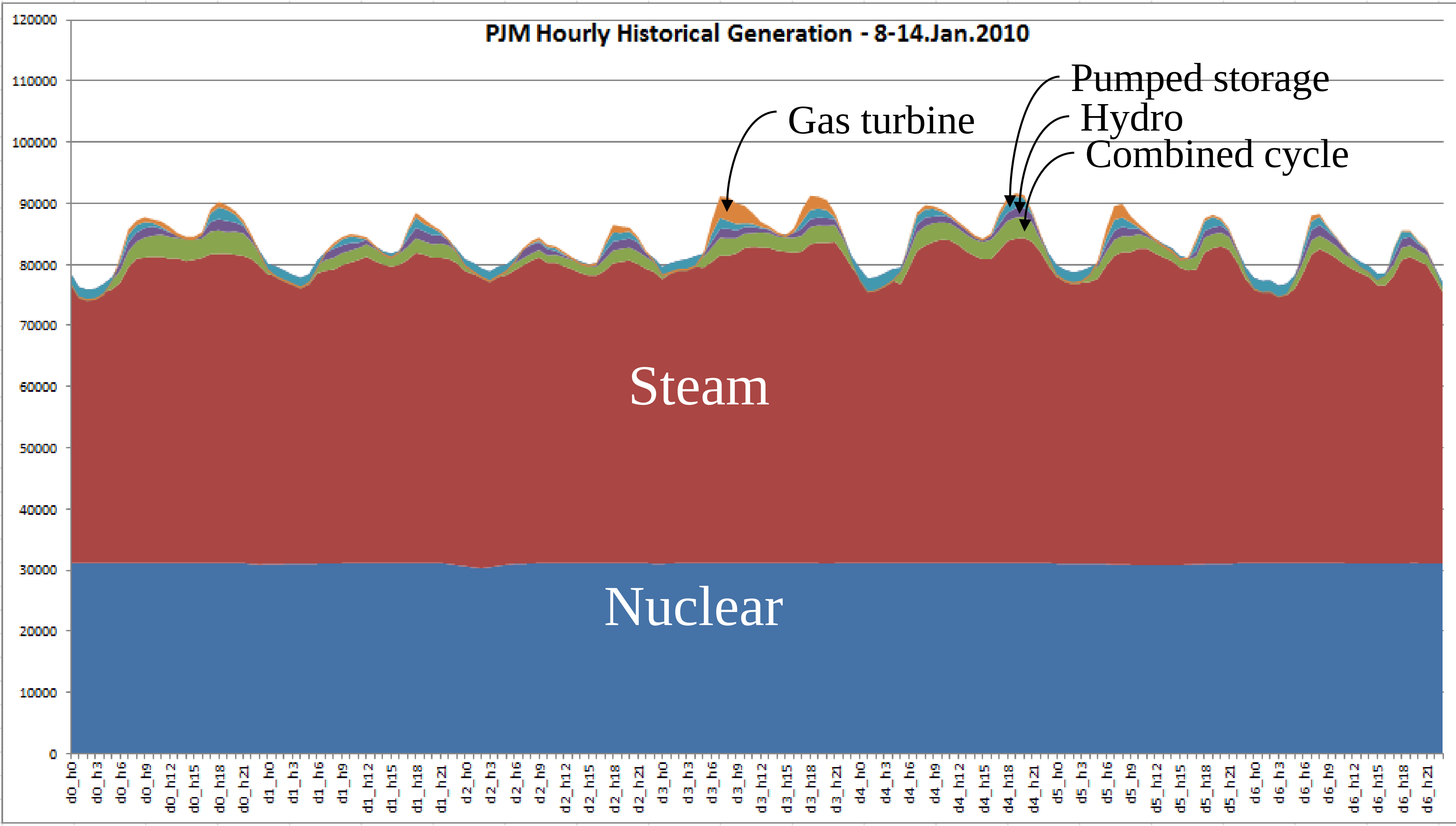


SMART-ISO



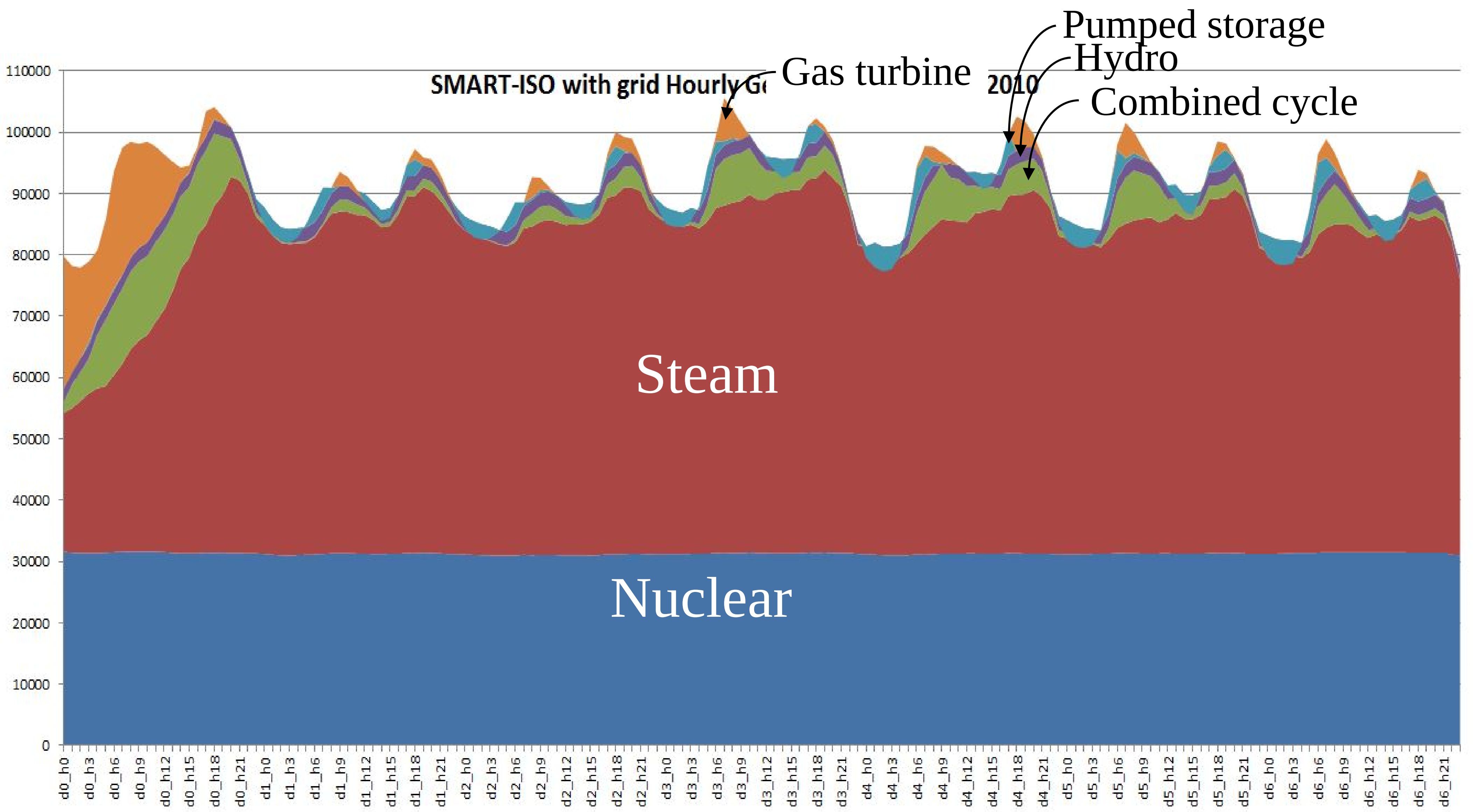
SMART-ISO

- Historical power generation during Jan 8-14 2010



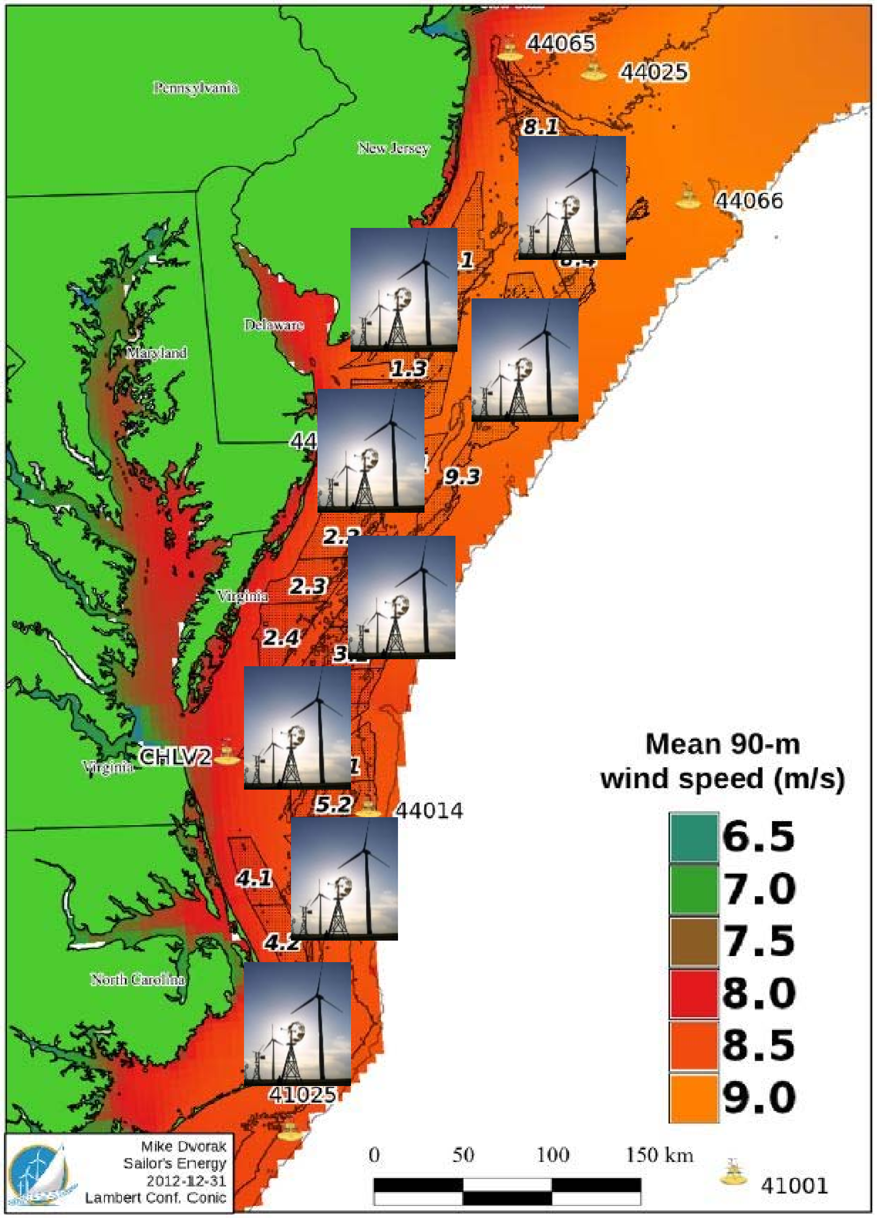
SMART-ISO: Calibration

- Simulated power generation during Jan 8-14 2010



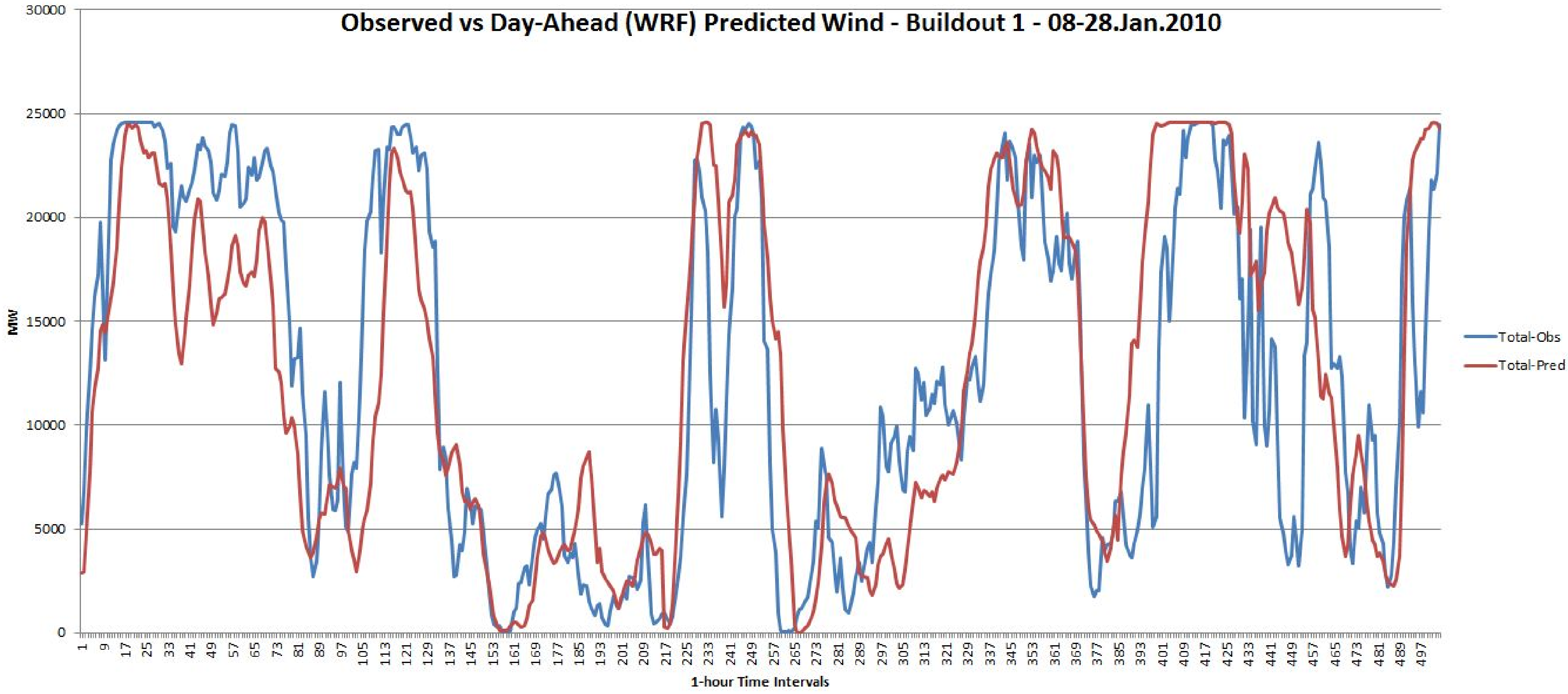
SMART-ISO: Mid Atlantic Offshore Wind

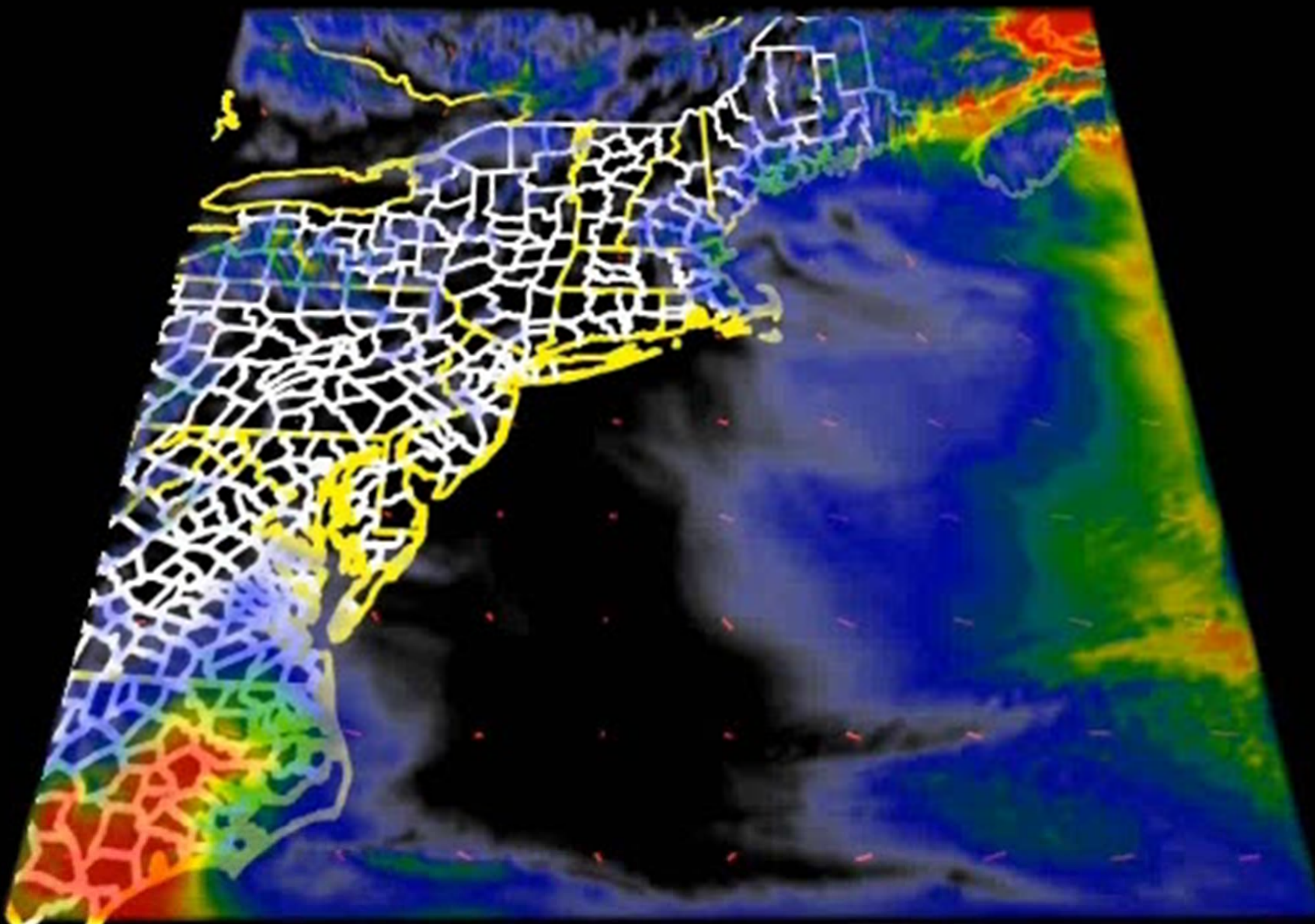
- Mid-Atlantic Offshore Wind Integration and Transmission Study (U. Delaware & partners, funded by DOE)
- 20+ offshore sub-blocks in 4 build-out scenarios:
  - » 1: 27 GW
  - » 2: 49 GW
  - » 3: 64 GW
  - » 4: 77 GW
- Compare to total 70-80 GW usage for entire PJM grid.



SMART-ISO: Mid Atlantic Offshore Wind

- Observed vs WRF predicted wind on Jan 8-28, 2010

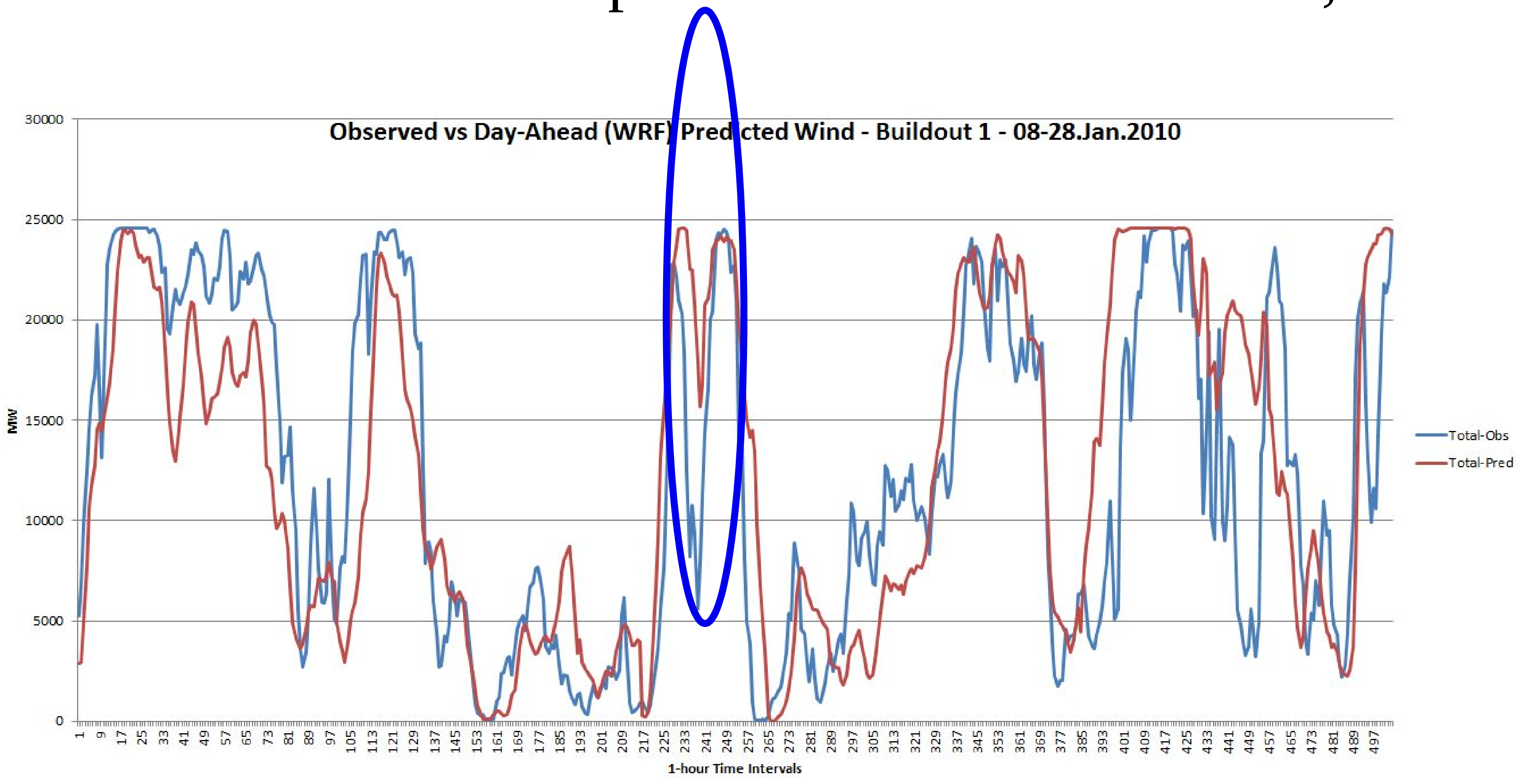




2010-01-17 03:00 EST

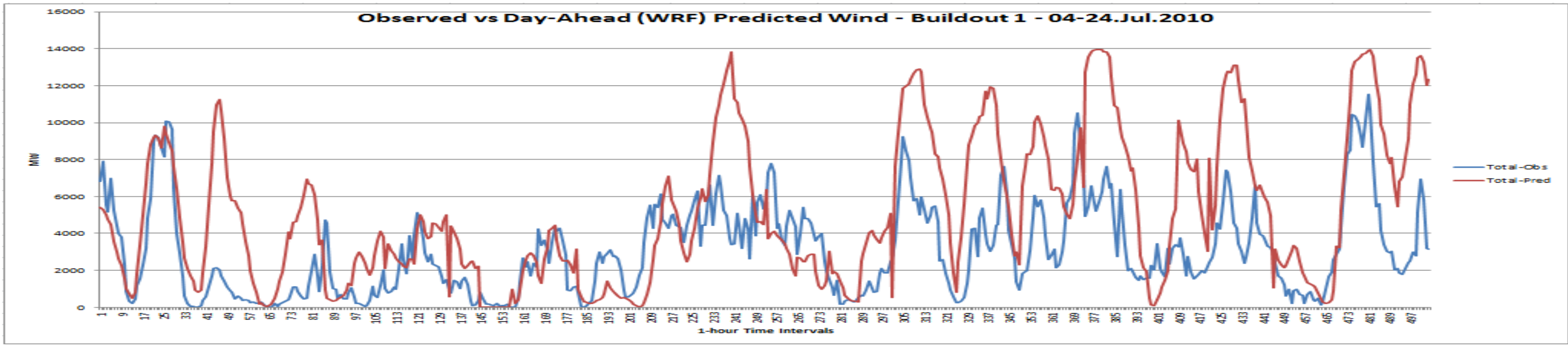
SMART-ISO: Mid Atlantic Offshore Wind

- Observed vs WRF predicted wind on Jan 8-28, 2010



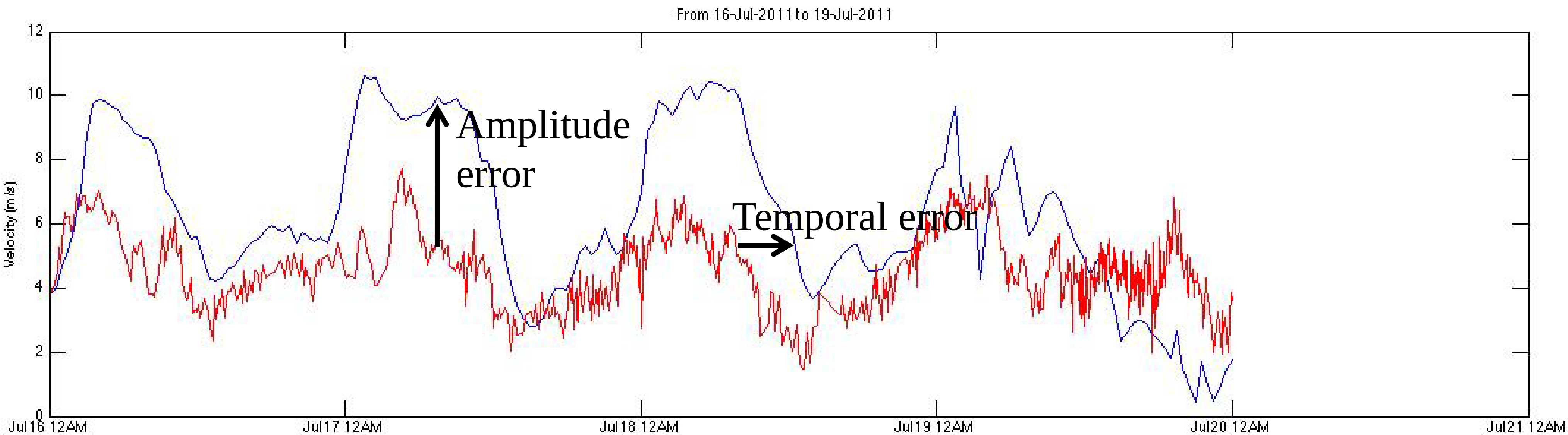
SMART-ISO: Mid Atlantic Offshore Wind

- ❑ July actual vs. forecasted (day ahead)
  - » Plotted same scale as January data



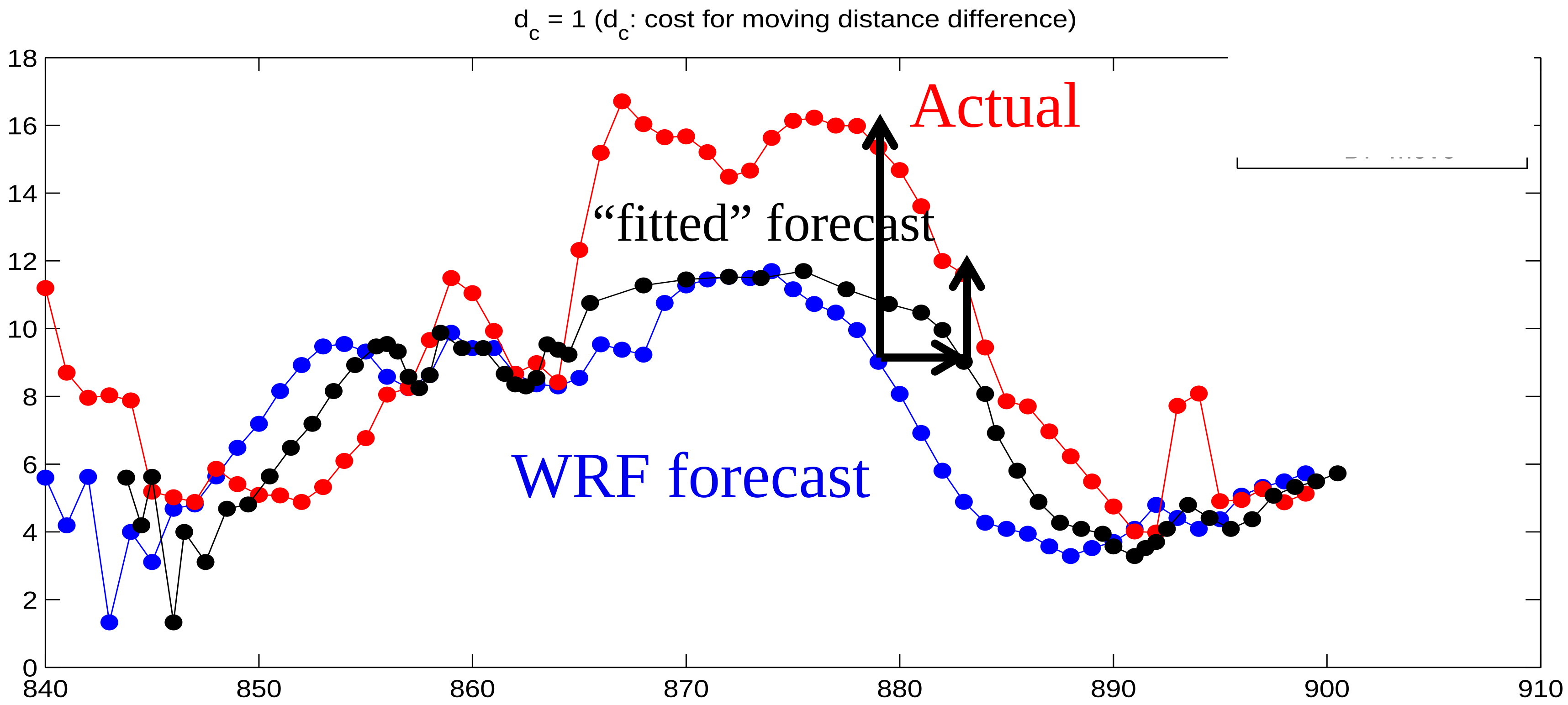
Modeling wind forecast errors

- 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).



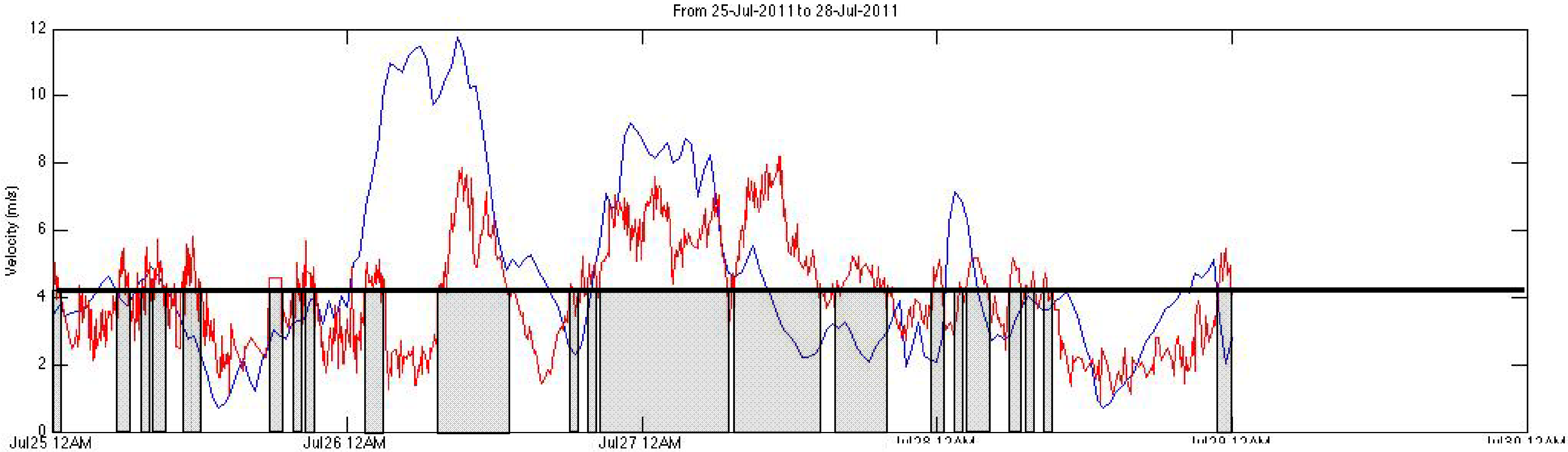
Modeling wind forecast errors

- 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)



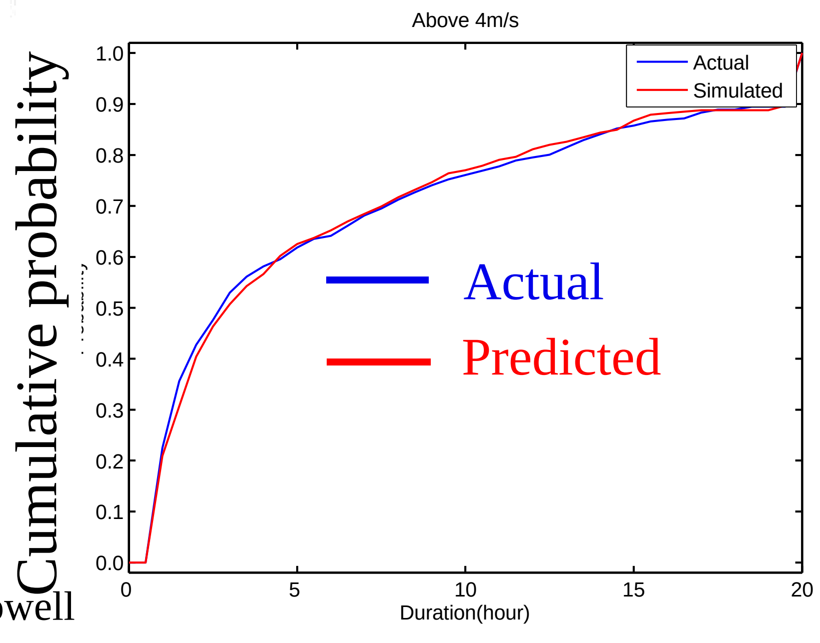
Modeling wind forecast errors

- The level crossing tests



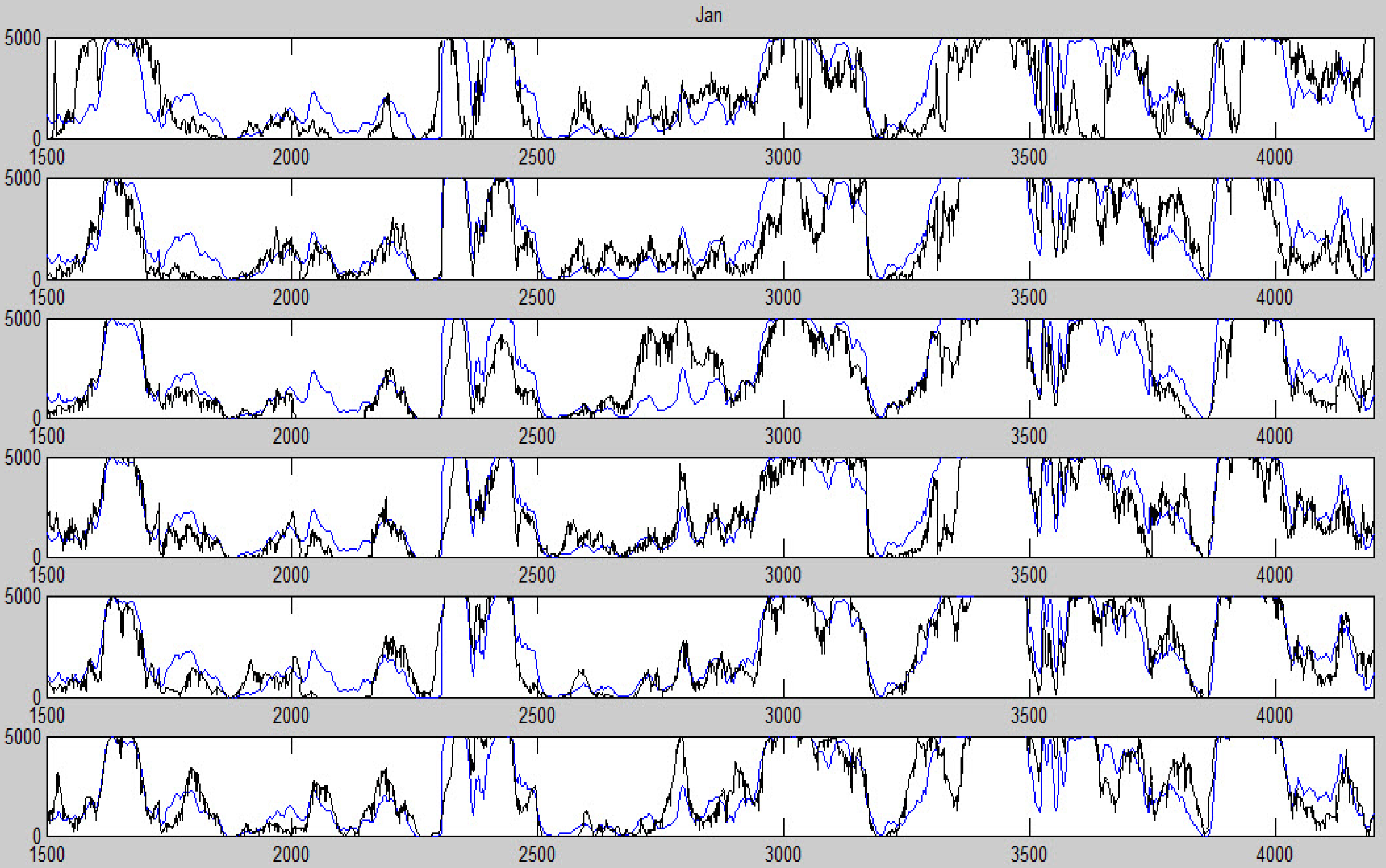
» Using temporal adjustments, we get a very accurate match with historical level crossing distributions.

» Joint research with Prof. Elie Bou-Zeid and Jinzhen Jin (CEE)



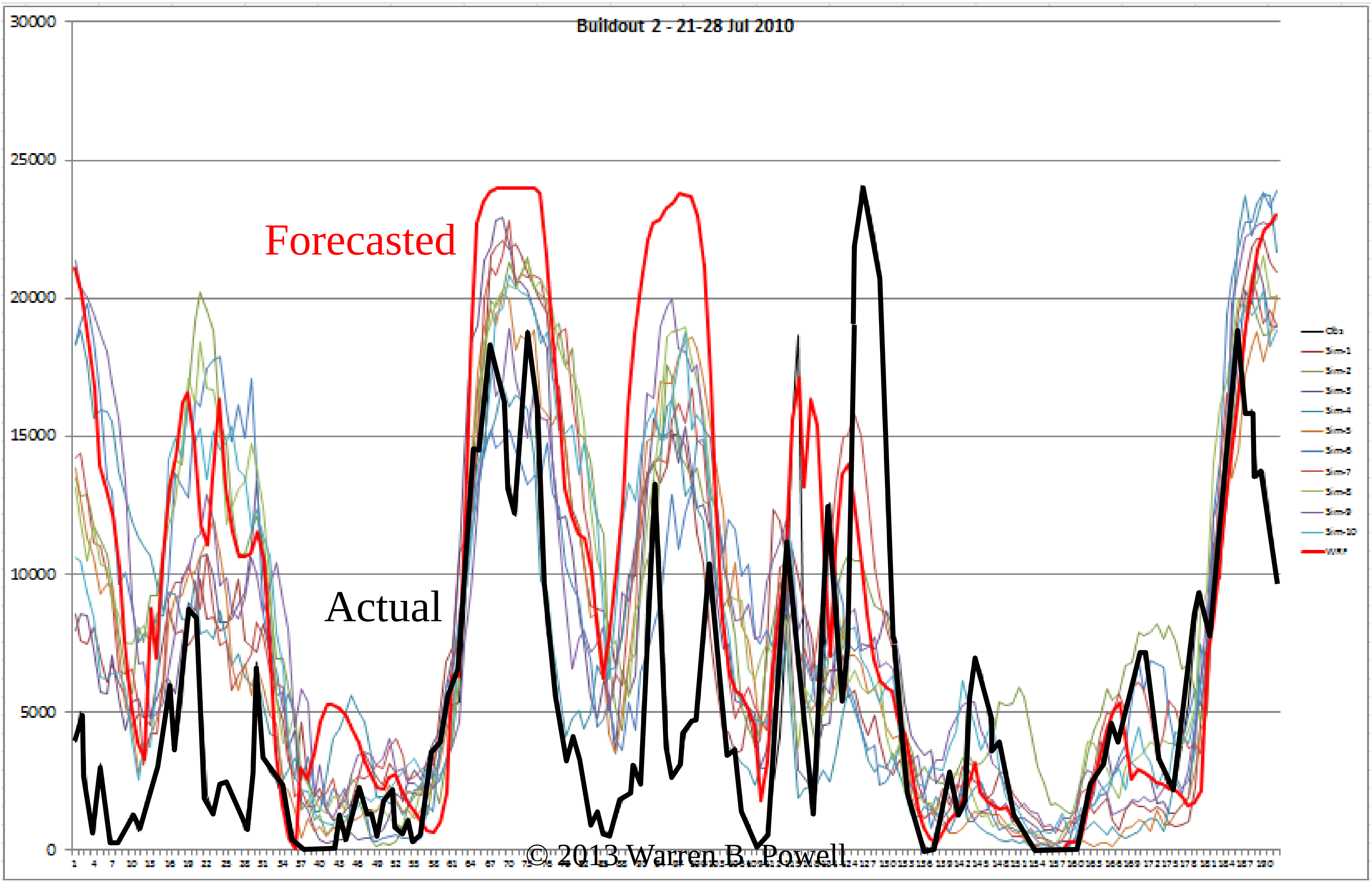
SMART-ISO: Mid Atlantic Offshore Wind

- Sample paths from stochastic model

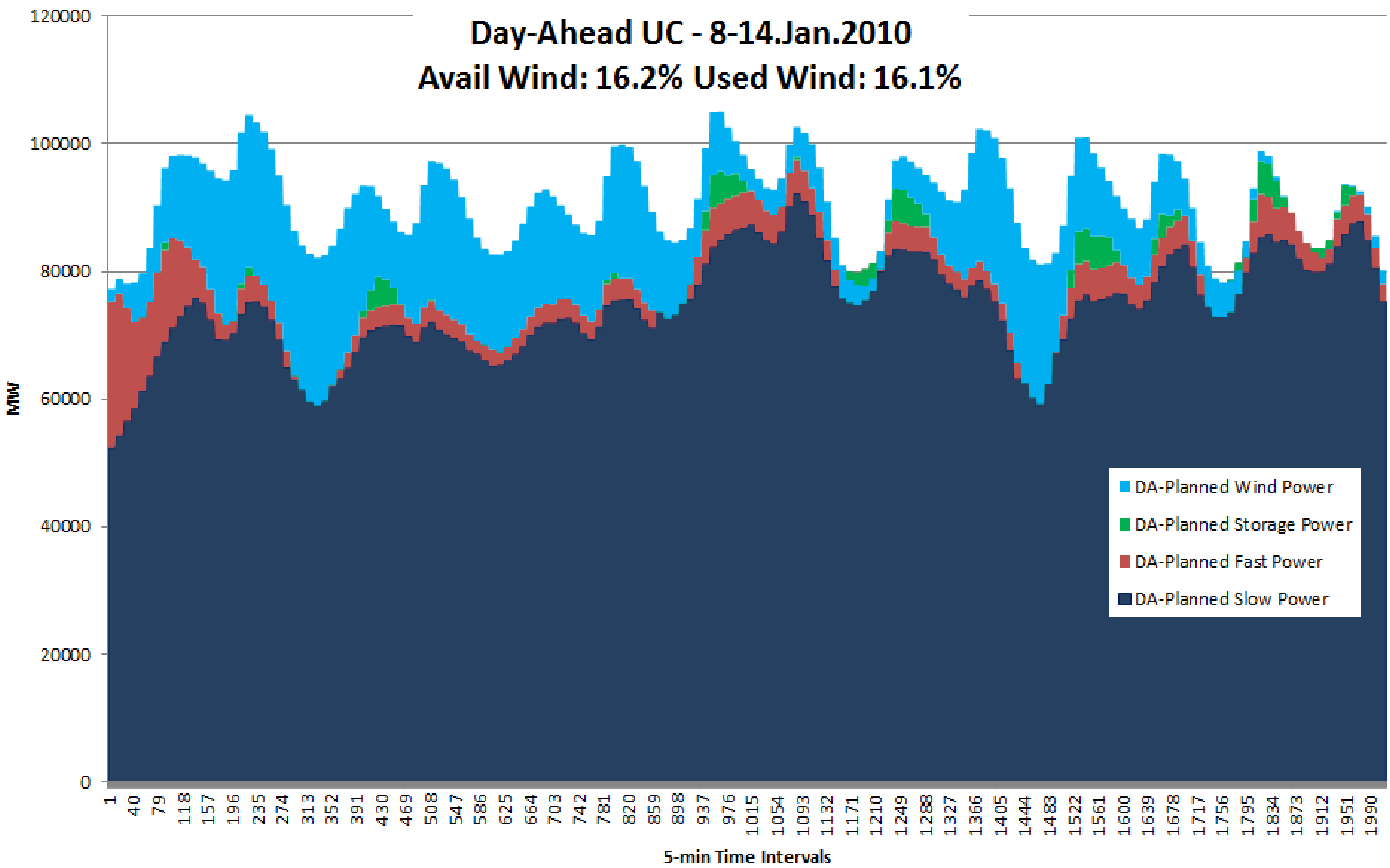


SMART-ISO: Mid Atlantic Offshore Wind

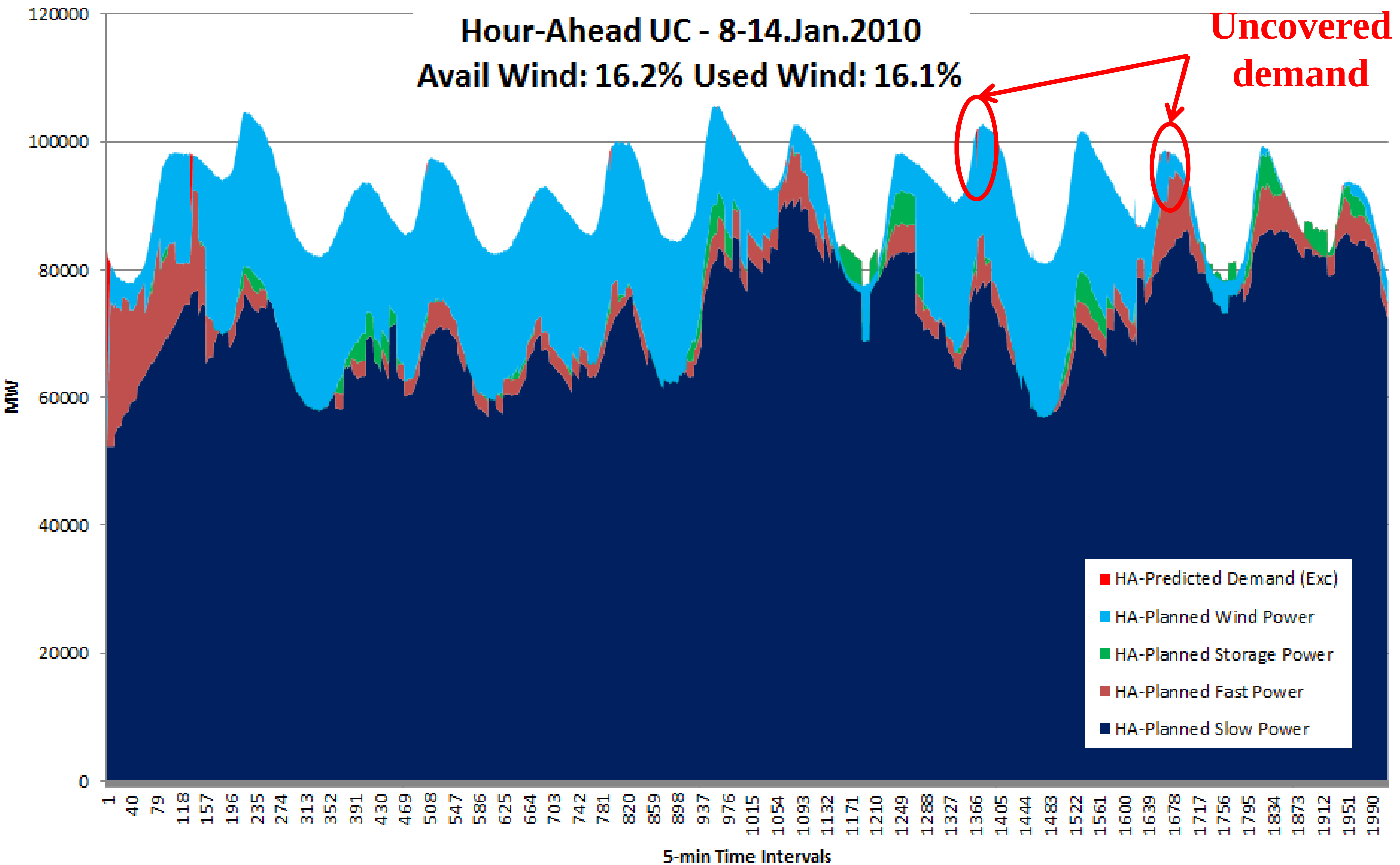
- Sample paths from stochastic model



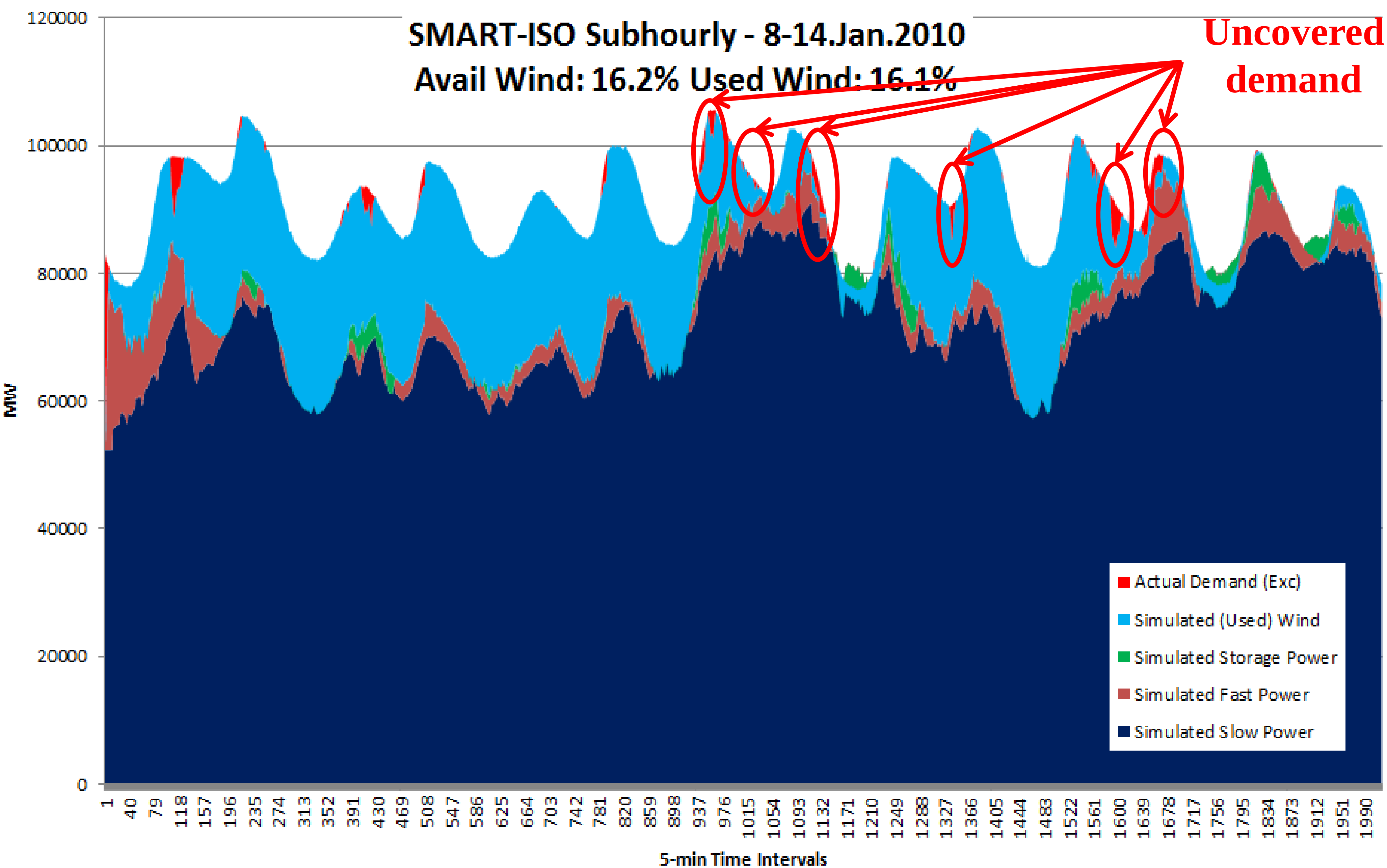
SMART-ISO: Mid Atlantic Offshore Wind



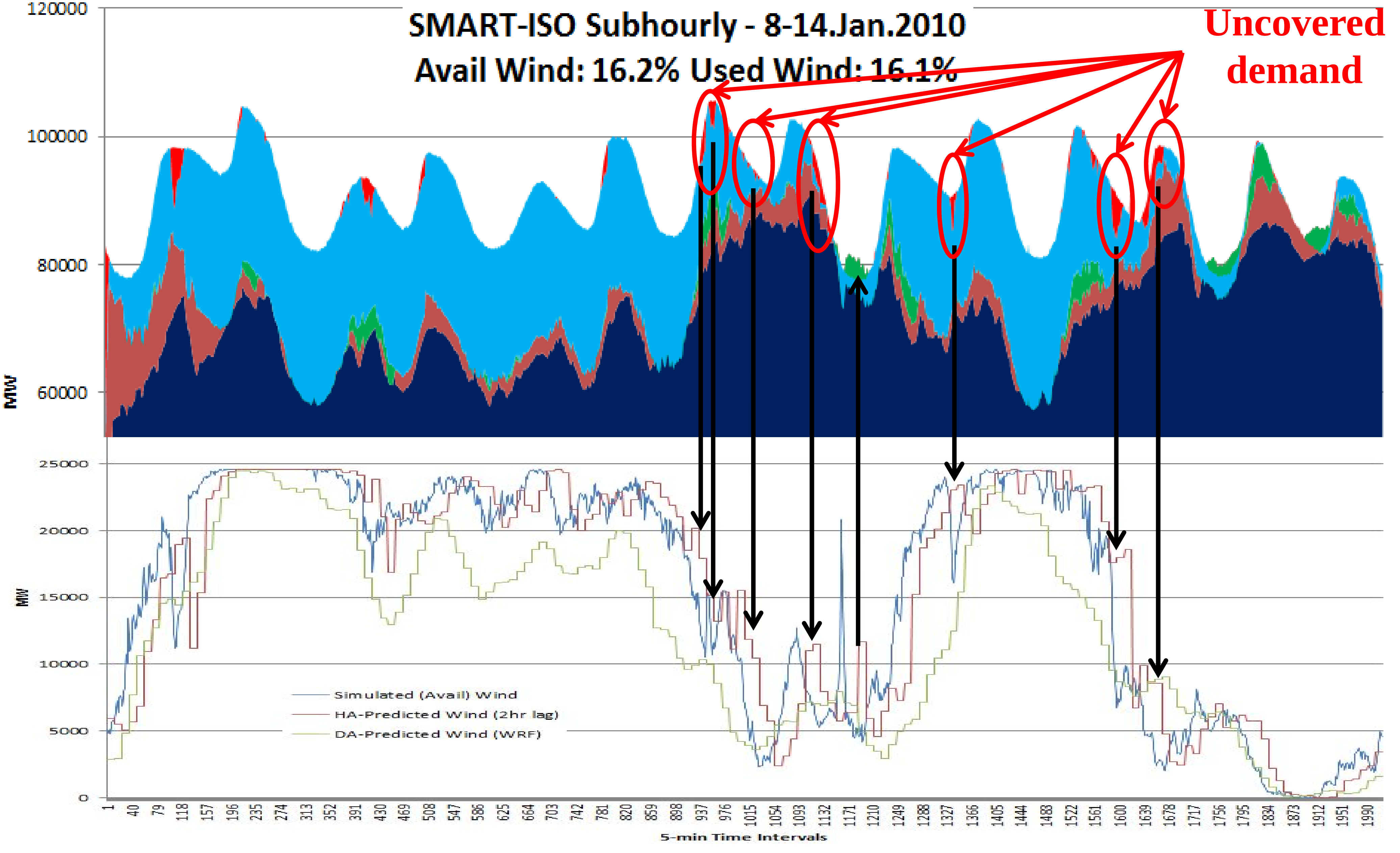
SMART-ISO: Mid Atlantic Offshore Wind



SMART-ISO: Mid Atlantic Offshore Wind



SMART-ISO: Mid Atlantic Offshore Wind



The stochastic unit commitment problem

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

$$F_t(S_t|\theta) = \min_{\substack{(x_{tt'})_{t'=1,\dots,24} \\ \pi}} \mathbb{E} \sum_{t'=1}^{24} C(x_{tt'}, Y_{t'}^{\pi}(S_{tt'}))$$

$$x_{t,t'}^{\max} - x_{t,t'} \geq \theta L_{tt'}$$

Reserve must be a fraction of the load

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

  - The challenge now is to adaptively estimate the ramping constraints $\theta_{tt'}$, and the policies $Y^{\pi}(S_{tt'})$.

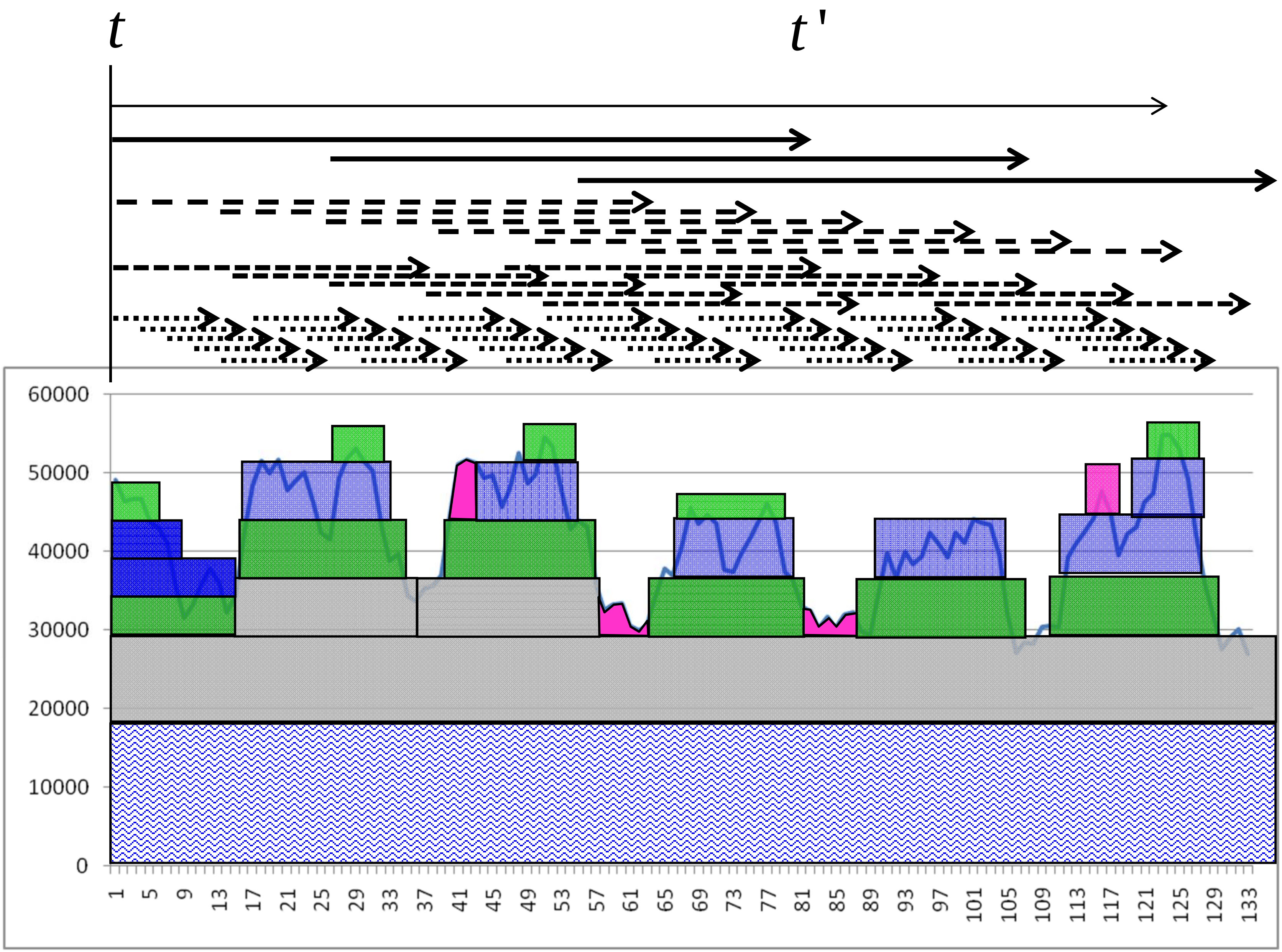
The stochastic unit commitment problem

- Our hybrid policy
  - » The decision $x_{tt'}$ is constrained by time-dependent lower bounds $\theta_{tt'}$ on the amount of fast ramping capacity, which are adaptively updated during the simulation.
  - » The policy $Y^{\pi}_{t'}(S_{tt'})$ is constrained by the solution x_t .
  - » Updates to $\theta_{tt'}$ are based on stochastic gradients which capture their impact on both x_t and on $y_{tt'} = Y^{\pi}_{t'}(S_{tt'})$.

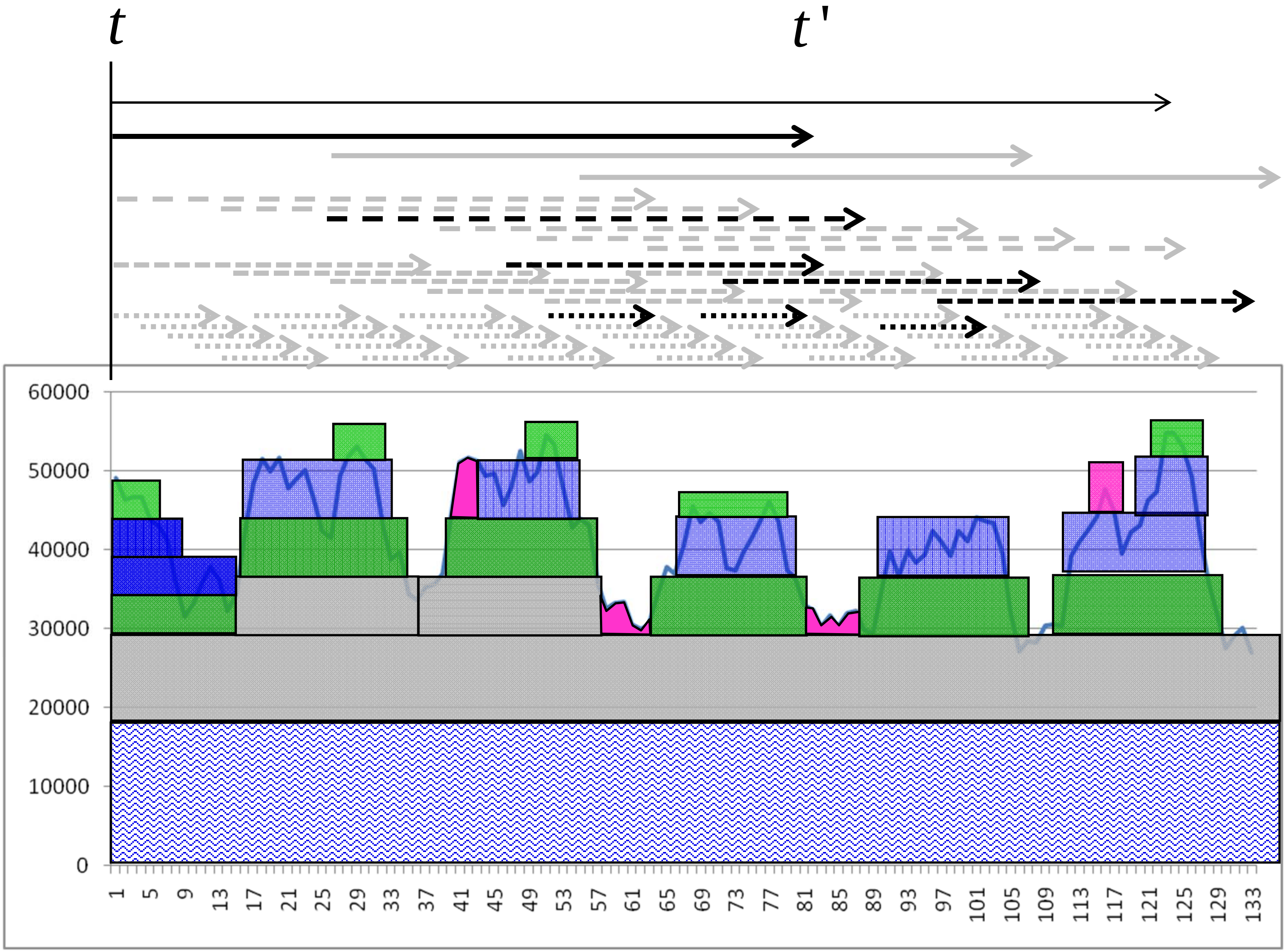
$$\frac{dF(S_t\,|\,\theta)}{d\theta_{tt'}} = \frac{dF(S_t\,|\,\theta)}{dx_{tt'}}\frac{dx_{tt'}}{d\theta_{tt'}} + \frac{dF(S_t\,|\,\theta)}{dy_{tt'}}\frac{dY^{\pi}_{t'}(S_{tt'})}{dx_{tt'}}\frac{dx_{tt'}}{d\theta_{tt'}}$$

  - » Parameters that determine the behavior of $Y^{\pi}_{t'}(S_{tt'})$ are updated in a similar way.
  - » This produces a *nested, adaptive policy* which requires solving sequences of deterministic problems.

The nesting of decisions



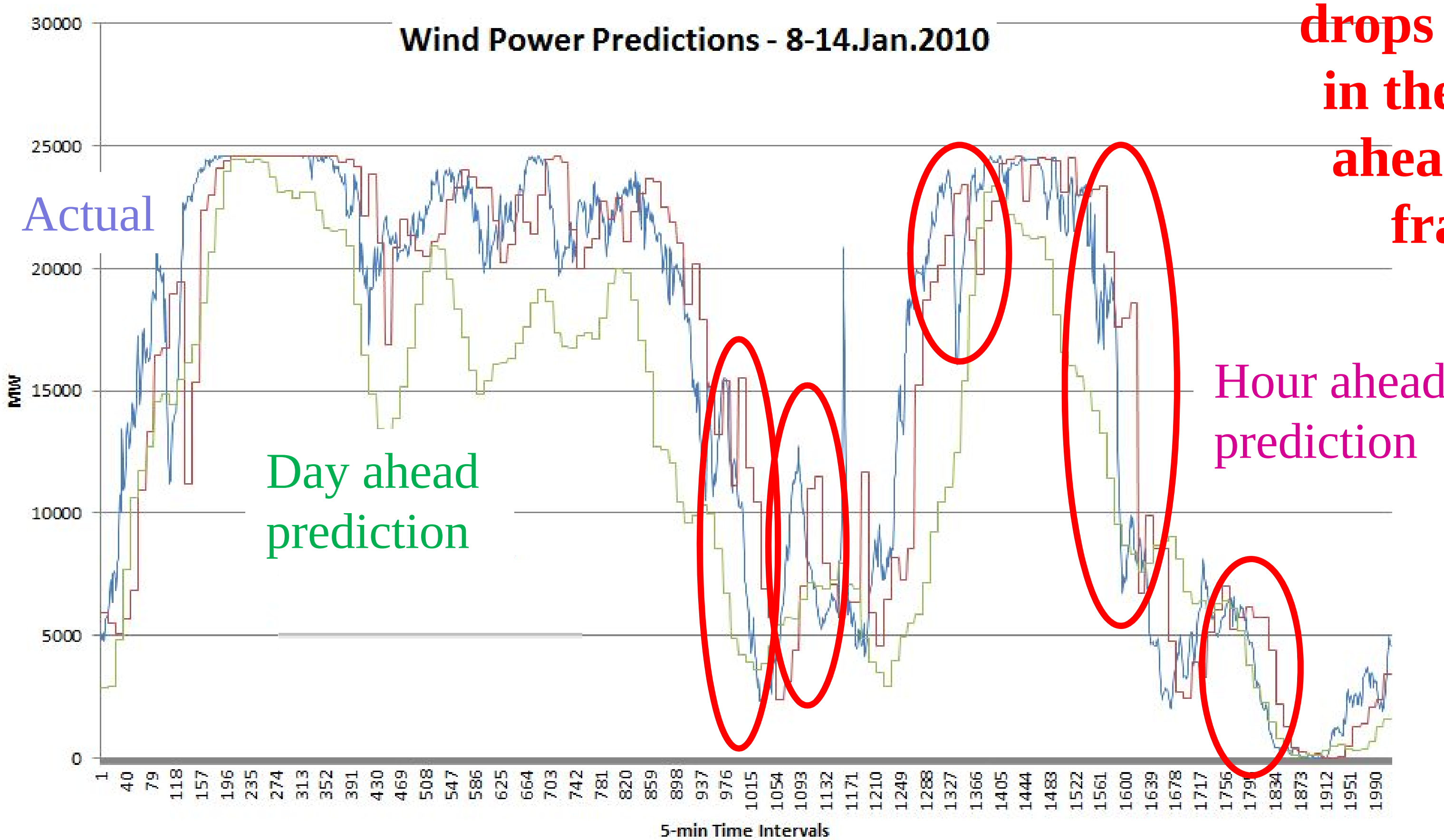
The nesting of decisions



SMART-ISO: Mid Atlantic Offshore Wind

- Wind build-out 1 – forecasts

Difficult to forecast precipitous drops in wind in the hour-ahead time frame

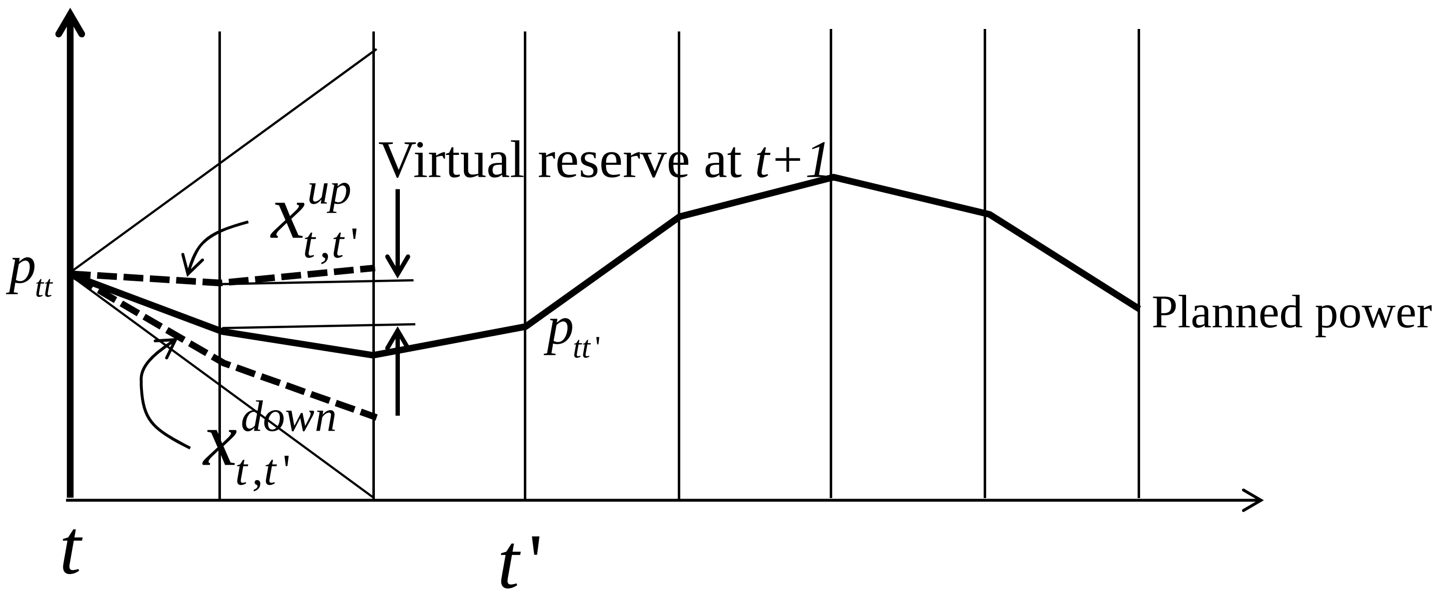


The stochastic unit commitment problem

- Observations
  - We encountered the most difficulty from forecast errors in the hour-ahead model.
  - Ensuring enough reserve capacity did not provide sufficient protection against variations in wind.
  - The problem is that generators ramp at different rates. We may have enough reserve capacity, but if the ramp rates are not fast enough, we cannot access it in time.
  - Simple idea: require a certain level of fast ramping capacity.
  - More sophisticated idea: nested reserve capacity management (due to Prof. Boris Defourny)

The stochastic unit commitment problem

- Ramping reserve constraints



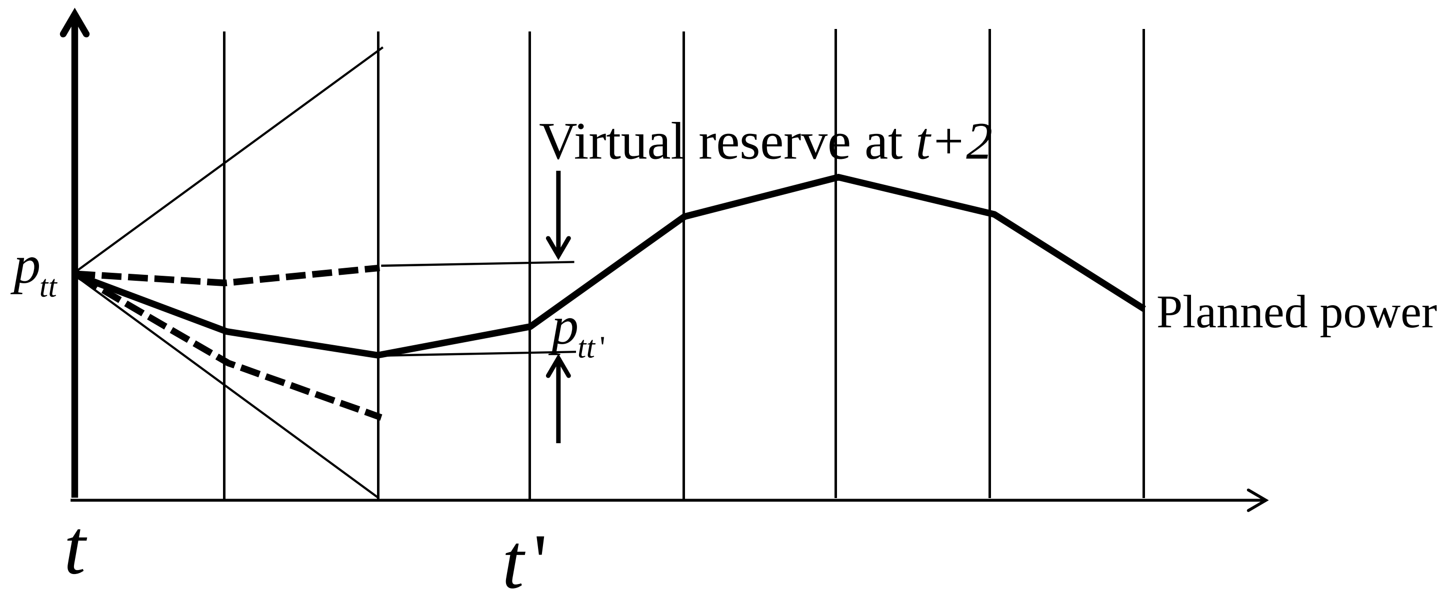
$$p_{tt'} = \text{Planned power level for time } t' \text{ when planning at time } t$$

$$x^{up}_{t,t',t''} = \text{"Virtual" up-reserve planned for time } t'', \text{ for nested lookahead indexed at time } t' \text{ (within lookahead for time } t\text{).}$$

$$x^{down}_{t,t',t''} = \text{"Virtual" down-reserve planned for time } t'', \text{ for nested lookahead indexed at time } t' \text{ (within lookahead for time } t\text{).}$$

The stochastic unit commitment problem

- Ramping reserve constraints



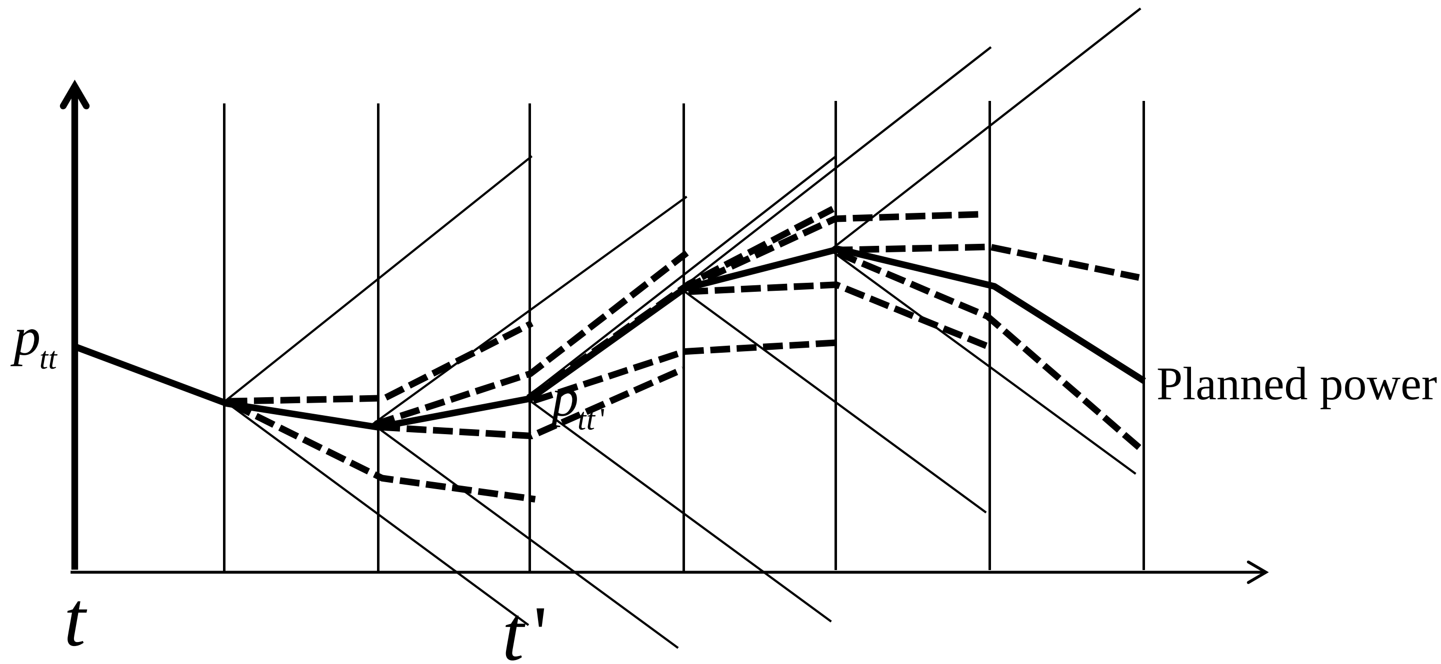
$$\left(x^{up}_{t,t',t'+\Delta t,g} - p_{t,t',g}\right) = \text{Up-ramping reserve for generation } g$$

$$\left(x^{up}_{t,t'+1,t'+1+\Delta t,g} - p_{t,t',g}\right) - \left(x^{up}_{t,t',t'+\Delta t,g} - p_{t,t',g}\right) = \text{Change in up-ramping reserve for } g$$

$$\sum_g \left(x^{up}_{t,t'+1,t'+1+\Delta t,g} - p_{t,t',g}\right) - \left(x^{up}_{t,t',t'+\Delta t,g} - p_{t,t',g}\right) \geq \theta^{up} L_t \quad \text{Systemwide change in up-ramping}$$

The stochastic unit commitment problem

- Ramping reserve constraints



» We impose systemwide up- and down- ramping constraints for *each* of the nested lookahead models.

» This is all solved within a single, “deterministic” lookahead model solved as an integer program….

» …. a very large integer program.

The stochastic unit commitment problem

- The day-ahead unit commitment model
  - » We solve a *cost function approximation*

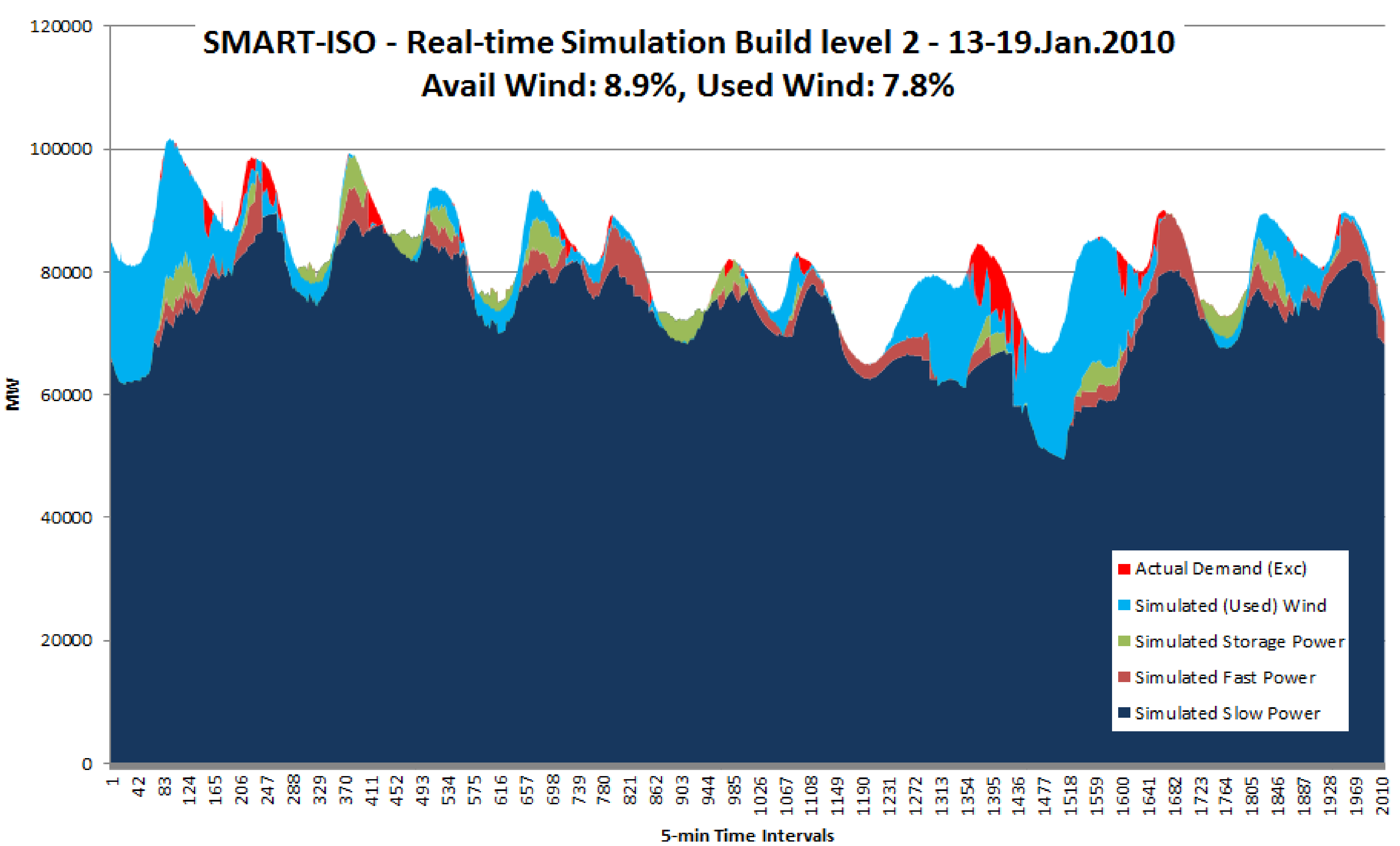
$$F_t(S_t \mid \theta^{up}, \theta^{down}) = \min_{\substack{(x_{t,t'})_{t'=1,\ldots,24} \\ (y_{t,t'})_{t'=1,\ldots,24}}} \sum_{t'=t}^{t+48} C(x_{t,t'}, y_{t,t'})$$

  - » Subject to:
    - Generator constraints
    - Demand constraints
    - DC power flow constraints
    - Up and down ramping reserve constraints:

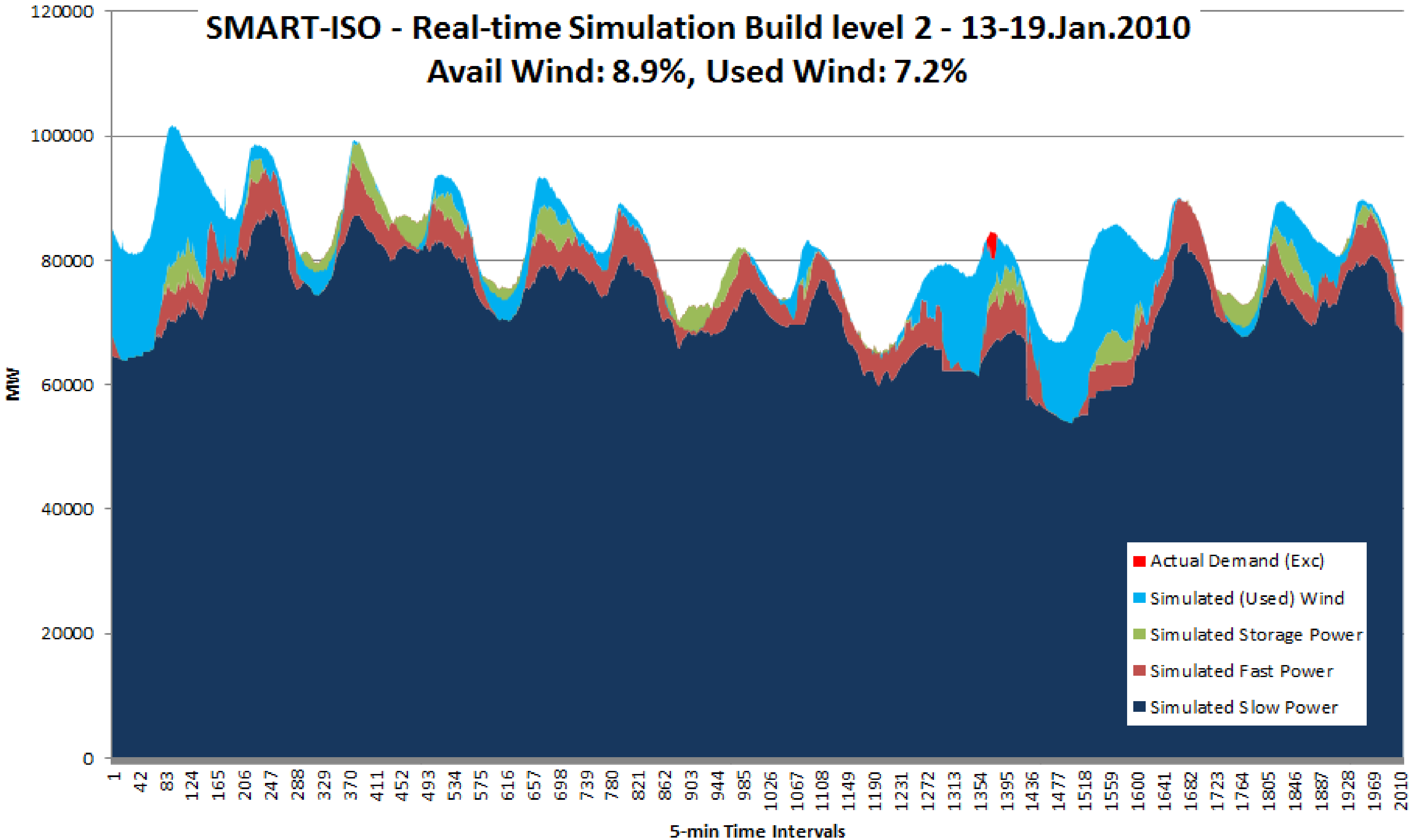
$$\sum_g \left(x^{up}_{t,t'+1,t'+1+\Delta t,g} - p_{t,t',\mathrm{g}} \right) - \left(x^{up}_{t,t',t'+\Delta t,g} - p_{t,t',\mathrm{g}} \right) \geq \theta^{up} L_t$$ Systemwide change in up-ramping

$$\sum_g \left(p_{t,t',\mathrm{g}} - x^{down}_{t,t'+1,t'+1+\Delta t,g} \right) - \left(p_{t,t',\mathrm{g}} - x^{down}_{t,t',t'+\Delta t,g} \right) \geq \theta^{down} L_t$$ Systemwide change in down-ramping

SMART-ISO: Mid Atlantic Offshore Wind



SMART-ISO: Mid Atlantic Offshore Wind



SMART-ISO website

http://energysystems.princeton.edu/smartiso.htm

SMART-ISO

SMART-ISO is a stochastic, multiscale model of grid operations under development at PENSA, which is currently being built around the full PJM grid. The purpose of this website is to provide ongoing documentation of what we have accomplished, what we are working on and features that we hope to develop in the near future. (If you have been to this website before, be sure to hit your refresh button to ensure you have the latest version.)

When the model settles down, we will begin providing indications of "what is new." For now (summer, 2012) the model is going through rapid evolution and we ask for everyone's patience.

A video of SMART-ISO (new! August 28, 2012)

Overview of SMART-ISO

Studies (ongoing and anticipated)

Features (current and planned) (updated August 28, 2012)

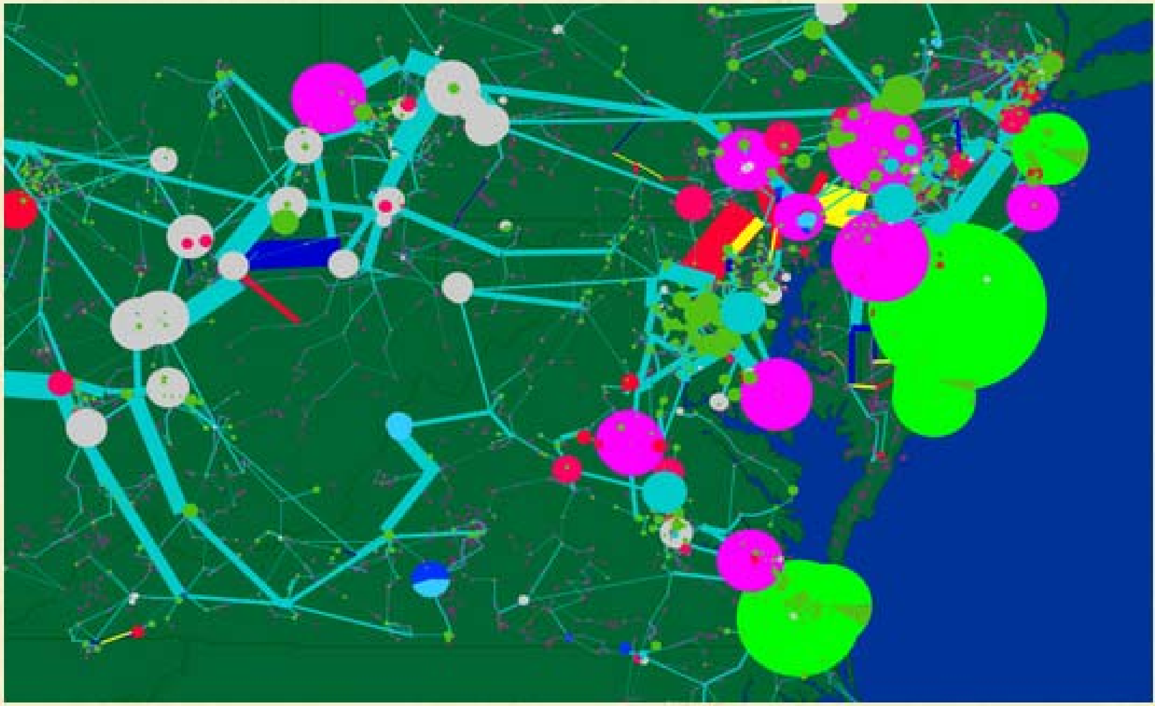
Supporting data (updated August 28, 2012)

System components (updated August 28, 2012)

Handling uncertainty

Documentation

The SMART-ISO development team



Lecture outline

- Types of uncertainty
- Modeling stochastic, dynamic systems
- Optimizing energy storage
  - Using Bellman error minimization
  - Using policy search and optimal learning
- SMART-ISO – Robust unit commitment using a lookahead policy
- Observations

Observations

- *The real problem is not just variability, but uncertainty. Our ability, or inability, to forecast an event is critical.*
- *The best way to solve a problem under uncertainty depends on the structure of the problem. Even small variations can fundamentally change the algorithmic strategy.*
- *Getting verifiable, high quality solutions to even fairly simple problems is astonishingly difficult. Just because you have a method that provides a number, it does not mean it is a good number!*
- *Applications in energy introduce a rich set of challenges that go beyond known algorithms.*