MATH 56A: STOCHASTIC PROCESSES
CHAPTER 4

4. Optimal stopping time

On the first day I explained the basic problem using the example in the book. On the second day I explained how the solution to the problem is given by a "minimal superharmonic" and how you could find one using an iteration algorithm. Also, a simple geometric construction gives the solution for fair random walks. On the third day I explained the variations of the game in which there is a fixed cost per move or if the payoff is discounted. I also explained the transition to continuous time.

4.1. **The basic problem.**

The problem is to find a "stopping time" which optimizes the expected value of a payoff function. I think I gave the same example as in the book: You roll a die. If you get a 6 you lose and get nothing. But if you get any other number you get the value on the die (1,2,3,4 or 5 dollars). If the value is too low you can roll over. The question is: When should you stop? The answer needs to be a strategy: "Stop when you get 4 or 5." or maybe "Stop when you get 3,4 or 5." You want the best "stopping time."

4.1.1. *stopping time.*

Definition 4.1. *In a stochastic process a* stopping time *is a time T which has the property that you can tell when it arrives. I.e., whether or not T is the stopping time is determined by the information that you have at time T.*

Basically, a stopping time is a formula which, given $X_1, X_2, \cdots, X_n$ tells you whether to stop at step n. (Or in continuous time, given X_t for $t \leq T$, tells you whether T is the stopping time.)

Some examples of stopping time are:

(1) the 5th visit to state x
(2) the smallest time T at which $X_1 + X_2 + \cdots + X_T > 100$.
(3) the first visit to the set $\{3, 4, 5\}$.

Date: October 22, 2006.

If T is the first visit to state x then $T-1$ is not a stopping time. (You cannot say "stop right before you get to x." since the process is stochastic and you can't tell the future.)

4.1.2. *payoff function.* The *payoff function* assigns to each state $x \in S$ a number $f(x) \in \mathbb{R}$ which can be positive or negative. This represents what you gain (or lose) if you stop at state x. To figure out whether to stop you need to look at what you can expect to happen if you don't stop.

(1) If you stop you get $f(x)$.
(2) If, starting at x, you take one step and then stop you get

$$\sum p(x,y)f(y)$$

We assume that there is only one transient communication class and $f(x) = 0$ on all recurrent classes.

4.1.3. *value function.* The *value function* $v(x)$ is the expected payoff using the optimal strategy starting at state x.

$$v(x) = \mathbb{E}(f(X_T) \,|\, X_0 = x)$$

Here T is the optimal stopping time. If you don't know what T is then you need another equation:

$$v(x) = \max_T \mathbb{E}(f(X_T) \,|\, X_0 = x)$$

This says you take all possible stopping times T and take the one which gives the maximal expected payoff.

Theorem 4.2. *The value function $v(x)$ satisfies the equation*

$$v(x) = \max(f(x), \sum_y p(x,y)f(y))$$

The basic problem is to find the optimal stopping time T and calculate the value function $v(x)$.

4.2. **Solutions to basic problem.** On the second day I talked about solutions to the optimal stopping time problem. I started with an outline:

(1) Minimal superharmonic is optimal.
(2) Iteration algorithm converges to minimal solution.
(3) Random walks have concave solutions.
(4) Solution for continuous time.

I explained the solutions for discrete time, then converted these into solutions for continuous time.

4.2.1. *minimal superharmonic.*

Definition 4.3. *A* superharmonic *for the Markov chain X_n is a real valued function $u(x)$ for $x \in S$ so that*

$$u(x) \geq \sum_{y \in S} p(x,y)u(y)$$

In matrix form the definition is

$$u(x) \geq (Pu)(x)$$

where u is a column vector.

Example 4.4. *Roll one die and keep doing it until you get a 6. (6 is an absorbing state.) The payoff function is:*

states x	*payoff $f(x)$*	*probability $\mathbb{P}$*
1	150	1/6
2	150	1/6
3	150	1/6
4	300	1/6
5	300	1/6
6	0	1/6

The transition matrix in this example is actually 6×6. But I simplified this to 3 states: $A = 1, 2$ or 3, $B = 4$ or 5 and $C = 6$:

states x	payoff $f(x)$	probability $\mathbb{P}$
A	150	1/2
B	300	1/3
C	0	1/6

Then P is a 3×3 matrix:

$$P = \begin{pmatrix} 1/2 & 1/3 & 1/6 \\ 1/2 & 1/3 & 1/6 \\ 0 & 0 & 1 \end{pmatrix}$$

The best payoff function you can hope for is $u =$ the column matrix $(300, 300, 0)$. Then

$$Pu = \begin{pmatrix} 1/2 & 1/3 & 1/6 \\ 1/2 & 1/3 & 1/6 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 300 \\ 300 \\ 0 \end{pmatrix} = \begin{pmatrix} 250 \\ 250 \\ 0 \end{pmatrix}$$

The equation $u(x) \geq (Pu)(x)$ means the x-entry of the matrix x is $\geq$ the x-entry of the matrix Pu. So, $300 \geq 250$ makes $u = (300, 300, 0)$ superharmonic.

Theorem 4.5. *The value function $v(x)$ is the minimal superharmonic so that $v(x) \geq f(x)$ for all states x.*

This gives a theoretical solution which is useful in some cases (such as the simple random walk).

4.2.2. *iteration algorithm.* As I explained it, $u(x)$ is your estimated expected payoff. The algorithm works like this. You start with u_1 which is the most optimistic. This the payoff you get if you cheat on the next roll.

$$u_1(x) = \begin{cases} 0 & \text{if } x \text{ is absorbing} \\ \max f(y) & \text{if } x \text{ is transient} \end{cases}$$

Next, u_2 is your expected payoff if you play fair for one round and then cheat. u_n is your payoff if you wait n turns before cheating. The recursive formula for u_{n+1} given u_n is

$$u_{n+1}(x) = \max(f(x), (Pu_n)(x))$$

At each stage u_n is superharmonic and $u_n(x) \geq f(x)$ but the values get smaller and smaller and become minimal in the limit:

$$v(x) = \lim_{n\to\infty} u_n(x)$$

$v(x)$ is your expected payoff if you put off cheating indefinitely.

In the example,

$$\begin{array}{rcrrr} u_1 & = & (300, & 300, & 0) \\ u_2 & = & (250, & 300, & 0) \\ u_3 & = & (225, & 300, & 0) \\ u_4 & = & (212.5, & 300, & 0) \\ v = u_\infty & = & (x, & 300, & 0) \end{array}$$

where $x = 200$ is the solution of the equation:

$$x = \frac{x}{2} + \frac{300}{3}$$

4.2.3. *convex value function.* Suppose you have a simple random walk with absorbing walls. Then a function $u(x)$ is superharmonic if

$$u(x) \geq \frac{u(x-1) + u(x+1)}{2}$$

In other words, the point $(x, u(x))$ is above the point which is midway between $(x-1, u(x-1))$ and $(x+1, u(x+1))$. So, superharmonic is the same as convex (concave down). So, the theorem that the value function $v(x)$ is the minimal superharmonic so that $v(x) \geq f(x)$ means that the graph of $v(x)$ is the convex hull of the graph of $f(x)$.

4.2.4. *continuous time.* In a continuous Markov chain you have an infinitesmal generator A which is a matrix with transition rates $\alpha(x, y)$ which are all nonnegative except for $\alpha(x, x) = -\alpha(x)$. Since the rows add to zero we have

$$\alpha(x) = \sum_{y \neq x} \alpha(x, y)$$

So, you get a probability matrix P with entries

$$p(x, y) := \frac{\alpha(x, y)}{\alpha(x)}$$

for $x \neq y$ (and $p(x, x) = 0$). This is the probability of first jumping to y from x:

$$p(x, y) = \mathbb{P}(X_J = y \mid X_0 = x)$$

where J is the first jump time:

$$J = J_1 := \inf\{t \mid X_t \neq X_0\}$$

Anyway, you use the discrete probability transition matrix P and transform it into continuous time by looking only at jump times: The optimal stopping time for the continuous process is J_T, the T-th jump time where T is the stopping time for the discrete process.

4.3. **Cost functions.** The *cost function* $g(x)$ gives the price you must pay to continue from state x. If T is your stopping time then you continued T times. So your total cost is

$$g(X_0) + g(X_1) + \cdots + g(X_{T-1}) = \sum_{j=0}^{T-1} g(X_j)$$

So, your net gain is

$$f(X_T) - \sum_{j=0}^{T-1} g(X_j)$$

The value function $v(x)$ is the expected net gain when using the optimal stopping time starting at state x. It satisfies the equation:

$$v(x) = \max(f(x), (Pv)(x) - g(x))$$

4.3.1. *iteration algorithm.* First, you take $u_1(x)$ to be your most optimistic estimate of expected gain. If you go one step in the Markov chain then you have to pay $g(x)$ so your best possible net gain would be

$$\max_{y\in S} f(y) - g(x)$$

If this is less than $f(x)$ you can't possibly do better by continuing. So

$$u_1(x) = \begin{cases} 0 & \text{if } x \text{ is absorbing} \\ f(x) & \text{if } f(x) \geq \max_{y\in S} f(y) - g(x) \\ \max_{y\in S} f(y) - g(x) & \text{otherwise} \end{cases}$$

u_{n+1} is given in terms of u_n by:

$$u_{n+1}(x) = \max(f(x), (Pu_n)(x) - g(x))$$

where $(Pu_n)(x) = \sum p(x,y)u_n(y)$.

4.3.2. *random walk.* For the simple random walk with absorbing walls, the value function is the smallest function $v(x) \geq f(x)$ so that

$$v(x) \geq \frac{v(x-1) + v(x+1)}{2} - g(x)$$

4.4. **Discounted payoff.** Here we assume that the payoff is losing value at a fixed rate so that after T steps it will only be worth $\alpha^T f(x)$ where α is the discount rate, say $\alpha = .90$. Then the value function satisfies the equation

$$v(x) = \max(f(x), \alpha(Pv)(x))$$

Again there is a recursive formula converging to this answer:

$$u_{n+1}(x) = \max(f(x), \alpha(Pu_n)(x))$$

where you start with

$$u_1(x) = \begin{cases} 0 & \text{if } x \text{ is absorbing} \\ f(x) & \text{if } f(x) \geq \alpha f(y) \text{ for all } y \\ \max \alpha f(y) & \text{otherwise} \end{cases}$$