

Dynamic, Stochastic Models

- Key difference from *static* stochastic models – conditional decisions
 - Decisions at time t can be conditional on
 - | Realizations of random variables prior to time t
 - | Decisions prior to time t
 - | Knowledge of the conditional distributions for random variables beyond time t

Dynamic, Stochastic Problems

- For example,
 - | Farm management – decisions to apply pesticides may depend on whether pests are present
 - | Finance – decisions to refinance loans may depend the original interest rate on the loan, but may also depend on other refinancing activities and expectations for future interest rate movements

Examples of Dynamic Stochastic Decisions

- | Inventory – restocking plans depend on realized demand
- | Marketing – promotions/pricing may depend on realized yield from production processes
- | Natural resource management – harvesting rates for resources may depend on
 - Stochastic growth in the renewable case
 - Discoveries of reserves in the non-renewable case

Methods for Dynamic, Stochastic Problems

- | Three principle methods: Decision Analysis, Dynamic Programming, and Discrete Stochastic Programming
 - | Decision Analysis (DA) – Best suited to cases where choices and random outcomes are discrete
 - | Dynamic Programming (DP) – Best suited to cases where random outcomes are discrete
 - | Discrete Stochastic Programming (DSP) – Best suited to cases where random outcomes are discrete

Methods (cont'd.)

- Aside: Note that DA and DP can both be applied to problems with discrete choices (and discrete outcomes)
 - | One view is that these are the same method
 - | DA is a graphical, schematic based approach
 - | DP is a recursive equation-driven approach
 - | DP problems are often more structured than DA

Methods (cont'd.)

- From the description, it sounds like DP and DSP sound like the work on the same problem class
- True, but capabilities are different
 - | With DP, the size of problem that we can manage is much greater than with DSP
 - | With DSP, the *complexity* of the problem that we can manage is much greater than with DP

Methods (cont'd.)

- With dynamic programming,
 - ┆ The period-by-period decisions must be simple –
 - Inequality constraints are difficult to incorporate
 - The state variables must have a modest number of components/levels
 - Solutions must be easy to represent
 - ┆ Because of limited need for computer memory (when the state variable description is not extensive), dynamic programming models can accommodate a fairly long time horizon

Methods (cont'd.)

- With discrete stochastic programming
 - ┆ The period-by-period decisions can be relatively complex
 - Inequality constraints can be handled easily
 - The stochastic process describing the evolution of the state variables can be complex relative to DP
 - ┆ While both DP and DSP suffer from the “curse of dimensionality,” the problem is worse with DSP, and limits overall model complexity and time horizon

Discrete Stochastic Programming

- Discrete Stochastic Programming (Cocks, *Man. Sci.*15:72-79, 1969) is implemented as a finite horizon mathematical program
- The approach can handle:
 - | Randomness in right-hand sides for constraints
 - | Randomness in constraint coefficients
 - | Randomness in the objective

Discrete Stochastic Programming

- DSP is designed to accommodate conditional strategies
 - | Because these are implemented as large-scale mathematical programs, we would like to avoid integer variables when possible
 - | As with deterministic, dynamic problems, these problems will typically involve inventory constraints and may involve the scheduling of resources

DSP (cont'd.)

- | Unlike their deterministic cousins, DSP models involve variables that are conditional on preceding outcomes of random variables
 - This has important implications for interpretation, in particular units (e.g., a variable may denote the amount of grain to sell *given that yields are high and prices low*, or a constraint may limit expenditures to *realized* income)
 - Note that this same conditional interpretation of units is needed when addressing penalty costs or shadow prices

DSP (cont'd.)

- | Once a problem for analysis has been defined, several steps are needed to formulate a DSP
 - | The timing of events must be selected
 - | Problem variables and constraints must be selected (some of these will be conditional)
 - | A stochastic process for the random variables must be specified
 - | A problem objective must be selected

DSP (cont'd.)

- Specifying the timing of events
 - | Order of decisions
 - | Order of random events
 - | Interleaving of decisions and random events
- DSP is often called decision making with recourse – this does not mean we can reverse prior decisions, but rather that we can react to our prior decisions and realized prior random events

DSP (cont'd.)

- Problem variables and constraints
 - | The problem variables will reflect the decisions to be made
 - | Constraints will define the restrictions on decisions to be made
 - | Except for initial variables and constraints, all will be conditional on past decisions and realizations of random variables

DSP (cont'd.)

- Selecting a stochastic process for the random events
 - | Typically econometric procedures are used to develop these
 - | Often, some type of autoregressive process is used
 - | Sometimes an autoregressive process is augmented with some additional regression relationships

DSP (cont'd.)

- | For example a study might estimate an autoregressive process to represent average price movements over time for a set of crops (e.g., corn)
- | Additional regression equations might then be used to estimate the price differentials between specialty corn (e.g., high starch or Bt corn) as a function of the corn price
- | Using the AR process, the regressions of price differentials and their error terms, a stochastic process for all corn prices results

DSP (cont'd.)

- Careful attention should be paid to econometric issues in the estimation process
- Selecting the objective function -- two approaches
 - ┆ Expected utility (possibly risk neutral) of ending wealth
 - ┆ Sum of discounted expected utility (direct or indirect) of consumption over the planning horizon



DSP (cont'd.)

- Communicating the formulation of a DSP model is difficult
 - ┆ Equations do part of the job
 - ┆ Diagrams are a powerful supplement to equations
- Perhaps the most informative type of diagram is a tree structure that describes the sequencing of decisions and random events

DSP (cont'd.)

- Trees to describe DSP's – Some conventions must be used to denote the parts of the tree
 - | Time moves forward in the tree from left to right
 - | Nodes in the tree (points where branching occur) denote points where either a decision is made or a random variable is realized
 - | As we move from left to right, there get to be more and more branches

DSP (cont'd.)

- | The right-most part of the tree (the leaves) represents “terminal” states of nature – the payoffs to which the utility function is applied
- | The shape of the nodes in the tree define their type
 - Decision nodes are denoted by squares: 
 - Random nodes are denoted by circles: 

DSP (cont'd.)

- Only one branch emanates from a decision node representing the continuum of decisions that are made
- Several branches (at least two) emanate from a random node reflecting the discrete nature of the distribution of random events

DSP an Illustration

- To illustrate communication of a DSP model using a tree, we will describe a simple gambling game
 - In this game (similar to roulette), there is a wheel which contains 40 numbers 0,1,2,...,38, and 00.
 - Except for 0 and 00, the odd numbers are colored red, and the even numbers are colored black – 0 and 00 are colored green

DSP Illustration (cont'd.)

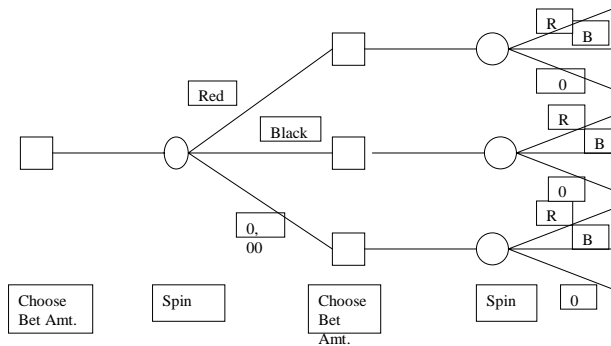
- The player can bet any amount of money on either red or black
 - | If the color they bet on comes up, then they win and receive double the amount of their bet
 - | If the color they bet does not come up, they lose the amount of their bet
 - | If either 0 or 00 comes up, they lose the amount of their bet

DSP Illustration (cont'd.)

- All numbers are equally likely with probability $1/40$
- To simplify this game, we will assume that the player only bets on red – so, the decision is only how much to bet on each turn of the wheel
- Two turns of the wheel will reflect our finite horizon for this problem

DSP Illustration (cont'd.)

■ DSP Tree for Two Spins of Simplified Roulette



Purdue University Ag. Econ. 652
Lecture 17

25

DSP Illustration (cont'd.)

- In constructing our tree, we have defined the time sequencing of decisions and random events
 - | We decide on a bet amount for the first spin
 - | The outcome of the first spin is realized
 - | We decide on a bet amount for the second spin
 - | The outcome of the second spin is realized

Purdue University Ag. Econ. 652
Lecture 17

26

DSP Illustration (cont'd.)

- I Our next step is to define our decision variables
 - I Let x denote the amount we choose to bet on the first spin
 - I Let $y(i)$ denote the amount we choose to bet on the second spin
 - Notice that y is indexed by the outcome of the first random variable (i =red, black, 0)
 - $y(i)$ is a *conditional* decision – e.g., we may bet more on the second spin if we won the first

DSP Illustration (cont'd.)

- I Now consider the constraints on these decisions
 - I In choosing x , we cannot bet more than our current capital, U – so, $0 \leq x \leq U$
 - I In choosing $y(i)$, we cannot bet more than our current capital, but how much is that?

$$V(i) = \begin{cases} U + x & \text{if } i = \text{red} \\ U & \text{otherwise} \end{cases}$$

- I and $0 \leq y(i) \leq V(i)$ (Notice, a conditional constraint)

DSP Illustration (cont'd.)

■ Note that holdings at the end of the two spins is dependent on the outcome of the first spin (i) and the outcome of the second spin (j)

■ We could write these outcomes as $W(i,j)$, and

$$W(i,j) = \begin{cases} U + x + y(i) & \text{if } i \text{ and } j = \text{red} \\ U + x - y(i) & \text{if } i = \text{red and } j = \text{black or } 0 \\ U - x + y(i) & \text{if } i = \text{black or } 0 \text{ and } j = \text{red} \\ U - x - y(i) & \text{if } i \text{ and } j = \text{black or } 0 \end{cases}$$

DSP Illustration (cont'd.)

■ Our stochastic process is terribly simple in this case

┆ Assuming that the spins are stochastically independent, the probabilities of the red, black and 0 branches are the same at each random node in the tree with $\Pr(\text{red}) = 19/40$, $\Pr(\text{black}) = 19/40$, and $\Pr(0) = 2/40$

┆ In terms of notation, denote by $p(i)$ the probability of outcome i (=red, black, or 0) for spin 1 and denote by $q(i,j)$ (=p(j)) the conditional probability of outcome j for spin 2 after outcome i for spin 1

DSP Illustration (cont'd.)

- For our objective, let us simply assume that the decision maker's goal is total expected funds that they take away from the table
- Given this information, we are now ready to write down our DSP problem

DSP Illustration (cont'd.)

$$\text{aximize } \sum_{i=R,B,0} \sum_{j=R,B,0} p(i)q(i,j)W(i,j)$$

subject to: $0 \leq x \leq U$, $0 \leq y(R) \leq U + x$,
 $0 \leq y(B) \leq U - x$, $0 \leq y(0) \leq U - x$,
 $W(R,R) = U + x + y(R)$, $W(R,B \text{ or } 0) = U + x - y(R)$
 $W(B \text{ or } 0,R) = U - x + y(B \text{ or } 0)$,

DSP Illustration (cont'd.)

- Note that the conditional probability $q(i,j)$ multiplies the unconditional probability $p(i)$ to give us the probability of the joint event of i and j occurring
- That is, we are using Bayes' Rule to get the probability of the outcomes of both spins
 $\Pr(i \text{ and } j) = \Pr(i)\Pr(j|i)$
- What is the optimal solution for this problem?