

Practical Dynamic Programming for Business and Forensic Economics

by Patrick L. Anderson

Principal, Anderson Economic Group LLC
East Lansing Michigan USA

*presented at the
National Association of Forensic Economists
International Conference
Dublin, Ireland
May 2005*

AEG Working Paper 2005-5

Minor revision, 2 June 2005

(c) 2005 AEG LLC. Rights for research and nonderivative use granted to NAFE members. All other rights reserved.

*Practical Dynamic Programming
for Business and Forensic Economics*

by Patrick L. Anderson

Abstract

One of the most important applications of forensic economics is estimating lost profits in commercial disputes and lost earnings of workers in cases of injury. The most common method for estimating such damages has been to reconstruct them using an “income” or “discounted cash flow” approach, based on assumptions about how similar workers or firms operate. Such methods can work well if the static assumptions accurately capture the dynamic optimization that actual workers and managers perform.

Another approach is to use the optimization technique of “dynamic programming,” introduced by mathematician Richard Bellman in 1957. Such a method can be utilized to estimate losses of income for workers or businesses, and do so while explicitly modeling the optimization behavior of the persons involved. This method has emerged as the tool of choice for many difficult optimization questions in finance and economics in the academic world, and it was recently introduced as a method applicable to business valuation.

In this paper, we (1) motivate the use of this method, using the concept of intertemporal optimization by a consumer; (2) describe the fundamental technique of dynamic programming, which converts an potentially intractable problem involving multiple variables and many time periods into a 2-period problem that may be tractable; (3) outline simple models for lost earnings of workers and lost profits for business; and (4) briefly note the history of the technique and cite sources for the mathematical and computational issues not addressed in this paper. An appendix includes an example business valuation problem that has been solved using this technique and recently-developed computational software.

I. Motivation and Outline

A method of solving complicated, multi-stage optimization problems called dynamic programming was originated by American mathematician Richard Bellman in 1957. Bellman's 1957 book motivated its use in an interesting essay that is still vital reading today, and is astonishingly devoid of obtuse mathematical notation.

Like most brilliant insights, the method promised a radical simplification of a set of difficult problems. And, like most brilliant insights, it was difficult to readily put into practice. However, the fundamental logic of the approach attracted some attention over the next four decades, resulting in it being further developed. In the last decade, advances in computational methods have addressed some of the still-intransigent aspects of the approach. In the last few years, it has been introduced as a method for business valuation, albeit with reservations about the current practicality.

In this paper, we first introduce two common problems that are common in applied economics: maximizing wages for the worker, and maximizing returns as an investor. We explain how these are We then organize these are intertemporal optimization problems, and then outline the recursive approach to solving them, using a simplified dynamic programming method. We note briefly how this approach differs from the static cash-flow models that are the workhorses of most financial analyses. We state some advantages of this approach, and also highlight the disadvantages, notably the significantly greater mathematical difficulty and its novelty.

(3) outline three uses in lost earnings for workers, and lost profits for business; and (4) briefly note the history of the technique and cite sources for the mathematical and computational issues not addressed in this paper. An appendix includes three additional examples that have been solved using recently-developed computational software.

II. Wage and Business Earnings as an Optimization Problem

We typically assume the manager of a business attempts to maximize profits, and that a worker attempts to maximize his or her wages, subject to constraints on time and resources. The worker and the manager have some ability to save or invest during one period, and the effects of this decision affect the state of the world in the next period. This framework is used in policy discussions, courtrooms, investment committees, management meetings, and family kitchens across the

globe. In mathematical terms, it is an intertemporal optimization problem under constraints.

Optimization Across Time: The Worker

Consider how the colloquial employee request “I want a raise” would be framed in mathematical terms:

1. I earn a wage or salary at a constant rate per time period, for working a certain number of hours in a certain occupation.
2. I am subject to a constraint on the amount of time that I can (or am willing to) work, and that constraint is currently binding.
3. I expect the salary or wage to continue in a path of regular secular increases, until my retirement.
4. I want a one-time adjustment upward in my wage per time period, which will affect the path of future wages by proportionately shifting it upward.

A worker’s efforts to work harder, in order to earn an increase in salary, is an example of intertemporal optimization. The employee is working harder today (and giving up satisfaction-producing activities to do it), in the hopes of increasing earnings tomorrow and from there into the future. Indeed, the worker’s sacrifices to gain an increase in salary would make little sense unless one considered the results of those sacrifices over time. Expanding the list of factors under the worker’s control to the ability to gain skills or find a different job, and expanding the variables outside his or her control to the expected work-life and the risks of incapacitating injury, adds richness to the problem. However, in structure it is still an intertemporal optimization problem with constraints.

Optimization Across Time: The Investor

We commonly assume that an investor wants to maximize “total return,” meaning the sum of current profits and capital gains.¹ Some investors, of course, are more interested in one form or earnings than another; we further know that investors have a broad range of risk preferences, not to mention outright prejudices about specific assets. The notion of “total return” implies intertemporal optimization.² Of

¹ This definition of investment performance is embedded in many metrics used in the investment community. For example, many analyses of past stock market returns assume reinvestment of dividends.

² An investment steadily returning a dividend with no price appreciation would often provide more cash return to an investor during a short time period than one with no dividend but which, when sold, yielded a significant capital gain. Investors that choose a mix of assets are therefore choosing the time periods during which

course, investors do not know the future, and must accept uncertainty about future returns, so their optimization must take place in an environment of uncertainty.

III. Discounting, Uncertainty, and Intertemporal Optimization

The idea of optimizing across time is as basic as the human practice of saving food for the next day. The fundamental microeconomics are well developed. To introduce the complicated mathematics of dynamic programming, we consider the simple example of a consumer maximizing the utility from consumption.

Simple Optimization Across Time Periods: The Consumer Saving Problem

Consider a consumer that has title to a stream of income over future periods, and wishes to maximize his or her satisfaction from purchasing goods and services with that income. Alas, this being the world and not paradise, the stream of income is both random, and insufficient to purchase all the worldly goods the consumer desires.

The consumer therefore develops a utility function, which we assume is quite typical of human behavior: the consumer prefers more goods to less, and consumption today to consumption tomorrow. Moreover, the consumer gains greater enjoyment from the first set of goods consumed than from the last, and tries very hard indeed to avoid a period of no consumption. Using the function $u(c)$ to denote utility as a function of consumption at a certain time, we denote these common-sense assumptions in the following equations:³

$$\lim_{c \rightarrow \infty} u'(c) = \infty \quad (1.1)$$

$$u'(c) > 0 \quad (1.2)$$

they expect to earn their returns. The standard discounted cash-flow technique values all cash inflows on the basis of net present value of all cash flows, which is a total return method. Users of spreadsheet models, whether they realize it or not, are often creating a NPV or IRR calculation that similarly presumes this definition.

³ We also assume that the function is twice continuously differentiable.

The assumption that the slope of the utility function approached infinity as consumption nears zero, known as the *Inada condition*, is important in these optimization problems. Without it, consumption can turn negative. Recall that this is an intertemporal optimization problem with constraints. One of those constraints is that human beings will not willingly allow themselves to starve.

$$u''(c) < 0. \quad (1.3)$$

We also assume the consumer prefers less risk about the future to more risk. However, we do not impose restrictions beyond these minimal assumptions. In particular, we do not assume away uncertainty, assume “risk neutral” preferences, or even quadratic or other symmetric utility. We also introduce the discount factor β . In many cases this is defined as $1/(1+r)$ or the inverse of the gross rate of return, although we have not restricted it here.⁴ We assume $0 << \beta \leq 1$, with the discount factor often assuming magnitudes of around .85 in applied work.

Consider the simple maximization of utility across time periods, given an income stream y , and a constant gross return on savings (in a generic asset A) of $R = (1+r)$. The consumer faces the following problem: choose consumption today to maximize the discounted benefit of consumption across time periods, subject to a budget constraint that allows him or her to save a portion of the income for next period.⁵ We state these in two equations: a “value” equation that sums the discounted expected utility over time, and a “transition” equation that describes how behavior in one period affects the state of the world in the next period:

$$v(c) = \max_c E_0 \sum_{t=0}^{\infty} \beta^t u(c_t). \quad (1.4)$$

$$A_{t+1} = R(A_t + y - c_t). \quad (1.5)$$

We use the expectation operator to indicate that the exact value of future variables are unknown, and that the consumer or investor makes decisions on an expectation of that value. Unless otherwise stated, we assume the expectation is conditioned only on information available at the current time.

Let’s solve this problem during two time periods, t and $t+1$. If you consume today, you cannot save for tomorrow. However, you prefer consumption today to consumption tomorrow; you also prefer more to less, and you are offered an interest rate on anything you manage to save.

We then use the standard tools of comparative statics, differentiate the equation with respect to the control variable c , and use the first order condition to make the

⁴ This notation is common in the literature using this technique. Note that this is not related to the “beta” of the CAPM model.

⁵ This presentation follows Ljungqvist & Sargent (2005, chapter 1), although without explicitly allowing time-varying interest rates.

A similar, though more mathematically intense, example is the “cake-eating” problem in Stokey & Lucas (1989), which is occasionally used as an example in other texts. In that example, we assume a cake has been given to you, and ask how much should you eat each day.

differential equal to zero. This provides the following rule for optimizing consumption:

$$u'(c_t) = \beta E[Ru'(c_{t+1})] \quad (1.6)$$

Equation (1.6) is known as a *Euler Equation*. It relates the marginal benefit of consuming now with that of saving for future consumption. Euler equations are frequently used in intertemporal optimization models. Note that we did not restrict the discount rate to be the inverse of the gross return ($\beta = R^{-1} = \frac{1}{1+r}$), but if we had, the equation would simplify to consuming today to achieve the same level of satisfaction as you expect to receive (after both discounting for the future, and counting the return on your savings) for the same consumption tomorrow.

IV. The Recursive Approach

Availability of the Recursive Approach

In the example above, we solved the problem for only 2 time periods. Given some additional convenient assumptions (no time-varying income, preferences, discount rates, or interest rates; given initial values of reasonable magnitudes), we can use this optimization repeatedly to attack the entire problem. The key is setting it up as a series of 2-period problems; then repeatedly solving the 2-period problems to arrive at the optimum for the entire problem. This type of calculation is known as a *recursion*; this approach to solving multi-period optimization problems is known as the recursive approach.

Later we will discuss the conditions under which this approach is likely to work, but for now we will assume that the necessary conditions have been fulfilled.

Elements of the Recursive Approach

There are five critical elements of this approach, which we now make explicit:

1. Relating one period to the other is a *transition equation*, which in the example above was the budget constraint (1.5). The transition equation could describe how earnings grow, how a market develops, and how the action of an optimizing agent (such as to invest or spend) affect future conditions.

2. There is a *state* variable that summarizes the results of the last period, and which is available at the beginning of the next period. In the example above, the state was the amount of the savings asset. This state variable normally be affected by the optimizing agent.

Note that the transition equation for the state had only a one-period lag. A time series variable in which all the information useful in predicting the future is summarized in the most recent period is said to have the *Markov property*. As the recursive approach involves a series of 2-period optimization problems, it is essential that all the information necessary for the management (optimization) decision is summarized in the current state variables.

3. The optimizing agent has at least one *control variable* (also known as an action variable) that affects the state, and in most cases the reward. The maximization (or minimization) of the objective function is over this control variable.
4. The optimization problem focused on a *reward function*, which in this case was the utility gained from consumption in a specific time period. The reward function depended on the *action* (or control) taken by the optimizing agent in the model, as well as the state.
5. The entire system can be summarized in a *functional equation* which states the optimization problem in terms of a sequence of reward functions. This is also known as a *Bellman equation*. Note that the Bellman equation is not a regular “function” of one variable; it is an optimization over many possible paths of variables determined by other functions. For this reason, this “function of functions” is known as a “functional.”

These same elements will be repeated in the examples below.

V. Applications

Example 1: Lost Earnings of a Worker

A regular task for forensic economists is to estimate the lost earnings of a worker that, through no fault of his or her own, can no longer work in the same occupation. A simple cash-flow model can sum the cash flows that would be expected under two scenarios over time, and discount them back to the present.

However, the forensic economist is often forced to make (explicit or implicit) assumptions about the optimizing behavior. A common set of assumptions is that the worker would otherwise stay in the same occupation, and get increases at the secular rate that is projected for other workers in that industry. The other risks that an actual worker would face, such as another incapacity, or the need to search for another job, or an opportunity to move or gain another skill, or even that the employer would go bankrupt, are difficult to include explicitly in a simple cash-flow model. Economists may ignore them entirely, use their judgement, or use a proxy calculation. One expects they are sometimes accurate about these, and often not.

As we will see below, the dynamic programming method allows such factors to be made an explicit part of the optimization problem, with potentially better results.

Model for Lost Earnings

State Variables:	Worker's capacity to work; perhaps the labor market conditions, remaining potential work life.
Reward Function:	Current-year wages
Control Variable	Worker's decision to work; in this example we assume the worker decides to work a full number of hours. A larger set of control variables could include training, work hours, and searching for a different job.
Transition Equations:	<p>These would govern growth in wages each year, with perhaps an allowance for training, This could involve some random element. The worker's capacity to work, the market conditions for labor, or other state variables could be in the equation as well, in which case the transition equation must specify how they change.</p> <p>A simple version, allowing for an unusual event to interrupt the workers capacity to earn wages, is (2.2).</p>
Functional Equation:	Using a small set of variables, the

functional equation could be (2.3).

Equations for Example

$$w_{t+1} = s_t \times w_t \times (1 + g); \quad s_0 = 1 \quad (2.1)$$

$$s_{t+1} = s_t \times \varepsilon_t; \quad \varepsilon = \begin{cases} 1, & \text{with probability } 1 - \lambda \\ 0, & \text{with probability } \lambda \end{cases}; \quad (2.2)$$

$$v(s) = \max_s E_0 \sum_{t=0}^T \beta^t w(s_t) \quad (2.3)$$

Here, we use the symbol w for the annual wage, g for the growth rate, s for the state variable (which could be a vector of variables in some applications), and lamda (λ) for the mean arrival rate of an event. We assume $0 \leq \lambda \ll 1$, with the value often being close to 1% in applied work.⁶ The time index t runs from zero to retirement at time T .

In the transition equation shown, an event such as an incapacitating injury creates an absorbing state that does not allow the worker to earn wages in the future. More sophisticated transition equations may allow for a reduction in earnings, a temporary disruption, training to regain lost earning capacity, or the ability to increase earnings in the future through a change in occupation or increase in skills.

Using the Model

To use this model to attack a problem of estimating lost earnings for a worker, we would specify the functions and initial values, and then solve the equations using an initial state value that allowed for full capacity to work. This would generate a current value, discounted for time and risk, of the worker's future earnings. Changing the state variable and solving again would generate a different value, again discounted for time and risk. The difference between the two would be the expected discounted value of the lost earnings.

We will discuss the difficult mathematics of "solving the equations" below.

⁶ A discussion of "jump" processes in business valuation, and their use in modeling events that occur rarely or occasionally, is in Anderson (2004). See also Dixit & Pindyck (1994).

Example 2: Lost Profits for a Business

Another common task of the forensic economist is to estimate lost profits for a business. This problem could be attacked using an approach similar to the following outline:

Model for Lost Profits

State Variables:	The company's capacity to produce; market conditions for the product; contractual agreements.
Reward Function:	Current-year profits, which would be a function of the revenues (determined by the state variables) and management decisions, summarized here as an investment variable. An example is (3.1).
Control Variable	Investment. Note that this affects both current profits, and long-term profits.
Transition Equations:	<p>These would govern changes in market conditions, and therefore the potential revenue of the company. It could involve some random element. As the management decisions (such as investing in new plant or equipment; training; marketing; and other factors) affect the ability to earn revenues in the future, the transition equation should explicitly show how this occurs.</p> <p>A version, which conveniently leaves may interesting details in a unspecified function, is (3.2) and (3.3).</p>
Functional Equation:	Using a small set of variables, the functional equation could be (3.4).

Equations for Example

$$\pi_t = \pi(R_t) - x_t \quad (3.1)$$

$$R_{t+1} = s_t \times R_t \times (1 + g); \quad s_0 = 1 \quad (3.2)$$

$$s_{t+1} = s_t + I(x_t) + u_t \quad (3.3)$$

$$v(s) = \max_x E_0 \sum_{t=0}^T \beta^t \pi(s_t) \quad (3.4)$$

Here, we use the symbol π for the annual profits, which we made a function of revenue R ; g for the secular growth rate in the market; and s for the state variable (which could be a vector of variables in some applications). The time index runs from zero to an arbitrary terminal period T , which could be set to ten years in many circumstances. In the transition equation shown, an exogenous event that changes the state (such as a contract infringement) is shown as the variable u . If this variable takes a nonzero value in any period, it affects the current reward and the values of the state variables. However, it does not necessarily prevent the company from remaining in business.

The functional equation summarizes these relations into a value determined by the maximization of the reward function over time. The reward function, in turn, has the state variable as its argument. Given the state at one point in time, the actions of the agent (given whatever random events occur), determine the reward for this period and the state for the next period.

Using the Model

To apply this method to a problem of lost profits for a business, we would specify the functions and initial values, and then solve the equations using an initial state value. This would generate a current value, discounted for time and risk, of the firm's future earnings. Changing the state variable and solving again would generate a different value, again discounted for time and risk. The difference between the two would be the expected discounted value of the lost earnings.

Example 3: Anti Trust

Now consider a market in which there are few competitors, and two competitors begin to collude. The model for lost profits, above, could be adapted to this situation as described below in Section VI.

Model for Lost Profits

State Variables:	The company's capacity to produce; number of competitors; market-leader pricing.
------------------	--

Reward Function:	Current-year profits, which would be a function of revenues (determined by the state variables) and management decisions (control variables). An example is (4.1).
<hr/>	
Control Variable	Pricing. Note that this affects both current profits, and the state variable (and hence market share) long-term.
<hr/>	
Transition Equations:	This example uses two: The first sets revenue in the next period equal to this period's market share, multiplied by a growth factor and a state variable for the competitive market. The second governs the evolution of the state variable. This includes a function for market share under oligopoly conditions, which we define as a function of price.
	A version, again leaving details in unspecified functions, is (4.2) and (4.3).
<hr/>	
Functional Equation:	Using a small set of variables, the functional equation could be (4.4).

Equations for Example

$$\pi_t = \pi(R_t) - x_t \quad (4.1)$$

$$R_{t+1} = s_t \times R_t \times (1 + g); \quad s_0 = 1 \quad (4.2)$$

$$s_{t+1} = s_t + I(x_t) + u_t \quad (4.3)$$

$$v(s) = \max_x E_0 \sum_{t=0}^T \beta^t \pi(s_t) \quad (4.4)$$

Using the Model

To apply this method to an anti-trust problem, we would we would proceed as in the initial examples: specify the functions and initial values; solve the equations using an initial state value; changing the state variable; solve again.

The challenges of “specifying the functions” and “solving the equations” are significant, and we deal with them below.

VI. Specifying and Solving the Functional Equation

Thus far we have largely ignored the question of specifying and then solving the functional equation and the related variables. We now briefly discuss it.

Much of the development of the technique described in Section VII, Bibliography and History, involved outlining the conditions under which we know that the functional equation has a solution, and that it is unique. Although the examples in this paper are described only generally, given reasonable specifications they could be placed into a functional equation and solved.⁷

Much of the remaining history involves figuring out how to specify the problem so it is tractable. In particular, research by a handful of economists has resulted in a dramatic expansion of the conditions that can be described in the “state” variable.⁸ Finally, a small portion of this history involves creating solution methods for numerically solving the many problems that have not closed-form solution that can be found.

In this paper, we will not describe how to solve functional equation, derive the conditions under which it has a unique solution, or use numerical methods to solve these problems. The section on Bibliography and History outlines the best texts on both the mathematics of determining whether there is a solution, and the computational methods of obtaining it.

However, as we are interested in practical applications, we state two submethods that are, at this point in time, either practical or nearly so.

Practical Method 1: Solving by Hand

The first practical method we term “solving by hand,” because it largely does not involve a computer. The steps for this method are:

⁷ This statement assumes that we choose a bounded transition function; that all the functions involved are continuous; that the reward function is concave; and other limiting assumptions. For the seminal derivation of these requirements, see Stokey & Lucas (1989).

⁸ See the essay “A Brief History of the State” in Ljungqvist and Sargent (1999, preface), or Ljungqvist and Sargent (2004, section 1.4).

1. The economist carefully, but generally, specifies all the functions, state and control variables, and transition rules. The examples presented above illustrate how this can be done.
2. The next step is to describe qualitatively how the state has changed, if necessary revising the equations to match the actual conditions.
3. Finally, the economist hand-draws the results, often in a diagram that shows how the paths of the state or control variable over time could change, and how that would affect the value function. Specific numerical values can be associated with the nodes on each path, if there is other information about them.

We have used a variation of this method in at least two actual consulting engagements. In one case, the variables were primarily qualitative, and the analysis was consistent with a “game theory” approach to the same problem. In the second, there was an absorbing state that we specified, and then illustrated with a cash-flow schedule.⁹

Practical Method 2: Numerical Solutions

This should be termed “nearly practical,” as it is on the edge of being practical for the most motivated practitioner. There are now a very small set of computational routines that can be used on dynamic programming problems.¹⁰ We provide an example in the Appendix of a numerical solution to a business value problem, solved using dynamic programming.

VII. Bibliography and History

The mathematical foundations for the dynamic programming method, the computational ability to perform it practically, and its application to business economics proceeded, with some overlap, along different tracks. We describe each below.

⁹ The schedule was redundant, as it contained vectors of zeroes. However, given the novelty of this approach, we felt it necessary to provide the reader a second presentation in a format similar to traditional finance presentations.

¹⁰ See Section VII.

Mathematics and Mathematical Economics

Bellman (1957) first described the method in a brilliant, slim volume that is useful reading today. Blackwell (1965) provided one of the essential mathematical elements for its practical use: the necessary conditions for a “contraction” mapping to occur.

Its use in financial economics was pioneered by Merton (1973), who used it in his classic derivation of the Intertemporal Capital Asset Pricing Model. Today, the predecessor CAPM model is ubiquitous in finance and business valuation, and is often used incorrectly and without regard to its limitations. As we will see below, the study of dynamic programming for business valuation provides some parallels with Merton’s leap over the then-new orthodoxy of the CAPM model some three decades ago.

After Merton, a handful of mathematically adept economists began researching the technique for broader use.¹¹ One of these was Robert Lucas, Jr., famed for the “Lucas Critique” of Keynesian macroeconomics and for the development of the “rational expectations” model. He collaborated with Nancy Stokey and Edward Prescott to produce the 1989 book *Recursive Methods in Economic Dynamics*, which developed the technique’s mathematical foundations, explicitly stated the necessary conditions for it to provide a unique solution to an optimization problem, and argued that “recursive methods should be a part of every economist’s set of analytical tools.”¹²

Stokey and Lucas’ work brought the method to a wider audience, although still a quite specialized and mathematically adept one. Thomas Sargent collaborated with Lars Ljungqvist in the 1999 book *Recursive Macroeconomic Theory*, which brought the method to a range of economic problems that ranged from asset pricing to rational expectations models involving government policy. In the second edition (Ljungqvist & Sargent, 2004), they entitle the first part of the book “the imperialism of recursive methods,” stating that such methods are replacing others throughout many subfields of economics.

It is interesting to note how recently this tool has been available, even among academic economists. A review of standard “mathematics for economists” texts from the last few decades shows how neither the method, nor much of the fundamental math, was part of the typical graduate economics curriculum, until quite recently. Baumol (1977) contains no mention of the technique, although its treatment of comparative statics is quite good today. Chiang (1967, 1974) similarly does not describe it, although his 1999 book on optimization he describes its use in

¹¹ Before Merton, Blackwell (1965) provided some theoretical basis for its applications.

¹² Stokey & Lucas (1989, preface).

a very accessible way. Simon and Blume (1994), an excellent text used in many graduate schools today, and one that contains a perceptibly higher level of mathematical rigor than its predecessors, similarly do not describe it. A brief survey by the author indicates that those graduate economics courses that teach the method today rely upon the Stokey and Lucas (1989) or Ljungqvist and Sargent (1999, 2004) texts, often for both the mathematics and the application.

Computational

Although the challenging mathematics is a barrier to wider use of the method, the computational difficulties may be the most difficult obstacle to overcome. In the past decade, there has been significant progress. Judd (1998) and Miranda & Fackler (2001) describe numerical techniques to solve dynamic programming and related problems in optimization. The latter text contains a series of examples drawn from the academic literature, and a toolkit to implement them in a mathematical modeling software environment.

VIII. References

- Anderson, Patrick L., 2004. *Business Economics and Finance: Using Matlab, Simulation Models, and GIS*. CRC Press.
- Baumol, William J., 1977. *Economic Theory and Operations Analysis, 4th ed.* Prentice-Hall.
- Bellman, R. 1957. *Dynamic Programming*. Princeton University Press; reissued by Dover, 2003.
- Blackwell, David, 1965. "Discounted Dynamic Programming." *Annals of Mathematical Studies* 36: 226-35.
- Chiang, A.C. 1999. *Elements of Dynamic Optimization*. Waveland Press.
- Chiang, Alpha C., 1974 [1967]. *Fundamental Methods of Mathematical Economics, 2d ed.* McGraw-Hill.
- Damadoran, A., 2002 [1996]. *Investment Valuation, 2d ed.* Wiley.
- Dixit, A., & Pindyck, R. 1994. *Investment Under Uncertainty*. Princeton University Press.
- Duffie, D. 2001. *Dynamic Asset Pricing Theory, 3d ed.* New York: Princeton University Press.
- Miranda, M.J., & Fackler, P.L. 2002. *Applied Computational Economics and Finance*. MIT Press.
- Judd, Kenneth, 1998. *Numerical Methods in Economics*. MIT Press.
- Ljungqvist, L., & Sargent, T.J. 2004, [1999]. *Recursive Macroeconomic Theory*. MIT Press.

Merton, Robert C., 1973. "An Intertemporal Capital Asset Pricing model," *Econometrica* 41, 867-87.

Rustagi, J.S. 1994. *Optimization Techniques in Statistics*. Academic Press.

Simon, Carl P. and L. Blume, 1994. *Mathematics for Economists*. W.W. Norton.

Stokey, N. & Lucas, Jr., R.E. (with Prescott, E.C.) 1989. *Recursive Methods in Economic Dynamics*. Harvard University Press.

IX. Appendix: Example DP Application

This appendix describes a business model created by the author and solved using the CompEcon toolbox described in Miranda & Fackler (2002).

Model Description

1. The business's revenue (the state variable s) grows over time, but is subject to a stochastic revenue shock.
2. The business can make discretionary investments (the action variable x) that reduce distributed earnings, but increase future revenue through a production function.
3. The benefit function (current reward) is the (distributed) earnings of the firm from profits. The profit function is of the form $\pi(s, x) = s - \alpha * s - x$; where s is revenue; α is a cost factor; and x is discretionary investment.
4. The state transition function is of the form $s_{t+1} = \gamma * s + e * [h(x)]$, where e is a lognormal revenue shock with $E(e) = 1$. The drift term γ creates a secular growth path.
5. The production function $h(x) = \beta * x - (x^2 / \beta)$ produces the income which can be distributed to owners, or reinvested. Note that this form creates diminishing marginal returns on investment, which is both economically sound and necessary for the problem to have a unique solution.¹³

Model Equations

$$\pi = \text{div} \times (1 - \alpha) s - x \quad (4.5)$$

$$s_{t+1} = \gamma s_t + \varepsilon (\beta x - \frac{x^2}{\beta}) \quad (4.6)$$

$$v(s_t) = \max_x \{ \pi(s, x) + E[\beta \times v_{t+1}(s, x)] \} \quad (4.7)$$

¹³ One of the conditions outlined by Stokey & Lucas (1989) for the sequence $\{v(s)\}$ to converge is that the reward function is concave and bounded. (Otherwise, it could be possible to create an infinite value.) The concavity of the reward function is a consequence of the diminishing marginal returns. The bounds on the reward function are not shown explicitly here.

$$v_{t+1}(s, x) = v(s_{t+1}, x_{t+1}) \quad (4.8)$$

The value at time $t+1$ is determined by the same functional equation (4.7) as the value in time t , using an updated state variable and new action variable. The state variable has been determined by the transition equation (4.6).

Solving the Functional Equation: Numerical Method

The Bellman equation (4.7) is solved by repeatedly iterating using Newton's method, to solve the functional equation

The CompEcon toolbox routine "dpsolve" is used within the Matlab software environment. This numerical method attempts to solve the underlying value function at a grid of possible state values, then calculates again the optimum actions using the value function results, and compares the two. If there is any difference, the derivatives of the underlying transition and reward function are used to make a new guess for the value function results; the process continues iteratively until convergence is obtained or the machine times out.

The grid of state nodes are selected using two methods: the "Chebyshev" polynomial method, and the linear-quadratic method. In general, the former produces better results while the latter is easier to work analytically.

Exhibit 1 shows the results of these calculations on the maximization policy (choosing the best action variable, given the state).

Exhibit 2 shows the expected time path of wealth (the capital stock, which is the state variable) and investment (the action variable). The mean wealth path seems to be moving toward a steady state of a little less than 300 (top panel). Note this relatively simple specification quickly comes to an optimum action (bottom panel).

Exhibit 3 shows the results of a *monte carlo* experiment, using the random shocks in the model. The mean wealth at the steady-state (convergent path) is about 300, which is the same as shown in the previous exhibit as the mean of the time path of the state. The distribution of results need not be symmetric.

EXHIBIT 1

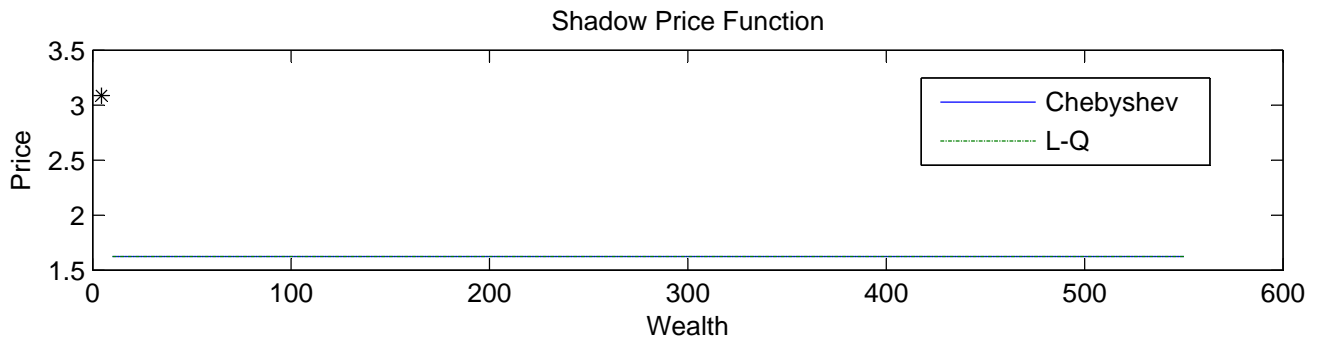
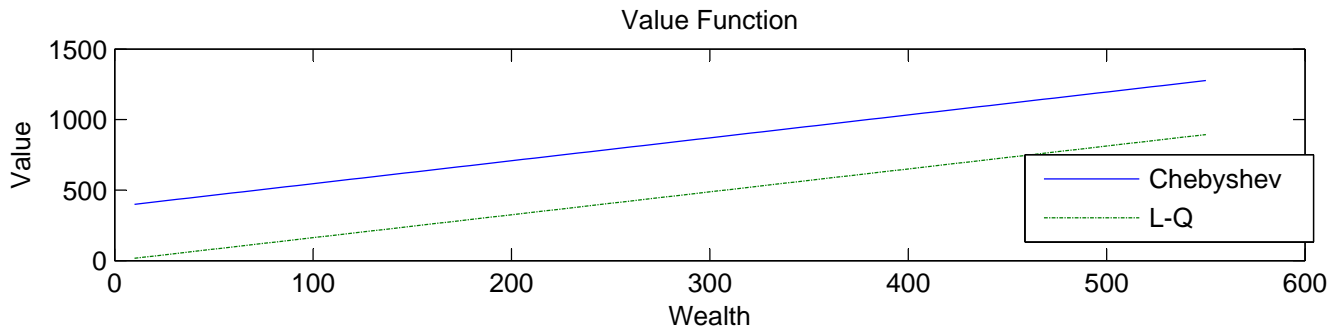
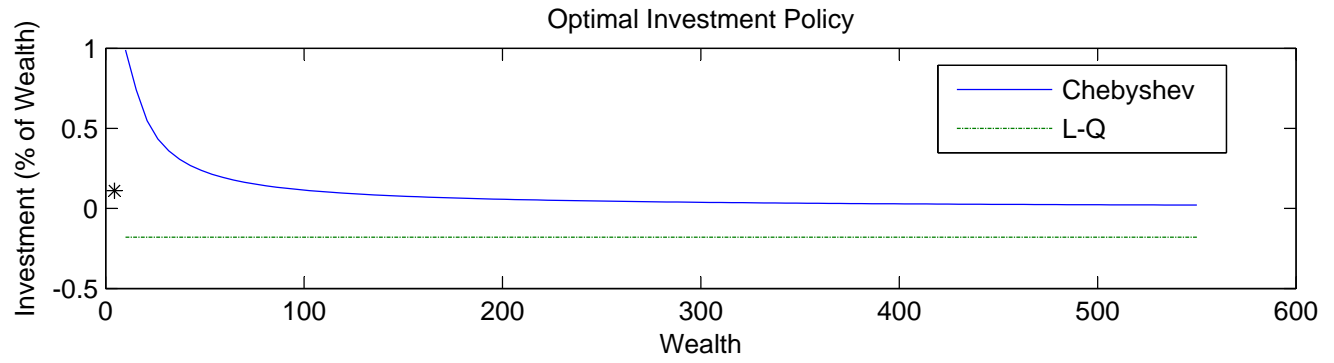


EXHIBIT 2

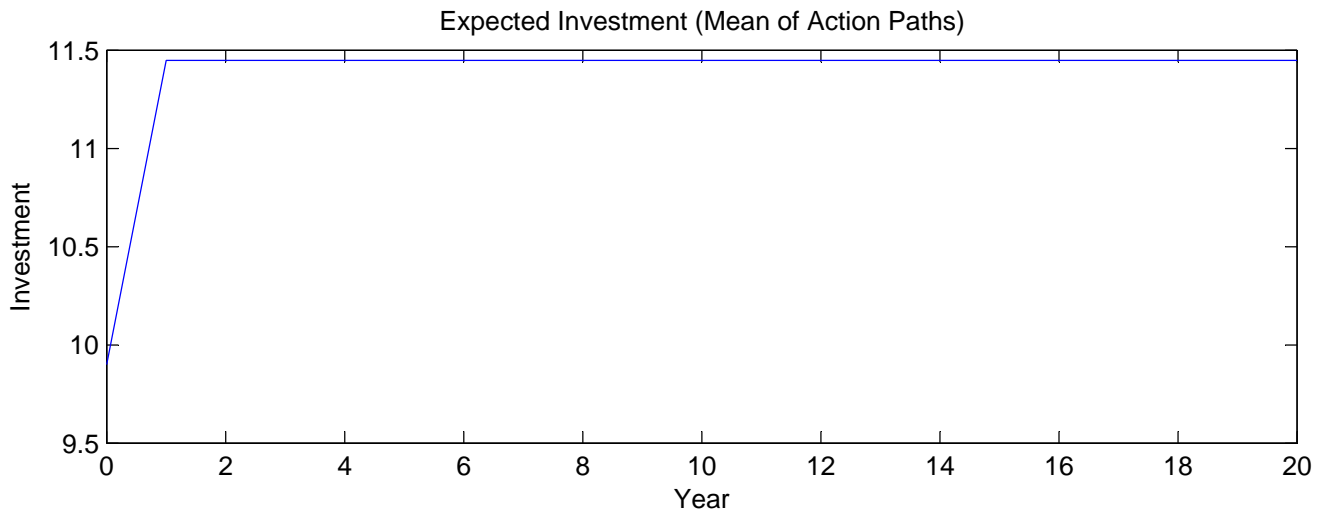
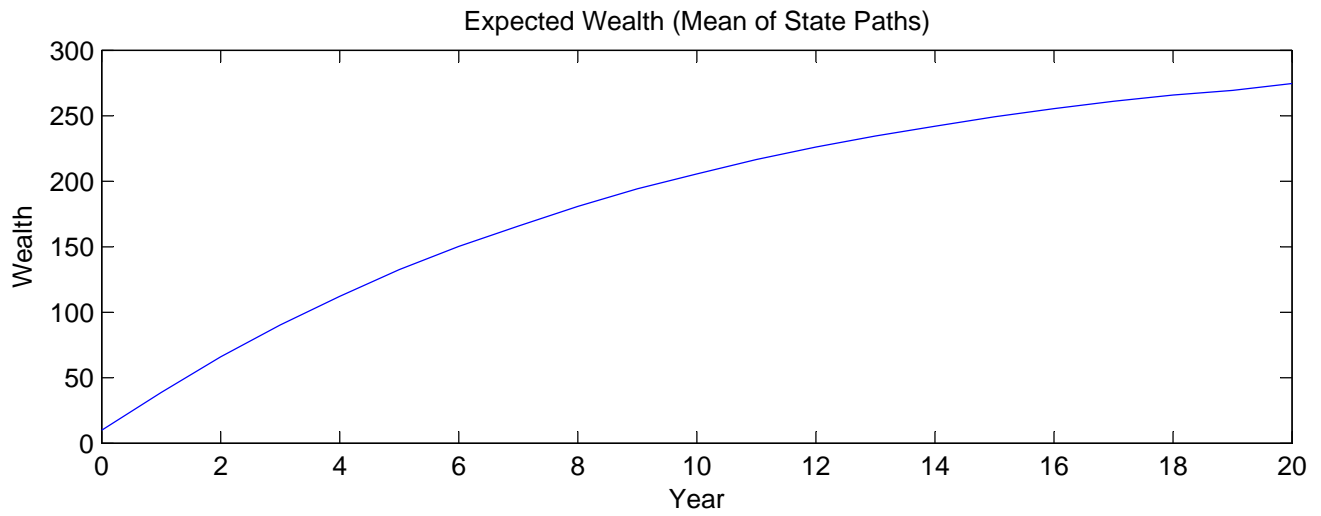


EXHIBIT 3

