IE 495 – Lecture 7

Stochastic Programming – Multistage Models and (S)MPS

Prof. Jeff Linderoth

February 3, 2003

Outline

- Multi-stage problems.
  - The “tree” version
- Creating Stochastic Programming Instances
  - MPS format
  - SMPS format

Please don't call on me!

- Name your favorite eight-syllable word?
- What is "block separable recourse"?
  - *Hint:* You probably don't know the answer to this.

Jacob and MIT

- We are given a universe N of investment decisions
- We have a set $\mathcal{T} = \{1, 2, \ldots T\}$ of investment periods
- Let $\omega_{it}, i \in N, t \in \mathcal{T}$ be the return of investment $i \in N$ in period $t \in \mathcal{T}$.
- If we exceed our goal G, we get an interest rate of q that Helen and I can enjoy in our golden years
- If we don't meet the goal of G, Helen and I will have to borrow money at a rate of r so that Jacob can go to MIT.
- We have $\$b$ now.

Variables

- $x_{it}, i \in N, t \in \mathcal{T}$: Amount of money to invest in vehicle i during period t
- y : Excess money at the end of horizon
- w : Shortage in money at the end of the horizon

(Deterministic) Formulation

maximize

$$qy + rw$$

subject to

$$\begin{aligned}
\sum_{i \in N} x_{i1} &= b \\
\sum_{i \in N} \omega_{it} x_{i,t-1} &= \sum_{i \in N} x_{it} \qquad \forall t \in \mathcal{T} \setminus 1 \\
\sum_{i \in N} \omega_{iT} x_{iT} - y + w &= G \\
x_{it} &\geq 0 \qquad \forall i \in N, t \in \mathcal{T} \\
y, w &\geq 0
\end{aligned}$$

One Way to Model

- One way to model this is to create copies of the variables for every scenario at every time period.
- Then we need to enforce *nonanticipativity*...
- Define S_s^t as the set of scenarios that are equivalent (or indistinguishable) to scenario s at time t

A Stochastic Version – Explicit Nonanticipativity

maximize

$$\sum_{s \in S} p_s (q y_s - r w_s)$$

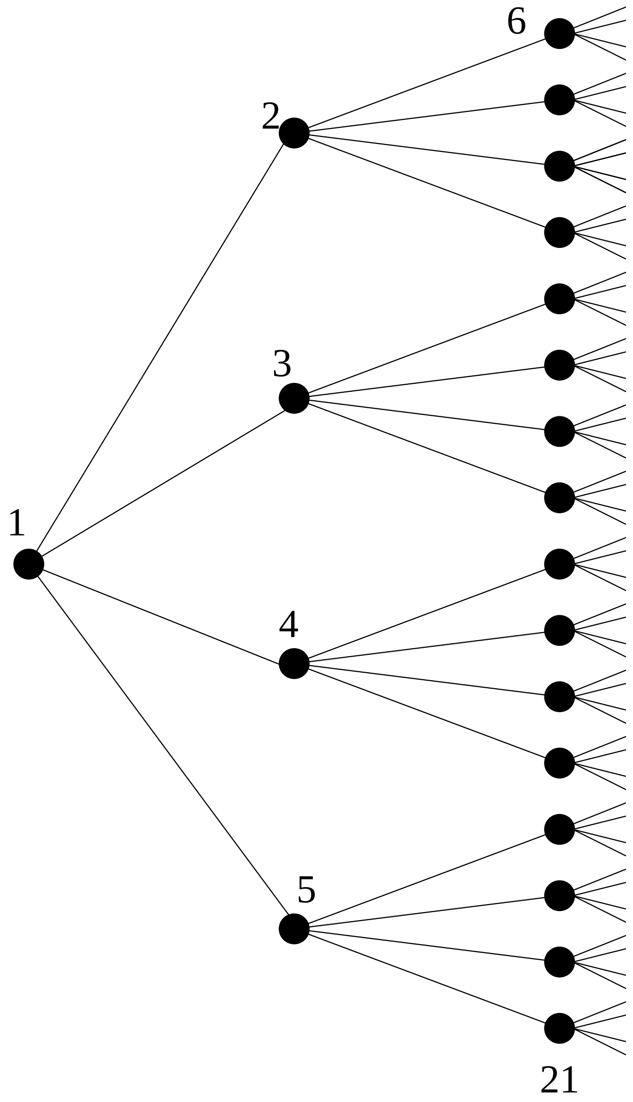
subject to

$$\begin{aligned}
\sum_{i \in N} x_{i1} &= b \\
\sum_{i \in N} \omega_{its} x_{i,t-1,s} &= \sum_{i \in N} x_{its} \qquad \forall t \in \mathcal{T} \setminus 1, \forall s \in S \\
\sum_{i \in N} \omega_{iT} x_{iTs} - y_s + w_s &= G \qquad \forall s \in S \\
x_{its} &= x_{its'} \ \forall i \in N, \forall t \in T, \forall s \in S, \forall s' \in S_s^t \\
x_{its} &\geq 0 \qquad \forall i \in N, t \in \mathcal{T}, \forall s \in S \\
y_s, w_s &\geq 0 \qquad \forall s \in S
\end{aligned}$$

Another Way

- We can also enforce nonanticipativity by just not creating the "wrong" variables
- We have a vector of variables for each node in the tree.
- This vector corresponds to what our decision would be given the realizations of the random variables we have seen so far.
- Index the nodes $l = 1, 2, \ldots \mathcal{L}$.
- We will need to know the "parent" of any node.
- Let $A(l)$ be the ancestor of node $l \in \mathcal{L}$ in the scenario tree.

Jacob-MIT Event Tree



Another Multistage formulation

maximize

$$\sum_{s \in S} p_s (q y_s - r w_s)$$

subject to

$$\begin{aligned}
\sum_{i \in N} x_{i1} &= b \\
\sum_{i \in N} \omega_{il} x_{i,A(l)} &= \sum_{i \in N} x_{il} \qquad \forall l \in \mathcal{L} \setminus 1 \\
\sum_{i \in N} \omega_{iA(s)} x_{iA(s)} - y_s + w_s &= G \qquad \forall s \in S
\end{aligned}$$

AMPL

```
set Investments;
param NumNodes := 21;
param NumScen := 64;
param NumOutcome := 4;
param Return{0 .. NumOutcome-1, Investments};

var x{1 .. NumNodes,Investments} >= 0;
var y{1 .. NumScen} >= 0;
var w{1 .. NumScen} >= 0;

param A{k in 2 .. NumNodes} = ceil((k-1)/NumOutcome);
param A2{s in 1 .. NumScen} = 5 + ceil(s/NumOutcome);
param O{k in 2 .. NumNodes} = (k-2) mod NumOutcome;
param O2{s in 1 .. NumScen} = (s-1) mod NumOutcome;

maximize ExpectedWealth:
 sum{s in 1 .. NumScen} 1/NumScen * (q * y[s] - r * w[s]);

subject to InvestAllMoney:
sum{i in Investments} x[1,i] = b;

subject to WealthBalance1{k in 2 .. NumNodes}:
sum{i in Investments} x[k,i] = sum{i in Investments} Return[O[k],i] * x[A[k],i];

subject to Shortage{s in 1 .. NumScen}:
sum{i in Investments} Return[O2[s],i] * x[A2[s],i] - y[s] + w[s] = G;
```

A History Lesson

- Long before the days of algebraic modeling languages, there were optimization solvers
- Just how did people express their instances and input them to the solver?
- The answer – MPS format

MPS Files

- A "columnwise" formulation of the problem, with the following sections.
  - NAME
  - ROWS
  - COLUMNS
  - RHS
  - RANGES
  - BOUNDS

⋆ *fixed format*!

Example Format

```
*23456789 123456789 123456789
NAME          Example
```

```
*23456789 123456789 123456789
ROWS
 N   OBJECT
 G   ROW2
```

```
*23456789 123456789 123456789 123456789 123456789 123456789
COLS
    COL1      ROW2      2.0                 OBJECT    100.
```

MPS, Cont.

```
*23456789 123456789 123456789
RHS
    RHS1      ROW2      50.
```

```
*23456789 123456789 123456789
BOUNDS
 UP BOUND1    COL1      20.
```

- Also ways to specify integer restrictions and a quadratic objective

MPS Example

- $N = \{1,2\} \equiv \{\text{Stocks, Bonds}\}$.
- $T = 3$

maximize

$$qy + rw$$

subject to

$$\begin{aligned}
x_{S1} + x_{B1} &= b \\
\omega_{St} x_{S,t-1} + \omega_{Bt} x_{S,t-1} &= x_{St} + x_{Bt} \qquad \forall t \in \{2,3\} \\
\omega_{ST} x_{ST} + \omega_{BT} + x_{BT} - y + w &= G \\
x_{it} &\geq 0 \qquad \forall i \in N, t \in \{1,2,3\} \\
y, w &\geq 0
\end{aligned}$$

MPS Example

```
NAME          jake-cor
ROWS
 E  R0001
 E  R0002
 E  R0003
 E  R0004
 N  R0005
COLUMNS
    C0001     R0001     1
    C0001     R0002     0.6092090563
    C0002     R0001     1
    C0002     R0002     0.189873053
    C0003     R0002     -1
    C0003     R0003     0.6092090563
    C0004     R0002     -1
    C0004     R0003     0.189873053
    C0005     R0003     -1
    C0005     R0004     0.6092090563
    C0006     R0003     -1
    C0006     R0004     0.189873053
    C0007     R0004     -1
    C0007     R0005     0.05
    C0008     R0004     1
    C0008     R0005     -0.1
RHS
    B         R0001     10000
    B         R0004     15000
ENDATA
```

AMPL

- How did I create that, you ask?
- AMPL to the rescue...

? Who thinks MPS is ugly?

SMPS – Even Uglier

- How do we specify a stochastic programming instance to the solver?
- We could form the deterministic equivalent ourselves, but you saw how unnatural that seems.
  - ◇ For *really* big problems, forming the deterministic equivalent is out of the questions.
  - ◇ We need to just specify the random parts of the model.
- We can do this using SMPS format
  - ◇ Actually, other than (explicity) forming the deterministic equivalent in a modeling language, this is the ONLY way.
  - ⋆ There *is* some recent research work in developed stochastic programming support in an AML.

SMPS Components

- Core file
  - Like MPS file for "base" instance
- Time file
  - Specifies the time dependence structure
- Stoch file
  - Specifies the randomness

SMPS

- The SMPS format is *broad*[a]
  - There are very few (if any) full implementations, that simply read the full format
  - No solver will solve all instances that can be expressed in the format.
- The SMPS format is (seemingly) being changed.
- A good site...
  - `http://www.mgmt.dal.ca/sba/profs/hgassmann/SMPS2.htm`

[a] *Too* broad, IMO

Papers

- I handed out to you...
  - ◇ J.R. Birge, M.A.H. Dempster, H.I. Gassmann, E.A. Gunn, A.J. King and S.W. Wallace, "A standard input format for multiperiod stochastic linear programs", COAL Newsletter #17 (1987) pp. 1-19.
- There are more. For example...
  - ◇ H.I. Gassmann and E. Schweitzer, "A comprehensive input format for stochastic linear programs", Annals of Operations Research 104 (2001) 89-125.

Did You Hear Me Swearing?

- This morning, I tried to create the SMPS for our little Jacob-MIT example.
- It has been quite a while since I have been so *(@@*()*#$%!@$#!@ angry.
- I don't think it is right, but I'm not sure why.
- Let me show you what I did...
- Many of *YOU* will need to know and use the SMPS format for your projects[a]

[a] This is not necessarily true, but I am looking into useful alternatives

NEOS

- If we have files in SMPS format, how do we solve the resulting instance?
- We can use NEOS!

`http://www.mcs.anl.gov/neos`

Next Time

- Hopefully, I will be able to tell you how to correctly specify the "Jacob-MIT" problem in that !̂&*!̂&$%*!!̂!)((!) SMPS format.
- (Maybe) go over the homework
- Properties of the recourse function
  - ◇ And this time I *mean* it!