7. Introduction to Numerical Dynamic Programming

AGEC 642 - Spring 2024

I. An introduction to Backwards induction

Shively, Woodward and Stanley (1999) provide some recommendations about how to approach the academic job market. Their recommendations are summarized in the table below. The thing to note here is that they start with year 5 when you begin your job. Starting at the end, they then move backward in time, making suggestions about how a student should put all the pieces into place so that when the end comes his or her application packet is as strong as possible. The process of looking first to the end is called backward induction. It is not only a critical skill for evaluating almost any problem that we face; it is the central concept in dynamic programming.

Timetable of Job-Search Activities

Time	Activity
year 5	• Start new job
	• Obtain job offers and negotiate
	• On-campus interviews
year 4	• Interview at professional meetings
	• Begin applying for positions
	• Complete outline of dissertation and one chapter
	• Teach a class
year 3	• Revise and resubmit paper
	• Look for teaching assignment
	• Attend AAEA meetings
	• Write and submit a book review
	• Revise paper and submit to journal & AAEA meetings
	• Obtain first teaching experience
year 2	• Give an informal seminar based on the revised paper
	• Submit grant proposals
	• Research grant opportunities
	• Revise paper
	• Take qualifying exams
	• Write course paper with an eye toward eventual publication
year 1	• Choose field, advisor, and dissertation topic
	• Start classes

[*] Taken from Shively, G., R.T. Woodward, and D. Stanley. 1999. "Strategies and Tips for Graduate Students Entering the Academic Job Market." *Review of Agricultural Economics*.21 (2):513-526.

A. *The two-period consumption problem*

Now let's consider a more standard economic problem, the two-period consumption problem that we looked at in Lecture 1. In that problem, we said that the individual had a utility function over two goods, z_a and z_b, which she consumes in two periods, 1 and 2:

$$u(\mathbf{z})=u_1(z_{1a},z_{1b})+u_2(z_{2a},z_{2b})\ .$$

The resource endowment x could be consumed over the two periods, so that

$$\mathbf{p}'z_1+\mathbf{p}'z_2\le x.$$

Solving for the second period first, we found that

$$V_2\left(x_2\right)=\max_{z_2} u_2\left(z_{2a},z_{2b}\right) \quad s.t.$$

$$p_a z_{2a}+p_b z_{2b}\le x_2$$

We could then solve the first-period problem,

$$\max_{z_1} u\left(z_{1a},z_{1b}\right)+V_2\left(x_2\right) \quad s.t.$$

$$x_2=x_1-\left(p_a z_{1a}+p_b z_{1b}\right).$$

This is the dynamic programming approach: in each period we solve for the value function, which is a function of the state variable(s). Then, working backward, we find the optimal choice in the previous period and the corresponding value function in that period. This process is repeated until the initial period is reached. In this course, we will solve these problems numerically and we start with a very simple example.

II. Numerical solutions of a simple DDP

A. Consider the following problem.

- You have a contract to sell N=300 units of your product at the start of period T=4. If you deliver your product, you will be paid p=\$10 per 100 units. However, if you are unable to fulfill the contract you have to pay a penalty of d=\$2 per 100 units you are short. Hence, the salvage value equation as a function of the x_T units held in period T is:

$$S(x_T)=\begin{cases} p\dfrac{N}{100} & \text{if } x_T\ge N \\ p\dfrac{x_T}{100}-d\dfrac{\left(N-x_T\right)}{100} & \text{if } x_T<N \end{cases}$$

- Building inventory (using your choice variable, z_t) is expensive and gets more expensive the more you produce in a single period, i.e., $c\left(z_t\right)=\alpha\left(\dfrac{z_t}{100}\right)^2$, with α=1
- Holding inventory is also expensive, $c_B\left(x_t\right)=\gamma\dfrac{x_t}{100}$, with γ=3.
- Production increases next period's inventory; you cannot increase your inventory and sell that increase in the same period. ⇒ $x_{t+1}=x_t+z_t$.
- Hence the benefit function for periods 1-3 is

$$B\left(z_t,x_t\right)=-\alpha\left(\frac{z_t}{100}\right)^2-\gamma\frac{x_t}{100}.$$

- A formal statement of the optimization problem is, therefore,

$$\max_{z_t}\sum_{t=1}^{3}B\left(z_t,x_t\right)+S(x_4) \qquad s.t.\ x_{t+1}=x_t+z_t, x_0=0.$$

B. How do we solve this problem using a DP setup?

We will fill out the circle & arrow diagram on the last page of these notes to develop the intuition. Remember, DP is about working backward, so we start in stage 4 and then work backward.

A video in which this example is worked is available on the class website. I encourage you to watch this video while completing your own circle-and-arrow sheet.

First, we need to move to the final period when the transaction is made. We find the value of the ending inventory as a function of the possible terminal values of x_T. For simplicity, we will assume that inventory is created in 100-unit increments. The value of being in period T with a stock of x_T can be written $V_T(x_T)=S(x_T)$ for all possible values of x=0, 100, 200, 300, 400, …. Since $V(x_T)$ does not increase beyond x_T=300, so there is no point in evaluating values greater than 300.

The next step is when we start doing DP. We move backward from period $t=T=4$ to period $t=T-1=3$ and find the best choice from the available options (z_3=0,100,200, or 300) at each value of x_3. The value of being at a particular point (say x_3=200) is equal to the highest value that can be achieved if we happen to end up at that point, i.e.

$$V_3\left(x_3\right)=\max_{z_3=0,1,2,3} B\left(z_3,x_3\right)+\beta V_4\left(x_4\left(x_{3,}z_3\right)\right)$$

where $x_4\left(x_{3,}z_3\right)$ is the state equation, a mapping from x_3 to x_4 as a function of z_3. In this case, the state equation is simple: $x_4 = x_3+z_3$. This type of equation is called a Bellman's equation, after the applied mathematician, Richard Bellman.[1]

Let's write out the Bellman's equation in even more detail for two points in the state space, where x_3=0 and where x_3=200.

If x_3=0, then the Bellman's equation can be written:

$$V_3\left(x_3=0\right)=\max_{z_3}\begin{cases}B(0,0) \quad +\beta V_4(0) \quad \text{if } z_3=0\\ B(0,100)+\beta V_4(100)\text{if } z_3=100\\ B(0,200)+\beta V_4(200)\text{ if } z_3=200\\ B(0,300)+\beta V_4(300)\text{ if } z_3=300\end{cases}$$

If x_3=200, then the Bellman's equation can be written:

$$V_3\left(x_3=200\right)=\max_{z_3}\begin{cases}B(200,0) \quad +\beta V_4(200)\text{ if } z_3=0\\ B(200,100) +\beta V_4(300)\text{ if } z_3=100\\ B(200,200) +\beta V_4(300)\text{ if } z_3=200\\ B(200,300) +\beta V_4(300)\text{ if } z_3=300\end{cases}$$

[1] An interesting biographical sketch on Richard Bellman is provided in the article, Dreyfus, S. 2002. Richard Bellman on the Birth of Dynamic Programming. *Operations Research* 50(1):48-51.

Notice that here we are using the fact that we know that $V_4(x)$ reaches a maximum when x_4=300. While this is not what we usually think of as a *function*, these two expressions, each with four lines on the right, are functions evaluated at a point, mappings from the variable x_3 to a real number. In this case, the function $V(\cdot)$ is only defined at discrete values of x_3, {0, 100, 200, …}.

The optimal choice, $z_3^*(x_3)$, is the one that solves the maximization problem above. Or, more technically,

$$z_3^*(x_3) = \arg\max_{z_3}\left\{B(z_3,x_3)+\beta V_4\left(x_4\left(x_{3,}z_3\right)\right)\right\}.$$

Once the Bellman's equation is solved for all possible values of x_3 in period 3, we can move backward to period 2, solving

$$\begin{aligned} V_2(x_2) &= \max_{z_2=0,1,2,3} B(z_2,x_2)+\beta V_3\left(x_3\left(x_{2,}z_2\right)\right) \\ &= \max_{z_2=0,1,2,3} B(z_2,x_2)+\beta V_3(x_2+z_2). \end{aligned}$$

We continue moving backward in this fashion until the $V(\cdot)$ and $z^*(\cdot)$ are identified for all points in the grid. This tells us the value function and the optimal path.

***Bellman's principle of optimality*:** If you end up at a particular value of x, then the best thing you can do from that point forward is the same thing you would do if you were starting at that value of x.

Bellman's principle of optimality is central to the dynamic programming and is embedded in the *Bellman's Equation*,

$$V(x_t,t) = \max_{z_t} u(z_t,x_t,t)+\beta V(x_{t+1},t+1)$$

in which $V(\cdot)$ is known as the *value function*, and β is the *discount factor*, typically equal to $\dfrac{1}{1+r}$ where r is the discount rate. Writing the Bellman's equation more specifically for the inventory control problem, in which we are not discounting, we obtain

$$V(x_t,t) = \begin{cases} \max\limits_{z_t} -\left(\alpha\cdot\left(\dfrac{z_t}{100}\right)^2+\gamma\dfrac{x_t}{100}\right)+V(x_{t+1},t+1); & \text{for } t<T \\ S(x_T);\ \text{otherwise.} & \end{cases}$$

Note here that the value function is a function of both x_t and t; it can take on a very different form in each period.

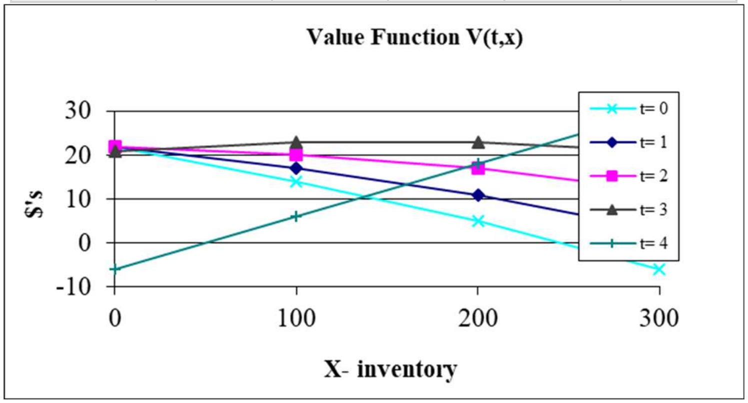
III. Presentation of your results

The key outputs of a DP problem are the value function and the optimal choices as a function of the state and stage. It is often useful to present these in either graphical or tabular form.

For example, the value function and policy functions are presented below. From the value function, we see that stock (x_t) is not always valuable; the marginal value of the stock is strictly less than zero in periods 0, 1, and 2, only becoming valuable in period 3. This is reflected in the policy function, in which we see that it is optimal to add inventory only in period 3

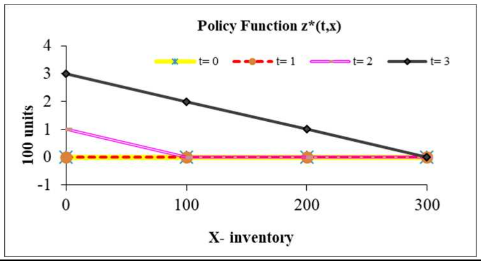
$V(x_t, t)$

x	t= 0	t= 1	t= 2	t= 3	t= 4
0	22	22	22	21	-6
100	14	17	20	23	6
200	5	11	17	23	18
300	-6	3	12	21	30



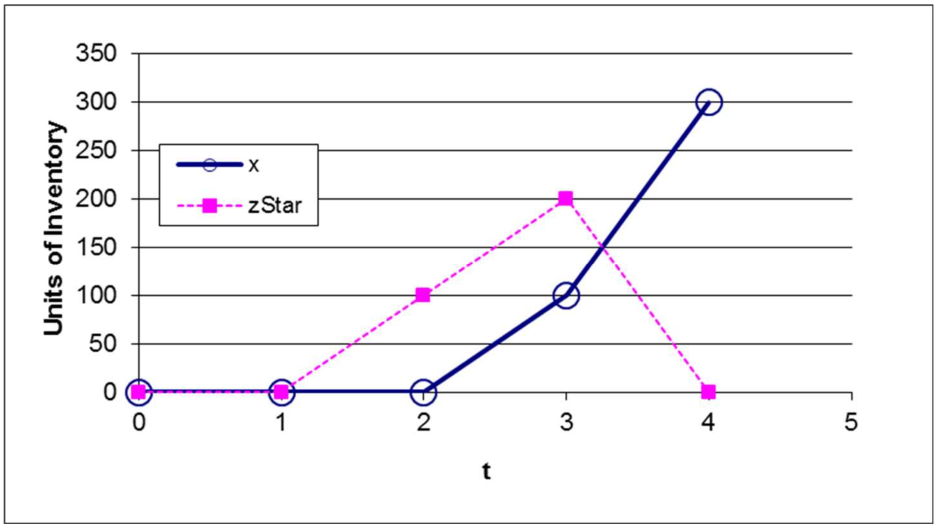
$z^*(x_t,t)$

x_t	t=0	t=1	t=2	t=3
0	0	0	100	300
100	0	0	0	200
200	0	0	0	100
300	0	0	0	0

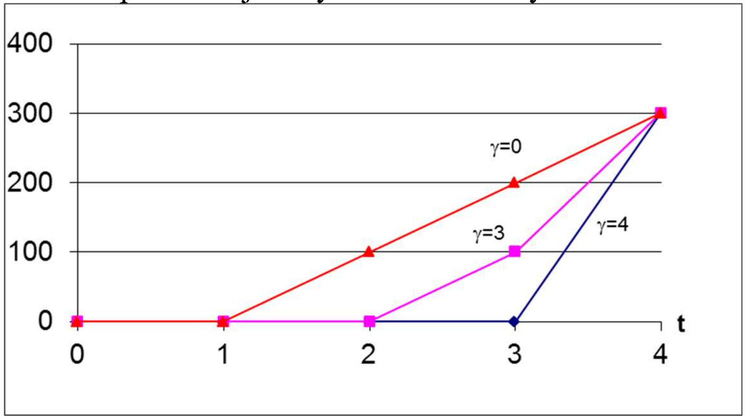


Another helpful way to look at your results is by simulating an optimal path. Again, graphical or tabular presentation is often useful as in the figure below.

Simulated optimal path



Much of the economic content of your solution might be obtained from comparative dynamic analysis, i.e., how do your results change as your parameters change? For example, as we vary γ, the cost of holding the inventory changes. In the figure below we see how this affects the optimal trajectory of the inventory stock.



IV. Extensions of the simple DDP model

One of the most attractive features of dynamic programming is that extending the basic structures in a variety of ways is relatively straightforward. Here are a couple of obvious extensions of the inventory control problem.

A. Multiple State variables (e.g., 2 different goods)

Inside your stage loop, nest a loop over each of your state variables.
Suppose, for example, the firm produces 2 goods, x^1 and x^2. In this case, the Bellman's equation could be written:

$$V\left(x_t^1, x_t^2, t\right) = \max_{z_t^1, z_t^2} u\left(z_t^1, z_t^2, x_t^1, x_t^2, t\right) + \beta V\left(x_{t+1}^1, x_{t+1}^2, t+1\right).$$

If costs are interrelated in a nonlinear fashion, then it is important to solve the joint optimization problem. The solution algorithm would require looking at the optimal choices at each state-stage combination, e.g.,

t	*4*	*4*	*4*	*4*	*4*	*4*	*4*	*4*	*4*	*4*	*4*	*4*	*4*	*4*	*4*	*4*		*3*	*3*	…
x_1^t	0	1	2	3	0	1	2	3	0	1	2	3	0	1	2	3		0	1	…
x_2^t	0	0	0	0	1	1	1	1	2	2	2	2	3	3	3	3		0	0	…

At each of these stage-state combinations, solving the Bellman's equation requires finding the z_t^1, z_t^2 choice that maximizes $u(\cdot) + \beta V(\cdot)$.

To reiterate the point made above, notice that the solution algorithm starts with t=4 and moves backward.

B. Risk

One of the advantages of DP over optimal control is the ease with which risk can be incorporated into any problem. For example, we could consider a problem in which 100 units might be stolen or damaged from one period to the next and the probability that this happens is π. In this case, your Bellman's equation would take the form

$V_{\mathrm{t}}(x_t)= -(\alpha z^2 + \gamma x_t) + [\pi\cdot V_{t+1}(x_t-100+z_t) + (1\text{-}\pi)\cdot V_{t+1}(x_t+z_t)]$

In general, one can always add uncertainty to a DP problem by simply adding the expectation operator

$$V\left(x_t\right) = \max_{z_t} E\left[u\left(z_t, x_t\right) + V\left(x_{t+1}\right)\right].$$

If the utility function is deterministic, then this could be written

$$V\left(x_t\right) = \max_{z_t} u\left(z_t, x_t\right) + EV\left(x_{t+1}\right)$$

or

$$V\left(x_t\right) = \max_{z_t} u\left(z_t, x_t\right) + \sum_j \pi_j V\left(x_{t+1}\left(z_t, x_t, \varepsilon_j\right)\right)$$

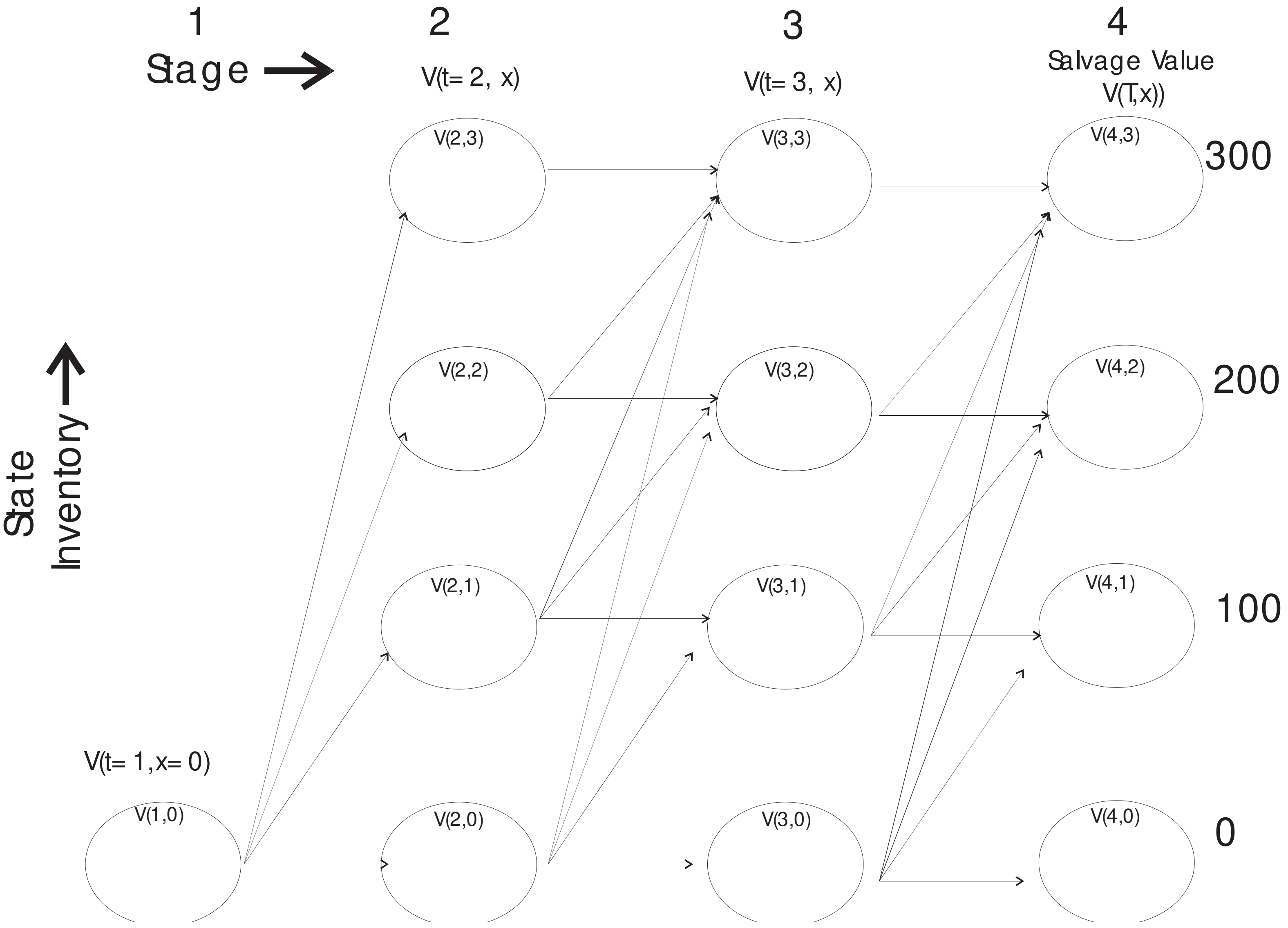
where π_j is the probability that ε_j occurs and $x_{t+1}(\cdot)$ is the state equation contingent on the random variable ε. Remember that the expectation operator must go <u>outside</u> the value function – do not use $V\left(Ex_{t+1}\right)$ instead of $EV\left(x_{t+1}\right)$.

V. References

Shively, Gerald, Richard Woodward, and Denise Stanley. 1999. Strategy and Etiquette for Graduate Students Entering the Academic Job Market. *Review of Agricultural Economics* 21(2):513-26.

VI. Readings for next class

Judd (1998) pp. 399-413.


1
2
3
4
Stage
V(t= 2, x)
V(t= 3, x)
Salvage Value
V(T,x))
V(2,3)
V(3,3)
V(4,3)
300
State
Inventory
V(2,2)
V(3,2)
V(4,2)
200
V(2,1)
V(3,1)
V(4,1)
100
V(t= 1,x= 0)
V(1,0)
V(2,0)
V(3,0)
V(4,0)
0