Storage problems

- This week's lectures will consider a series of "storage" problems of increasing complexity
  - » Managing a water reservoir – This is a simple problem with a scalar state and control problem similar to our budgeting problem, but now it is stochastic.
  - » Managing cash balance for a mutual fund – Similar to the water reservoir problem, but now the state variable is a vector.
  - » Purchasing OJ futures contracts – Here we will make decisions now to commit to purchase resources in the future, but excess purchases cannot be held (similar to the newsvendor problem).
  - » Modeling natural gas forward contracts – Similar to the OJ futures contracts, but now excess product can be held for the future.

Lecture outline

- Managing a water reservoir
- Managing cash balance in a mutual fund
- Purchasing OJ futures contracts
- Modeling natural gas forward contracts
- The four-step process for modeling stochastic, dynamic programs

Managing a water reservoir

- Problem description:
  - » We would like to manage the water in a reservoir. Our job is to control the release of water from the reservoir in the presence of random inflows.
- Notation

State variable

$$S_t = R_t = \text{Amount of water (gal) in the reservoir at hour } t$$

Decision variable

$$x_t = \text{Rate of release of water (gal/hr), decided at time } t, \text{ between } t \text{ and } t+1$$

$$X^\pi(R_t) = \text{Policy for determining } x_t \text{ given } R_t.$$

Exogenous information process:

$$\hat{R}_{t+1} = \text{Rate of rainfall (gal/hr) between } t \text{ and } t+1.$$

$$P^W(w) = \text{Prob}\left[\hat{R}_{t+1} = w\right] \text{ (assuming we have discretized } \hat{R}_{t+1}\text{).}$$

Managing a water reservoir

Transition function

$$R_{t+1} = R_t - x_t + \hat{R}_{t+1}$$

Objective function

$$C(R_t, x_t) = \text{Contribution earned if we release water at rate } x_t.$$

$$= \sqrt{x_t}$$

$$\max_{\pi \in \Pi} \mathbb{E} \sum_{t=0}^{T} \gamma^t C(R_t, x_t)$$

The objective function encourages holding water during high rain periods to increase the release rate during periods of low rainfall.

Managing a water reservoir

- Choosing a policy
  - » Start with a policy function approximation

    We might propose a simple rule:

$$X^{\pi}(R_t \mid \theta) = \begin{cases} 0 & \text{If } R_t < R^{\min} \\ \min\left\{x^{\max}, \theta_1 (R_t - R^{\min})\right\} & \text{If } R^{\min} < R_t < \theta_2 \\ x^{\max} & \text{If } R_t > \theta_2 \end{cases}$$

    This rule increases the release rate starting with a set minimum $R^{\min}$, but limited by $x^{\max}$.

  - » Then optimize the parameters of the policy:
    - For this application, we might optimize over a single, but very long sample path:

$$\max_{\theta} F^{\pi}(\theta, \omega) = \sum_{t=0}^{T} C\left(R_t(\omega), X^{\pi}(R_t(\omega) \mid \theta)\right)$$

Managing a water reservoir

- Rather than picking what appears to be some arbitrary function (even if it is sensible), we can sometimes use the dynamic programming recursion we first saw with our deterministic budgeting problem:

$$V_t(R_t) = \max_{0 \le x \le R_t} \left(C(R_t, x) + V_{t+1}(R_{t+1}) \right)$$
$$= \max_{0 \le x \le R_t} \left(C(R_t, x) + V_{t+1}(R_t - x) \right)$$

- To solve this we:
  - » Discretized R and x (in increments of 1 percent of reservoir size)
  - » Started with $V_{T+1}(R_{T+1}) = 0$
  - » Stepped backward in time finding each $V_t(R_t)$ for all possible values of R_t.

Managing a water reservoir

- Dynamic programming algorithm for our deterministic budget problem (from the beginning of the course):

Step 0: Initialize $V_{T+1}(R_{T+1}) = 0$ for $R_{T+1} = 0,1,...,100$

Step 1: Step backward $t = T, T-1, T-2,...$

Step 2: Loop over $R_t = 0,1,...,100$

Step 3: Loop over all decisions $0 \le x \le R_t$

Compute $Q_t(R_t, x) = C(R_t, x) + V_{t+1}(R_t - x)$

End Step 3;

Find $V_t^*(R_t) = \max_x Q_t(R_t, x)$

Store $X_t^{\pi^*}(R_t) = \arg\max_x Q(R_t, x)$. (This is our policy)

End Step 2;

End Step 1;

We now have the optimal policy $X_t^{\pi^*}(R_t)$. Or, if we only computed $V_t^*(R_t)$, our policy would be given by

$$X_t^{\pi^*}(R_t) = \arg\max_x \left(C(R_t, x) + V_{t+1}^*(R_t - x) \right)$$

Managing a water reservoir

- Now we want to modify our dynamic programming algorithm for our water reservoir problem. The introduction of uncertain rainfall means that we have to slightly modify our dynamic programming recursion:

$$
\begin{aligned}
V_t(R_t) &= \max_{0 \le x \le R_t} \left(C(R_t, x) + V_{t+1}(R_{t+1}) \right) \\
&= \max_{0 \le x \le R_t} \left(C(R_t, x) + V_{t+1}(R_t - x + \hat{R}_{t+1}) \right)
\end{aligned}
$$

But, $\hat{R}_{t+1}$ is a random variable, so we have to take the expected value of the value function:

$$
\begin{aligned}
V_t(R_t) &= \max_{0 \le x \le R_t} \left(C(R_t, x) + \mathbb{E}\left\{ V_{t+1}(R_t - x + \hat{R}_{t+1}) \mid R_t \right\} \right) \\
&= \max_{0 \le x \le R_t} \left(C(R_t, x) + \sum_{w=0}^{100} V_{t+1}\left(\min\left\{ R^{\max}, R_t - x + w \right\} \right) P^W(w) \right)
\end{aligned}
$$

Managing a water reservoir

- Dynamic programming algorithm with a random variable:

Step 0: Initialize $V_{T+1}(R_{T+1}) = 0$ for $R_{T+1} = 0,1,...,100$

Step 1: Step backward $t = T, T-1, T-2, ...$

Step 2: Loop over $R_t = 0,1,...,100$

Step 3: Loop over all decisions $0 \le x_t \le R_t$

Step 4: Take the expectation over all rainfall levels (also discretized):

$$\text{Compute } Q(R_t, x_t) = C(R_t, x_t) + \sum_{w=0}^{100} V_{t+1}(\min\left\{R^{\max}, R_t - x + w\right\}) P^W(w)$$

End step 4;

End Step 3;

Find $V_t^*(R_t) = \max_{x_t} Q(R_t, x_t)$

Store $X_t^{\pi^*}(R_t) = \arg\max_{x_t} Q(R_t, x_t)$. (This is our policy)

End Step 2;

End Step 1;

We now have the optimal policy $X_t^{\pi^*}(R_t)$. Or, if we only computed $V_t^*(R_t)$, our policy would be given by

$$X_t^{\pi^*}(R_t) = \arg\max_{x_t} \left(C(R_t, x_t) + \sum_{w=0}^{100} V_{t+1}(\min\left\{R^{\max}, R_t - x + w\right\}) P^W(w) \right)$$

Managing a water reservoir

- Notes:
  - » We now have four nested loops:
    - Backward over time
    - Over all states R_t
    - Over all actions (decisions) x_t
    - Over all realizations of the random variable $\hat{R}_{t+1}$
  - » You have to choose an appropriate level of discretization. We assumed discretization of the reservoir to the nearest one percent, but perhaps you may need to do .1 percent (or 5 percent).
  - » We can store the policy as a lookup table (this is a form of "policy function approximation"), or just store the value function (which makes it a "policy that depends on a value function approximation"). For this problem, the approximation arises because we discretized the problem.

Lecture outline

- Managing a water reservoir
- Managing cash balance in a mutual fund
- Purchasing OJ futures contracts
- Modeling natural gas forward contracts
- The four-step process for modeling stochastic, dynamic programs

Managing cash in a mutual fund

- Imagine that you are the mutual fund manager in the email we showed earlier in the course.
- You have to decide how much cash you want in your mutual fund. This is a *lot* like the water reservoir problem, with a few twists:
  - » You can move money from cash to equities to obtain higher returns, or back. So the "flow" variable x can be positive or negative.
  - » The money invested in equities earns a return that varies randomly over time.
  - » The money held in cash earns interest from money markets that also varies randomly over time.

Managing cash in a mutual fund

- The model:

State variables:

p_t = Current value of a stock index (e.g. S&P 500)

I_t = Current interest rate

R_t = Amount held in cash; R_t^s = Amount held in stocks

$S_t = \left(p_t, I_t, R_t, R_t^s\right)$

Decision variable

x_t = Amount moved from stocks to cash (if >0) or cash to stocks

$X^\pi(S_t)$ = Policy for moving cash into stocks/back into cash.

Exogenous information

$\hat{p}_{t+1}$ = Change in stock index between t and $t+1$

$\hat{I}_{t+1}$ = Change in interest rates between t and $t+1$

$\hat{R}_{t+1}$ =Net deposits (if >0) or withdrawals (if <0)

$W_{t+1} = \left(\hat{p}_{t+1}, \hat{I}_{t+1}, \hat{R}_{t+1}\right)$

Managing cash in a mutual fund

- The model:

Transition function

$p_{t+1} = p_t + \hat{p}_{t+1}$ Stock index (e.g. S&P 500)

$I_{t+1} = I_t + \hat{I}_{t+1}$ Interest rates

$R_{t+1} = R_t + x_t + \hat{R}_{t+1}$ Amount in cash

$R^s_{t+1} = R^s_t - x_t$ Amount in stocks

Objective function

$C(S_t, x_t) =$ Return from stock market and interest from mutual fund, minus transaction costs

$$\max_\pi \mathbb{E} \sum_{t=0}^{T} \gamma^t C(S_t, X_t^\pi(S_t))$$

Managing cash in a mutual fund

- Let's try using our dynamic programming recursion

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

$$S_t = \begin{bmatrix} p_t \\ I_t \\ R_t \\ R_t^s \end{bmatrix} \qquad W_{t+1} = \begin{bmatrix} \hat{p}_{t+1} \\ \hat{I}_{t+1} \\ \hat{R}_{t+1} \end{bmatrix}$$

  - » The challenge – The state variable now has *five* dimensions, and the information variable has *three* dimensions.
  - » Extra dimensions have to be handled using multiple, nested loops.
  - » Computationally, this would take forever.

Managing cash in a mutual fund

- DP recursion for mutual fund problem:

Step 0: Initialize $V_{T+1}(S_{T+1}) = 0$ for $\left(p_{T+1}, I_{t+1}, R_{T+1}, R^s_{T+1}\right) = \left(0...100,\ 0...100,\ 0...100,\ 0...100\right)$

Step 1: Step backward $t = T, T-1, T-2, ...$

Step 2: Loop over S_t (need to do this as four loops, one for each dimension)

Step 3: Loop over all decisions x_t (a problem if x_t is a vector)

Step 4: Take the expectation over each random dimension:

$$\text{Compute } Q(S_t, x_t) = C(S_t, x_t) + \sum_{w_1=0}^{100}\sum_{w_2=0}^{100}\sum_{w_3=0}^{100} V_{t+1}\left(S^M\left(S_t, x_t, W_{t+1} = (w_1, w_2, w_3)\right)\right) P^W(w_1, w_2, w_3)$$

End step 4;

End Step 3;

Find $V_t^*(S_t) = \max_{x_t} Q(S_t, x_t)$

Store $X_t^{\pi^*}(S_t) = \arg\max_{x_t} Q(S_t, x_t)$. (This is our policy)

End Step 2;

End Step 1;

We now have the optimal policy $X_t^{\pi^*}(S_t)$. We can use $V_t^*(S_t)$ to represent the policy as

$$X_t^{\pi^*}(S_t) = \arg\max_{x_t}\left(C(S_t, x_t) + \sum_{w_1=0}^{100}\sum_{w_2=0}^{100}\sum_{w_3=0}^{100} V^*_{t+1}\left(S^M\left(S_t, x_t, W_{t+1} = (w_1, w_2, w_3)\right)\right) P^W(w_1, w_2, w_3)\right)$$

Managing cash in a mutual fund

- Notes:
  - This application introduces a problem known widely as the *curse of dimensionality*. Depending on the degree of discretization, three dimensions tend to be the upper limit of what we can handle.
  - It creates a problem when we try to calculate a value function in our dynamic program.
  - It also creates a problem if we try to create a policy that maps a discrete state to an action.
  - The curse of dimensionality is a problem when we use a *lookup table representation* for either the value function or the policy function.
  - We can overcome the curse of dimensionality if we are willing to approximate either the value function or policy using a *parametric approximation*.
  - There is an entire field dedicated to designing approximations for hard classes of problems. See http://adp.princeton.edu

Lecture outline

- Managing a water reservoir
- Managing cash balance in a mutual fund
- Purchasing OJ futures contracts
- Modeling natural gas forward contracts
- The four-step process for modeling stochastic, dynamic programs

Purchasing OJ futures contracts

- Setting:
  - » In the OJ game, you have the option of purchasing contracts for FCOJ (for example) in year *t*, to be delivered in year *t'*.
  - » The farther you purchase into the future, the lower the price.
  - » But prices vary from year to year. If you buy too much now, you may not be able to take advantage of price dips later.
  - » *Excess contracts cannot be held to the following year.* This is a central feature of this problem (we relax it in our next application).

Purchasing OJ futures contracts

- Futures from decisions spreadsheet for OJ game
  - » Year 2002:

Purchases at the Futures Market(tons) (ORA and FCOJ)

		1997		1998		1999		2000		2001		2002		
Type	Maturity	Price	Quantity	Price	Quantity	Price	Quantity	Price	Quantity	Price	Quantity	Price	Quantity	Sum
ORA Contracts	2002	0.690839	480000	0.718302		0.823148		0.806133		0.855665		Matured Amount:		480000
	2003			0.666955		0.759607	400000	0.796307		0.836728		0.718248		400000
	2004					0.725137	400000	0.77745		0.807833		0.653769		400000
	2005							0.756559	400000	0.745661		0.653937		400000
	2006									0.696338	400000	0.654186		400000
	2007											0.638994	400000	400000
FCOJ Contracts	2002	0.986933	0	0.984678		0.964059	300000	1.038188		1.14383		Matured Amount:		300000
	2003			1.001274		0.977488	300000	1.030097		1.13028		1.316084		300000
	2004					0.926291	300000	0.938946		1.094076		1.22293		300000
	2005							0.871367		1.044974		1.135535		0
	2006									1.019118		1.112037		0
	2007											1.05665		0

  - » Numbers for years 1997-2001 were made in the past.

Purchasing OJ futures contracts

- Notation

x_t = Contract to purchase x_t tons of FCOJ at time t to be delivered at time T.

p_t^f = Price per ton of FCOJ purchased at time t for delivery at time T

p_T^M = Market price at time T,

D_T = Demand at time T,

x_T^{sell} = Amount we sell to the market $= \min\left\{ D_T, \sum_{t=0}^{T-1} x_t \right\}$

- Understanding information:
  - » What is known at time t?
  - » What is random at time t?
  - » At time t=0 (that is, here and now), is x_t known or random (for t>0)?

Purchasing OJ futures contracts

- Assume everything is deterministic (known at time 0):
  - » We would formulate the optimization problem as:

$$\max_{x_0,\ldots,x_{T-1},x_T^{sell}} \left(x_T^{sell} p_T^M - \sum_{t=0}^{T-1} p_t^f x_t \right)$$

where

$$x_T^{sell} \le D_T \qquad \text{Cannot sell more than the market demand}$$

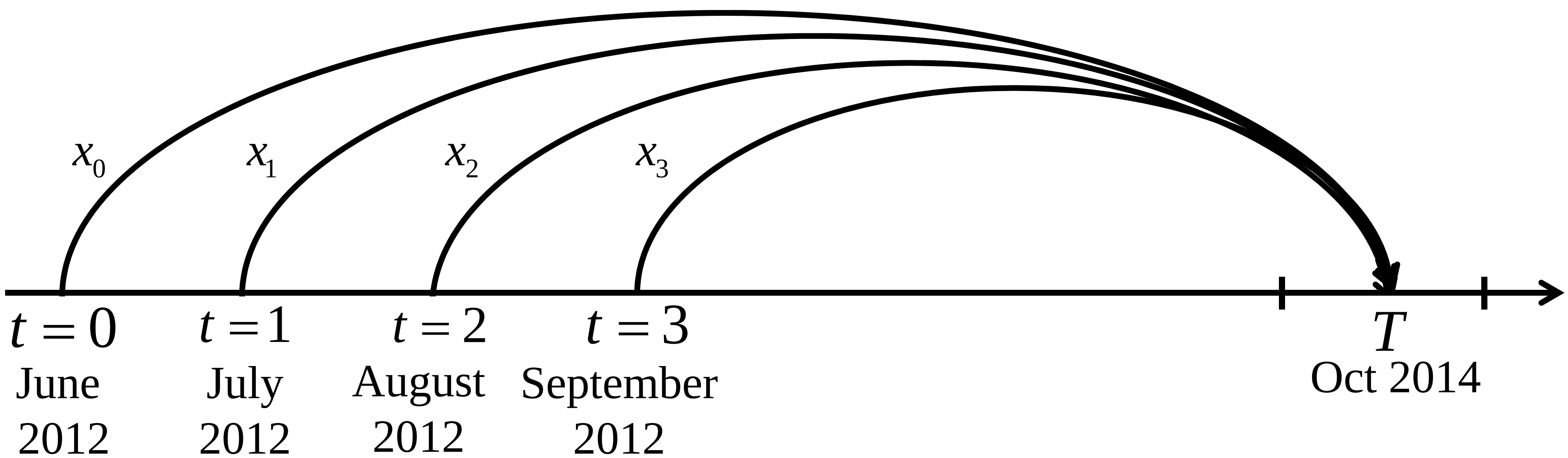
$$x_T^{sell} \le \sum_{t=0}^{T-1} x_t \qquad \text{Cannot sell more than you bought.}$$

$$x_t \ge 0 \quad t = 0,1,\ldots,T-1$$

$$x_T^{sell} \ge 0$$

Purchasing OJ futures contracts

- We can buy contracts at different points in time for a point in time T in the future.
  - » Prices rise on average, but may drop.



Purchasing OJ futures contracts

- Now assume that prices and demands only become known at the time they are indexed.

  This means at at time 0, x_t, p_t^f, p_T^M, and D_T are random variables.

  If we write:

  $$\max_{x_0,\ldots,x_{T-1},x_T^{sell}} \left(x_T^{sell} p_T^M - \sum_{t=0}^{T-1} p_t^f x_t \right)$$

  then we are trying to choose a set of random variables!

- How do we write down our problem???

  » One idea is to fix $x_0, x_1, \ldots, x_{T-1}$ and x_T^{sell} in advance. But this prevents us from adapting to new information. For example, what if the price drops? We would not be able to buy more.

Purchasing OJ futures contracts

- Method for writing down optimization problems in the presence of uncertainty.
  - » Step 1: Assume you are given a *policy* that we are going to call $X_t^\pi(S_t)$. This is a function that returns the decisions x_t for $t = 0,...,T-1$ and, for $t = T$, x_t^{sell}.

    We would then write our objective function as

$$\max_\pi \mathbb{E}\left(X_T^\pi(S_T) p_T^M - \sum_{t=0}^{T-1} p_t^f X_t^\pi(S_t) \right).$$

  - » We are going to deal later with the issue of how to find the best policy (although from the previous lecture, you should have an idea of where this is going).

Purchasing OJ futures contracts

- Method for writing down optimization problems in the presence of uncertainty.
  - Step 2: Fix the policy, and now run a simulation:

$$F^{\pi}(S_0,\omega) = \left(\underbrace{X_T^{\pi}(S_T(\omega))}_{x_T^{sell}} p_T^M - \sum_{t=0}^{T-1} p_t^f \underbrace{X_t^{\pi}(S_t(\omega))}_{x_t} \right).$$

  - To compute this, we assume we are able to *simulate* our way through the problem. If we need a price or demand, we sample it from a distribution. If we need a decision, we turn to our policy.

Purchasing OJ futures contracts

- Method for writing down optimization problems in the presence of uncertainty.
  - » Step 3: Now search for the best policy. This occurs in two stages:
    - First, choose the *type* of policy:
      - – Policy function approximation? For this problem, it means coming up with common sense rules.
      - – Cost function approximation? (not sure how to make this work on this problem)
      - – Policy based on value function approximation? This can work if the state variable is simple. If not, then you need someone with specialized training.
      - – Lookahead (hmm – this could work)
      - – Mixture of the two? (possible, but not in this course)
    - Second, you will have to tune any parameters that are used in each class of policy that you consider.

Purchasing OJ futures contracts

- Let's start with a policy function approximation.
  - » We first have to design a policy function. How about:

    Purchase a fraction θ_0 of the expected demand $\mathbb{E}D_T$ (we assume that someone has a forecast of D_T).

    Then, at any time t, purchase a fraction θ_1 of $\left(\mathbb{E}_t D_T - \sum_{t'=0}^{t-1} x_{t'}\right)$

    any time the price $p_t^f < \theta_2$.
  - » We can write this mathematically as:

$$X_t^\pi(S_t \mid \theta) = \begin{cases} \theta_0 \mathbb{E}_t D_T & t = 0 \\ \theta_1 \left(\mathbb{E}_t D_T - \sum_{t'=0}^{t-1} x_{t'}\right) & \text{If } p_t < \theta_{2,} \text{ for } t = 1, \ldots, T-1 \end{cases}$$

  - » Note: $\mathbb{E}_t D_T$ is the expected value of D_T (which is to say, our forecast of D_T), calculated at time t. This is the same as the conditional expectation

$$\mathbb{E}_t D_T = \mathbb{E}\{D_T \mid S_t\}$$

Purchasing OJ futures contracts

- Now we have to find the best policy in this class:
  - » Simulate the policy $X_t^\pi(S_t(\omega),\theta)$ given $\theta = \left(\theta_0, \theta_1, \theta_2\right)$

$$F^\pi(S_0,\theta,\omega) = \left(X_T^\pi(S_T(\omega),\theta)\, p_T^M(\omega) - \sum_{t=0}^{T-1} p_t^f(\omega) X_t^\pi(S_t(\omega),\theta) \right).$$

  - » Now we have to search for the best value of $\theta = \left(\theta_0, \theta_1, \theta_2\right)$
  - » For a finite-horizon problem like this, it is best to evaluate the policy using an average:

$$\bar{F}^\pi(\theta) = \frac{1}{N}\sum_{n=1}^{N} F^\pi(S_0,\theta,\omega^n)$$

Here, ω^n is the n^{th} simulation, which consists of a sample of prices and the demand at time T.

Purchasing OJ futures contracts

- Other ideas?
  - » Lookahead policy – we might forecast prices and demands, and then solve a deterministic optimization problem using the forecast.
  - » To do this, we first have to define *forecasts* of random variables that we estimate at time *t*.

$$p^f_{t,t'} = \text{Forecast of the contract price for purchases at time } t', \text{ made at time } t.$$

$$p^M_{t,T} = \text{Forecast of the market price at time } T \text{ made at time } t.$$

$$D_{t,T} = \text{Forecast of the demand at time } T \text{ made at time } t,$$

$$\tilde{x}_{t,t'} = \text{Amount that we plan on purchasing at time } t' \text{ when we plan our decisions at time } t.$$

$$\tilde{x}^{sell}_{t,T} = \text{Amount we plan on selling at time } T \text{ when we plan at time } t.$$

Purchasing OJ futures contracts

- We then define our lookahead policy as

$$X_t^\pi(S_t) = \arg\max_{\tilde{x}_t,\ldots,\tilde{x}_{T-1},\tilde{x}_T^{sell}} \left(\tilde{x}_{t,T}^{sell} p_{t,T}^M - \sum_{t'=t}^{T-1} p_{t,t'}^f \tilde{x}_{t,t'} \right)$$

where

$$\tilde{x}_{t,T}^{sell} \le D_{t,T}$$

$$\tilde{x}_{t,T}^{sell} \le \sum_{t'=0}^{t-1} x_{t',T} + \sum_{t'=t}^{T-1} \tilde{x}_{t,t'}$$ Note that $x_{0,T},\ldots,x_{t-1,T}$ are known at time *t*.

$$\tilde{x}_{t,t'} \ge 0 \quad t' = 0,1,\ldots,T-1$$

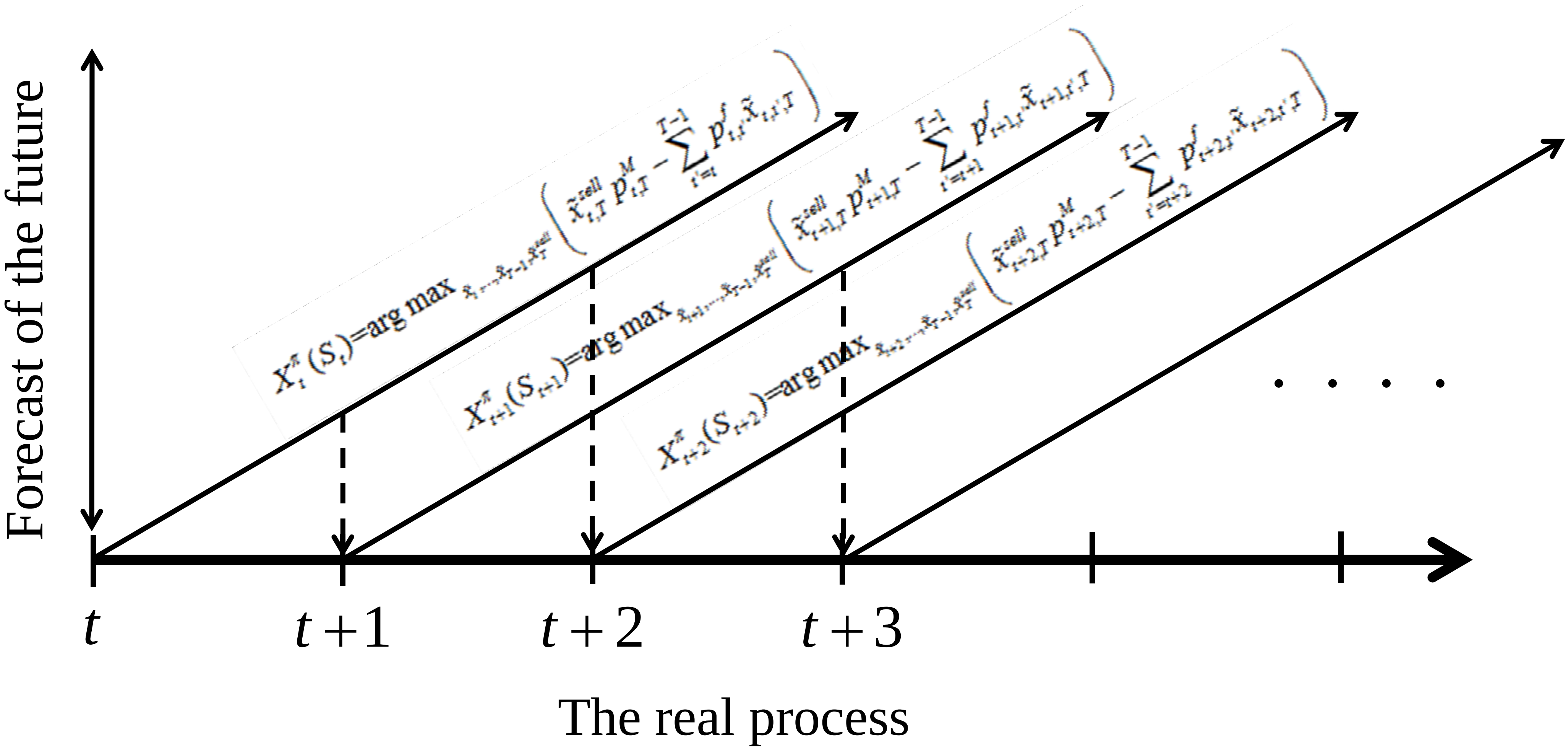
$$\tilde{x}_{t,T}^{sell} \ge 0$$

» Think of the vector $\tilde{x}_{t,t}, \tilde{x}_{t,t+1}, \ldots, \tilde{x}_{t,T-1},$ as a temporary plan of decisions you *might* make in the future, just as you put your finger on the bishop in chess and think about your next move.

» Although we optimize over $\tilde{x}_{t,t}, \tilde{x}_{t,t+1}, \ldots, \tilde{x}_{t,T-1},$ we only keep $x_{t,T} = \tilde{x}_{t,t}$. We then advance the clock and repeat.

Purchasing OJ futures contracts

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



Purchasing OJ futures contracts

- Notes:
  - Very important to keep in mind: t (the first time index) *always* represents when we are making the decision (which determines the information available) while t' (or the second time index), is *when the activity happens*.
  - When making a decision at time t, the first index of every variable in the lookahead policy has to be indexed by t or earlier.
  - If p_T^M is the market price at time T (a random variable at time t), then

$$p_{t,T}^M = \mathbb{E}_t p_T^M$$

is the forecast of p_T^M given what we know at time t. Here, $\mathbb{E}_t$ is referred to as the conditional expectation given what we know at time t.

  - The decision $\tilde{x}_{t,t'}$ can be viewed as a plan of what we are going to purchase at time t', given what we know at time t. This is very similar to the forecast of prices and demands.

Purchasing OJ futures contracts

- Notes:
  - We do not explicitly account for uncertainty within our lookahead model. Instead, we approximate the future deterministically, optimize using solvers for deterministic models, but then only implement the decision for the immediate time period.
  - There is a community known as *stochastic programming* that tries to solve lookahead models which explicitly account for uncertainty.
  - These models are very hard to solve. If you think a lookahead model is best for your problem, but you are not comfortable using a deterministic approximation, call an expert!

Lecture outline

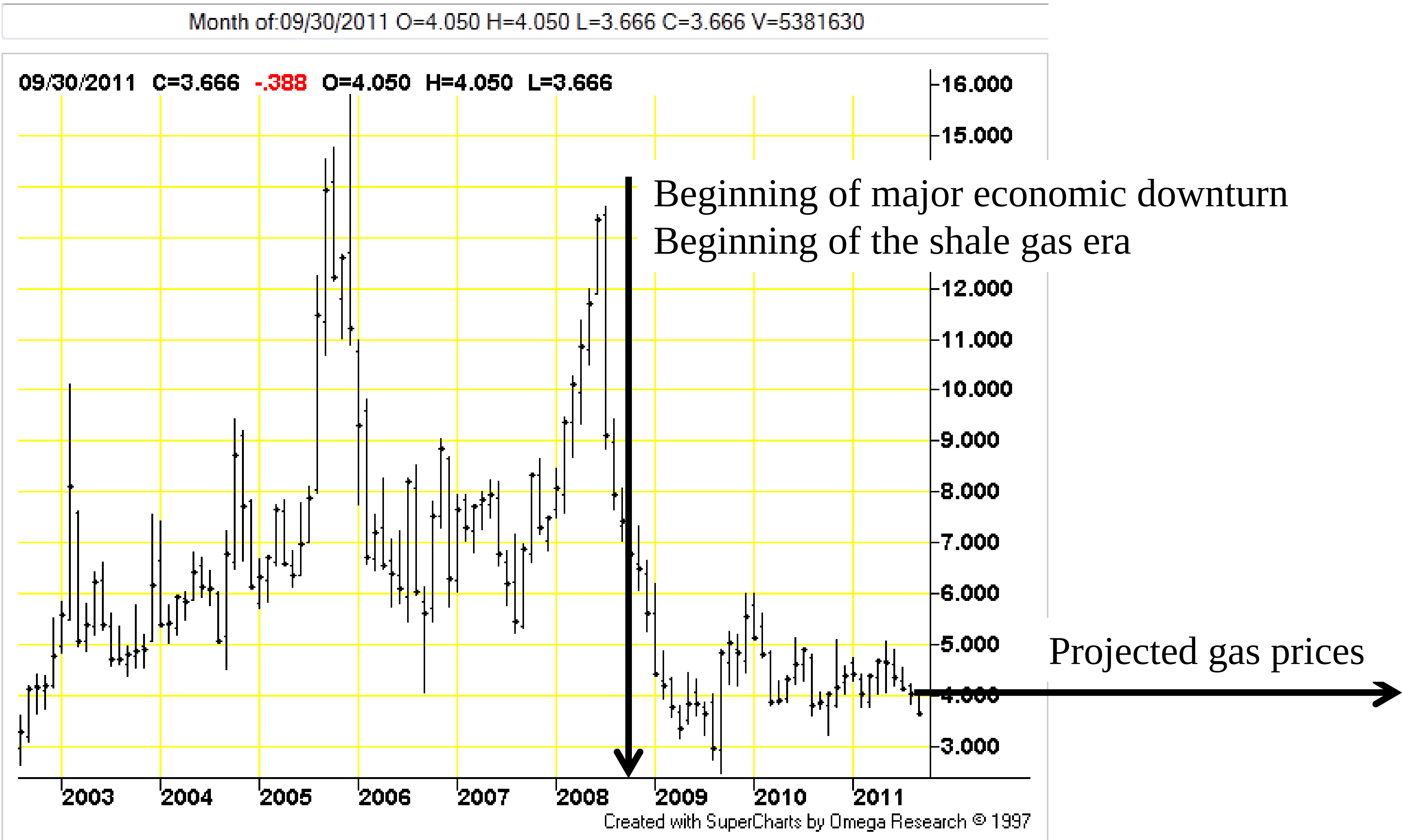
- Managing a water reservoir
- Managing cash balance in a mutual fund
- Purchasing OJ futures contracts
- Modeling natural gas forward contracts
- The four-step process for modeling stochastic, dynamic programs

Natural gas storage

- Basics
  - » Natural gas is a relatively clean, economical source of energy with approximately half the CO_2 footprint of coal.
  - » Widely used in the generation of electricity:
    - In steam generators
    - In combustion turbine generators, which can be turned on and off relatively quickly in response to unexpected changes in demands.
  - » Over the last 10 years, a new manufacturing process has made it possible to economically tap gas locked in shale

Natural gas storage

- Historical natural gas prices

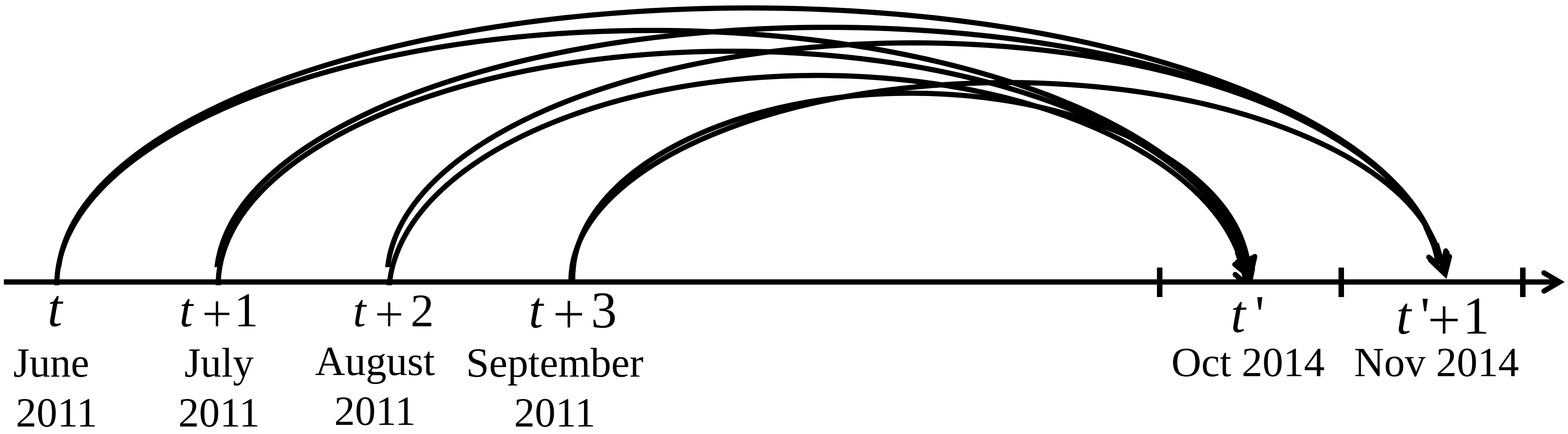


Natural gas storage

- Contrast:
  - » Purchasing forward contracts for electricity, which is a nonstorable commodity (for the most part).
  - » Purchasing forward contracts for a storable commodity such as natural gas. Product not used at t' is held over to time $t'+1$.
- Modeling implications:
  - » How does the issue of storage change how you would model these two problems?
    - What "decision" do you have to make for each problem?
    - How does the modeling of the dynamics change?

Hedging electricity futures

- We can also buy hedges for different points in time in the future.



Natural gas storage

- State variables

$$D_t = \text{Demand for natural gas (in mmBTU) } \textit{during} \text{ time interval } t.$$

$$R_t = \text{Natural gas on hand at the end of time period } t$$

$$= \text{Amount left over that is being stored for } t+1$$

$$p_t^{sales} = \text{Selling price of natural gas during time interval } t.$$

$$p_{t,t'}^{f} = \text{Forward price of gas at time } t \text{ to be delivered during time interval } t' > t$$

$$p_{t,t}^{spot} = \text{Spot price at time } t$$

» What is the complete state variable?

$$S_t = (???)$$

Natural gas storage

- Decision at time t

$$x_t = (x_{t,t'})_{t' \geq t}$$

$$x_{tt} = \text{Spot purchases}$$

$$x_{tt'} = \text{Forward contracts if } t' > t$$

$$y_t = \text{Amount of gas sold during period } t$$

» Constraints:

$$y_t \leq \min\left(D_t, R_{t-1} + \sum_{t''<t} x_{t'',t} + x_{tt} \right)$$

$$R_t \leq R^{\max} = \text{maximum allowable storage.}$$

We do not have to satisfy demand – we may wish to hold natural gas for future orders if prices right now are small.

Natural gas storage

- Exogenous information

$$\hat{D}_t = \text{Change in demand between time intervals } t-1 \text{ and } t.$$

$$\hat{p}^{spot}_{t,t'} = \text{Change in spot price for time } t' \text{ between } t-1 \text{ and } t.$$

$$\hat{p}^{f}_{t,t'} = \text{Change in forward price for time } t' \text{ between } t-1 \text{ and } t.$$

- Transition function

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

$$p^{spot}_{t+1,t+1} = p^{spot}_{t,t} + \hat{p}^{spot}_{t+1,t+1}$$

$$p^{f}_{t+1,t'} = p^{f}_{t,t'} + \hat{p}^{f}_{t+1,t'}$$

$$R_{t+1} = R_t + \sum_{t''\le t+1} x_{t'',t+1} - y_{t+1}$$

Natural gas storage

- Objective function
  - Contribution function:

$$C_t(S_t, x_t, y_t) = p_t^{sales} y_t - p_{tt}^{spot} x_{tt} - \sum_{t'>t} x_{t,t'} p_{t,t'}^f$$

  - Objective: find policies to solve

$$\max_\pi \mathbb{E} \sum_{t=0}^{T} C_t(S_t, X^\pi(S_t), Y^\pi(S_t))$$

Natural gas storage

- A myopic policy:
  - A myopic policy would solve

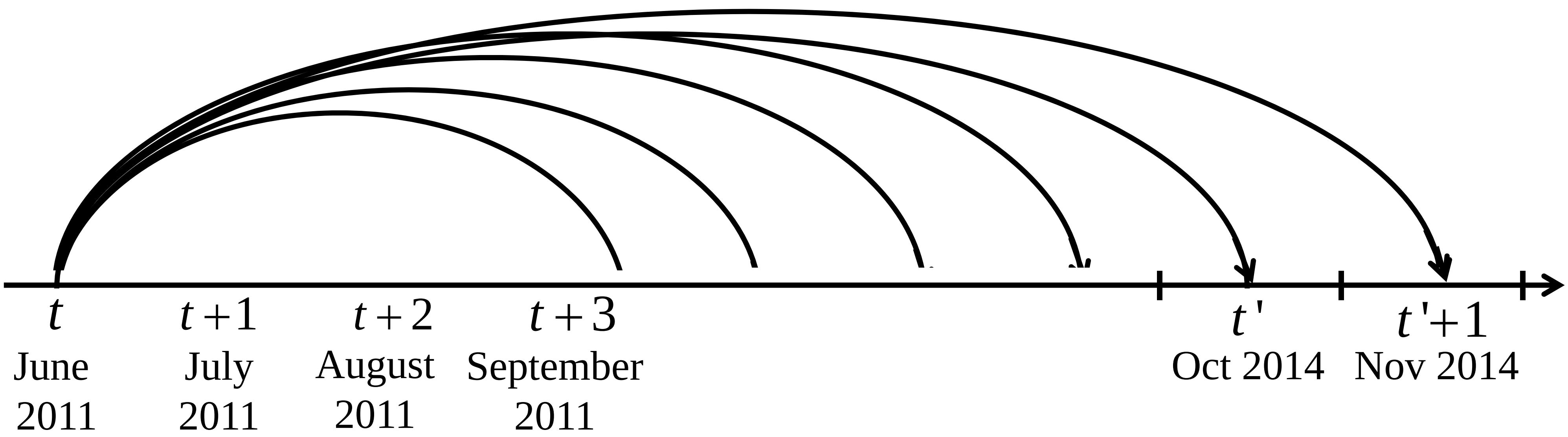
$$\left(X_t^\pi(S_t), Y_t^\pi(S_t)\right) = \arg\max_{\left(x_{t,t'}, y_{t,t'}\right)_{t'=0}^T} C_t(S_t, x_t, y_t)$$

$$= \arg\max_{\left(x_{t,t'}, y_{t,t'}\right)_{t'=0}^T} \left(p_t^{sales} y_t - p_{tt}^{spot} x_{tt} - \sum_{t'>t} x_{t,t'} p_{t,t'}^f \right)$$

  - This has the appearance of optimizing into the future, but it is a myopic policy because it does not consider decisions we can make in the future.

Hedging electricity futures

- A myopic policy only looks at decisions we make at time t. It ignores decisions we might make at times t+1, t+2, …



Natural gas storage

- A lookahead policy
  - » Let:

$$\tilde{x}_{t,t',t''} = \text{Amount we think we will purchase at time } t' \text{ for time } t'' \text{ when we are planning at time } t.$$

$$\tilde{x}_{t,t'} = \left(\tilde{x}_{t,t',t'}, \tilde{x}_{t,t',t'+1}, \ldots, \tilde{x}_{t,t',t'+H}\right)$$

$$\tilde{y}_{t,t'} = \text{Amount we think we will sell at time } t' \text{ when we are planning at time } t.$$

$$\tilde{y}_{t} = \left(\tilde{y}_{t,t}, \tilde{y}_{t,t+1}, \ldots, \tilde{y}_{t,t+H}\right)$$

$$\tilde{S}_{t,t'} = \text{Forecast of state at time } t' \text{ made at time } t$$

  - » We put "tilde's" on variables when they are describing the *lookahead model*. This helps us distinguish the lookahead model from the real model.

Natural gas storage

- A lookahead policy
  - Let:

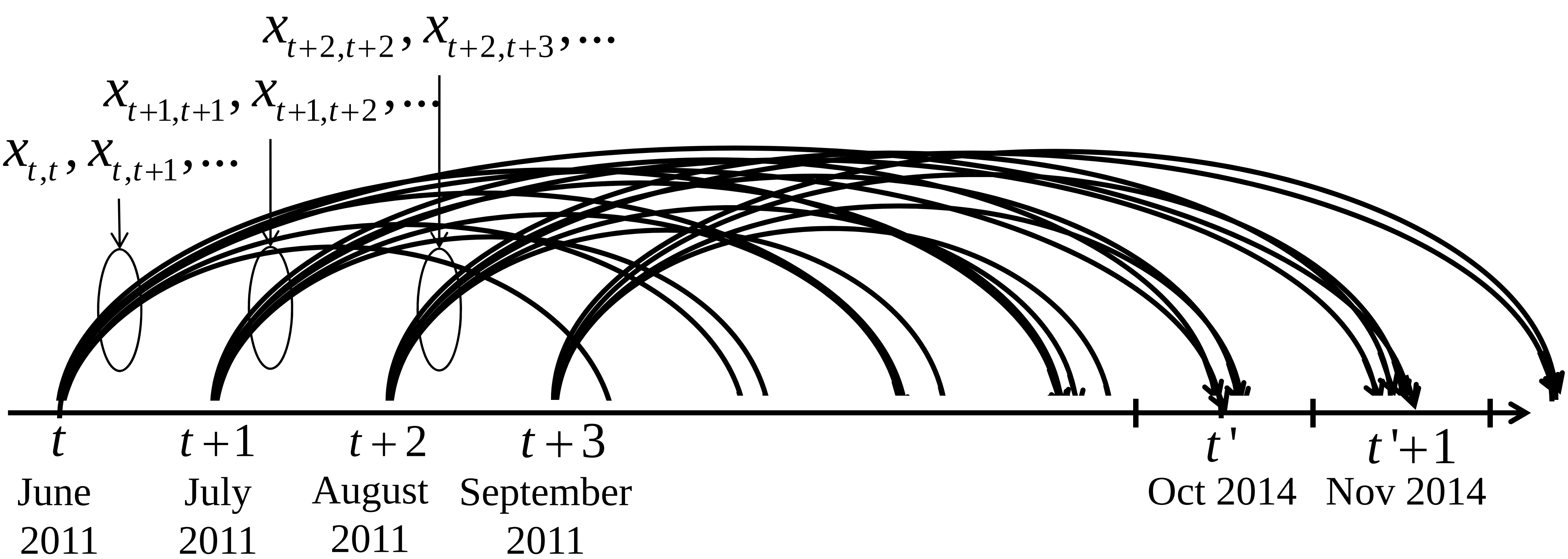
$$\left(X_t^\pi(S_t), Y_t^\pi(S_t) \right) = \arg\max_{\left(\tilde{x}_{t,t'}, \tilde{y}_t\right)} \sum_{t'=t}^{t+H} C_{t'}(\tilde{S}_{t,t'}, \tilde{x}_{t,t'}, \tilde{y}_{t,t'})$$

$$= \arg\max_{\left(\tilde{x}_{t,t'}, \tilde{y}_t\right)} \sum_{t'=t}^{t+H} \left(p_{t,t'}^{sales} \tilde{y}_{t,t'} - p_{t,t't'}^{spot} \tilde{x}_{t,t't'} - \sum_{t''\geq t'} \tilde{x}_{t,t',t''} p_{t,t'}^{f} \right)$$

  - This is a deterministic linear program.
  - All random quantities in the future have been replaced with forecasts.
  - Do you think this will work well?

Hedging electricity futures

- A myopic policy only looks at decisions we make at time t. A lookahead policy considers decisions we *may* make at times t+1, t+2, …, t+H



Lecture outline

- Managing a water reservoir
- Managing cash balance in a mutual fund
- Purchasing OJ futures contracts
- Modeling natural gas forward contracts
- The four-step process for modeling stochastic, dynamic programs

Stochastic optimization models

- The four step process for solving (sequential) stochastic optimization problems
  - » Step 1: Start by modeling the problem deterministically:

$$F(x) = \min_{x_0,\ldots,x_T} \sum_{t=0}^{T} C(S_t, x_t)$$

  - » Step 2: Every time you see a decision x_t replace it with the decision function (policy) $X_t^\pi(S_t)$ and take the expectation.

$$\min_\pi F^\pi = \mathbb{E} \sum_{t=0}^{T} C(S_t, X_t^\pi(S_t))$$

Stochastic optimization models

- The four step modeling process
  - Step 3: Now write out the objective function as a simulation. This can be done as one long simulation:

$$F^{\pi}(\omega) = \sum_{t=0}^{T} C\left(S_t(\omega), X_t^{\pi}\left(S_t(\omega)\right)\right)$$

  - … or an average over multiple sample paths:

$$\bar{F}^{\pi} = \frac{1}{N}\sum_{n=1}^{N}\sum_{t=0}^{T} C\left(S_t(\omega^n), X_t^{\pi}\left(S_t(\omega^n)\right)\right)$$

Stochastic optimization models

- The four step process modeling process
  - » Step 4: Now search for the best policy;
    - First choose a type of policy:
      - – Myopic cost function approximation
      - – Lookahead policy (deterministic, stochastic)
      - – Policy function approximation
      - – Policy based on a value function approximation
      - – Or some sort of hybrid
    - Then identify the tunable parameters of the policy θ
    - Tune the parameters

$$\min_\theta F^\pi(\theta,\omega) = \sum_{t=0}^{T} C\Big(S_t(\omega), X_t^\pi\big(S_t(\omega)\,|\,\theta\big)\Big)$$

… using your favorite stochastic search or optimal learning algorithm.

- Loop over other types of policies.