Chapter 5 Applied Interest Rate Analysis (1)

- **Investment and Optimization**
  - Several investment problems can be solved via *optimization*.
  - Examples of these problems are capital budgeting, bond portfolio structuring, management of dynamic investments, and firm valuation.
  - An optimization problem is generally concerned with finding the *optimal* solution, $\mathbf{x}^* \in \mathbf{R}^n$, that maximizes a real-valued function, $f : \mathbf{R}^n \rightarrow \mathbf{R}$, over a domain of $\mathbf{R}^n$ defined by a set of *constraints*.
  - That is,

$$\max f(\mathbf{x})$$
$$\text{subject to}$$
$$g_i(\mathbf{x}) \le b_i, \; i = 1, 2, \ldots, m \; .$$

  - If $f(\mathbf{x})$ and $g_i(\mathbf{x})$ are linear then the optimization problem is termed a *linear program*.
  - In general, an optimization problem is called a *mathematical program*.

- **Capital Budgeting**
  - Capital budgeting involves allocating a fixed budget among a set of investments or projects.
  - Usually, there are no well-established markets for these project or investments.
  - They are *lumpy* requiring discrete lumps of cash (as opposed to securities which can be traded in any number of shares).
  - One type of capital budgeting problems is that of selecting from a set of *independent* projects. That is, any subset of projects can be selected if it is within the available budget.
  - Consider a set of m projects. Let c_i and b_i be the initial cost and the present value of project i. Suppose that a total budget of C dollars is available.
  - Define the *decision variables* as x_i, $i = 1,\dots, m$, where $x_i = 1$ if project i is selected, $x_i = 0$, otherwise.
  - Then, the problem is solved with the following *integer-linear program* (*ILP*)

$$\max \sum_{i=1}^{m} b_i x_i$$

$$\text{subject to } \sum_{i=1}^{m} c_i x_i \le C$$

$$x_i = 0,1, \quad i = 1,\dots,m \,.$$

  - The optimal solution to the above ILP can be found using an optimization software (e.g. AMPL, www.ampl.com).

- Microsoft Excel has also a solver module which can handle small ILP problems.

- **Approximate (Heuristic) Capital Budgeting Solutions**
  - Approximate solutions to the capital budgeting ILP is obtained by ranking projects according to *benefit-cost ratio*.
  - This approximation method assumes that each project involves an initial outlay of cash (first cost) followed by a series of benefits.
  - The benefit-cost ratio for project i is the ratio of the *present value of benefits*, $b_i + c_i$ to the initial cost, c_i .
  - The approximate solution is obtained by selecting the project with the highest benefit-cost ratio, then the project with the second highest ratio, and so on, until the budget is exhausted.

- **Capital Budgeting Example**

Example 5.1 (A selection problem) During its annual budget planning meeting, a small computer company has identified several proposals for independent projects that could be initiated in the forthcoming year. These projects include the purchase of equipment, the design of new products, the lease of new facilities, and so forth. The projects all require an initial capital outlay in the coming year. The company management believes that it can make available up to \$500,000 for these projects. The financial aspects of the projects are shown in Table 5.1.

For each project the required initial outlay, the present worth of the benefits (the present value of the remainder of the stream after the initial outlay), and the ratio of these two are shown. The projects are already listed in order of decreasing benefit-cost ratio. According to the approximate method the company would select projects 1, 2, 3, 4, and 5 for a total expenditure of \$370,000 and a total net present value of $\$910{,}000 - \$370{,}000 = \$540{,}000$. However, this solution is not optimal.

TABLE 5.1
PROJECT CHOICES

Project	Outlay ($1,000)	Present worth ($1,000)	Benefit–cost ratio
1	100	300	3.00
2	20	50	2.50
3	150	350	2.33
4	50	110	2.20
5	50	100	2.00
6	150	250	1.67
7	150	200	1.33

The outlays are made immediately, and the present worth is the present value of the future benefits. Projects with a high benefit–cost ratio are desirable.

The proper method of solution is to formulate the problem as a zero–one optimization problem. Accordingly, we define the variables x_i, $i = 1,2,\ldots, 7$, with x_i equal to 1 if it is to be selected and 0 if not. The problem is then

$$\begin{aligned}&\text{maximize } 200x_1 + 30x_2 + 200x_3 + 60x_4 + 50x_5 + 100x_6 + 50x_7\\&\text{subject to } 100x_1 + 20x_2 + 150x_3 + 50x_4 + 50x_5 + 150x_6 + 150x_7 \le 500\\&\qquad\qquad x_i = 0 \text{ or } 1 \quad \text{for each } i.\end{aligned}$$

This can be solved using *Excel Solver*, as shown here.

- **Capital Budgeting with Interdependent Projects**
  - In some situations, the projects are *interdependent*.
  - E.g., assume that there are *m* goals. Goal *i* can be achieved by *one* of n_i project.
  - Define the decision variable as x_{ij}, with $x_{ij} = 1$ if Goal *i* is chosen and implemented with Project *ij*, and $x_{ij} = 0$, otherwise.
  - Then assuming a budget limit of *C* with b_{ij} and c_{ij} denoting the present value and the first cost of Project *ij*, the capital budgeting problem can be stated as follows.

$$\max \sum_{i=1}^{m} \sum_{j=1}^{n_i} b_{ij} x_{ij}$$

$$\text{subject to } \sum_{i=1}^{m} \sum_{j=1}^{n_i} c_{ij} x_{ij} \le C$$

$$\sum_{j=1}^{n_i} x_{ij} \le 1, \quad i = 1, \ldots, m$$

$$x_{ij} = 0,1$$

- An example of capital budgeting with interdependencies is Example 5.2 (text).
- It is a very good exercise for you to implement this example in Excel and solve it with Excel Solver.

- **Further Constraints**
  - The capital budgeting formulation is highly flexible, in the sense that additional project selection requirements can be added easily.
  - For example,
    - Requiring that either Project 1 or Project 2 (or none) is selected is written as $x_1 + x_2 \le 1$,
    - Requiring that Project 1 is not selected unless Project 2 is selected is written as $x_1 \le x_2$,
    - Requiring that Project 1 is not selected if Project 2 is selected is written as $x_1 \le M(1 - x_2)$, where M is a large number. (It is enough to have $M > 1$, here; e.g. $M = 2$.)

- **Issues with capital budgeting**
  - The *hard* budget constraint is inconsistent with the assumption than one can borrow unlimited funds.
  - However, in practice, borrowing limits usually apply (e.g. bank credit line, organizational budget).
  - It is instructive to solve the budget problem with different budget values to measure the sensitivity to the budget level.

Chapter 5 Applied Interest Rate Analysis (2)

- **Optimal Portfolios**
  - The term *optimal portfolio* usually refers to the construction of a portfolio of financial securities.
  - A simple optimal portfolio problem is the *cash matching problem*.
  - This problem involves structuring a bond portfolio to meet a series of future obligations from coupon payment and redemption (face) values.
  - Let $\mathbf{y} = (y_1, y_2, \ldots, y_n)$ be the cash flow stream representing obligations and let $\mathbf{c}_j = (c_{1j}, c_{2j}, \ldots, c_{nj})$ be the cash flow stream associated with Bond j, $j = 1, \ldots, m$.
  - Here y_i and c_{ij} represent cash flows at Time Period i.
  - (There are n time periods and m bonds.)
  - Define also p_j as the price of Bond j.
  - The decision variable is x_j the number of shares of Bond j in the portfolio, $j = 1, \ldots, m$.
  - The cash matching problem can be solved with the following linear program

$$\begin{aligned} \min \quad & \sum_{j=1}^{m} p_j x_j \\ \text{subject to } & \sum_{j=1}^{m} c_{ij} x_j \geq y_i \text{ , } i = 1, \ldots, n \\ & x_j \geq 0, \ j = 1, \ldots, m \end{aligned}$$

- **Cash Matching Example**

> **Example 5.3 (A 6-year match)** We wish to match cash obligations over a 6-year period. We select 10 bonds for this purpose (and for simplicity all accounting is done on a yearly basis). The cash flow structure of each bond is shown in the corresponding column in Table 5.3. Below this column is the bond's current price. For example, the first column represents a 10% bond that matures in 6 years. This bond is selling at 109. The second to last column shows the yearly cash requirements (or obligations) for cash to be generated by the portfolio. We formulate the standard cash matching problem as a linear programming problem and solve for the optimal portfolio. (The solution can be found easily by use of a standard linear programming package such as those available on some spreadsheet programs.) The solution is given in the bottom row of Table 5.3. The actual cash generated by the portfolio is shown in the right-hand column. This column is computed by multiplying each bond column j by its solution value x_j and then summing these results. The minimum total cost of the portfolio is also indicated in the table.

TABLE 5.3
CASH MATCHING EXAMPLE

	Bonds											
Yr	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**Req'd**	**Actual**
1	10	7	8	6	7	5	10	8	7	100	100	171.74
2	10	7	8	6	7	5	10	8	107		200	200.00
3	10	7	8	6	7	5	110	108			800	800.00
4	10	7	8	6	7	105					100	119.34
5	10	7	8	106	107						800	800.00
6	110	107	108								1,200	1,200.00
p	109	94.8	99.5	93.1	97.2	92.9	110	104	102	95.2	2,381.14	
x	0	11.2	0	6.81	0	0	0	6.3	0.28	0	*Cost*	

  - ➢ This example is solved using Excel Solver, here.

- **Issues with cash matching**
  - ➢ One issue with cash matching is that it implicitly assumes that surplus cash flows in a given time period are not *reinvested* (as if surplus is thrown away).
  - ➢ In practice, surplus cash flows are reinvested (in other bonds perhaps).
  - ➢ However, the reinvestment possibility can be accounted for through introducing "artificial" bonds (see text for details).

- **Valuation of a firm**
  - Different cash flow streams can be used to evaluate the worth of a firm. E.g., dividends, net earnings.
  - Different cash flows may lead to different valuations.
  - This kind of analysis also assumes deterministic cash flows which can be problematic.

- **The free cash flow approach**
  - This approach values the firm on the basis of net earnings from the cash flow stream with the maximum present value (maximum over different investing strategies).
  - Suppose the firm earns Y_n each year and decides to invest uY_n in growth. The growth rate is a function $g(u)$.
  - That is, $Y_{n+1} = (1+g(u))Y_n$.
  - Then, the annual capital C_n satisfies, $C_{n+1} = (1-\alpha)C_n + uY_n$, where α is the depreciation rate.
  - It can be shown that

$$Y_n(u) = (1+g(u))^n Y_0 ,$$

$$C_n(u) = (1-\alpha)^n C_0 + uY_0 \left\{ \frac{-(1-\alpha)^n + (1+g(u))^n}{g(u)+\alpha} \right\} \approx \frac{uY_0(1+g(u))^n}{g(u)+\alpha} .$$

  - Then the free cash flow, at a tax rate of T per year is

$$\begin{aligned} FCF_n(u) &= Y_n(u) - T(Y_n(u) - \alpha C_n(u)) - uY_n(u) \\ &= (1-T)Y_n(u) + \alpha T C_n(u) - uY_n(u) . \end{aligned}$$

- Upon simplification,

$$FCF_n(u) \cong \left[1 - T + T\frac{\alpha u}{g(u)+\alpha} - u\right](1+g(u))^n Y_0 .$$

- Then, the value of the firm at a growth investment rate u is

$$PV(u) = \sum_{n=0}^{\infty}\frac{FCF(u)}{(1+r)^n} = \left[1 - T + T\frac{\alpha u}{g(u)+\alpha} - u\right]\frac{Y_0(1+r)}{r-g(u)},$$

assuming $r > g(u)$, and following the same steps as before.

- Finally, *the* value of the firm according to the free cash flow approach is

$$PV = \max_{u \in (0,1)} PV(u) .$$

- **Free Cash Flow Example**

> **Example 5.8 (XX Corporation)** Assume that the XX Corporation has current earnings of $Y_0 = \$10$ million, and the initial capital[2] is $C_0 = \$19.8$ million. The interest rate is $r = 15\%$, the depreciation factor is $\alpha = .10$, and the relation between investment rate and growth rate is $g(u) = .12[1 - e^{5(\alpha - u)}]$. Notice that $g(\alpha) = 0$, reflecting the fact that an investment rate of α times earnings just keeps up with the depreciation of capital.

With a tax rate of 34%, the value of XX Corporation as a function of the growth rate u is

$$PV(u) = \left[0.66 + \frac{0.034u}{g(u)+0.1} - u\right]\frac{10(1.15)}{0.15 - g(u)},$$

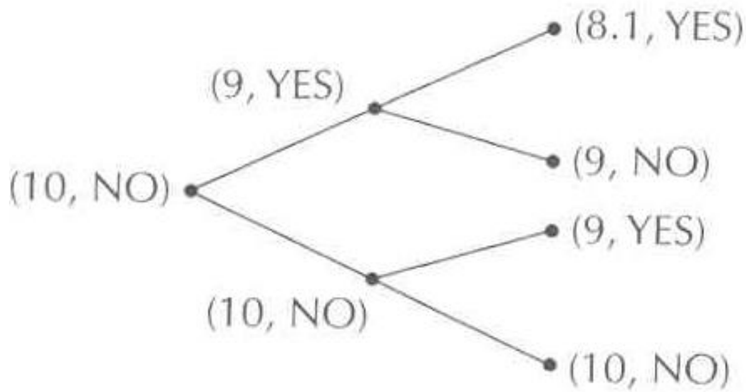
where $g(u) = 0.12[1 - e^{5(0.1-u)}]$.

The optimal value of u that maximizes $PV(u)$ is found using Excel Solver here. The optimal solution is $u^* = 37.8\%$, and the corresponding XX Corp. value is \$67.131 million.

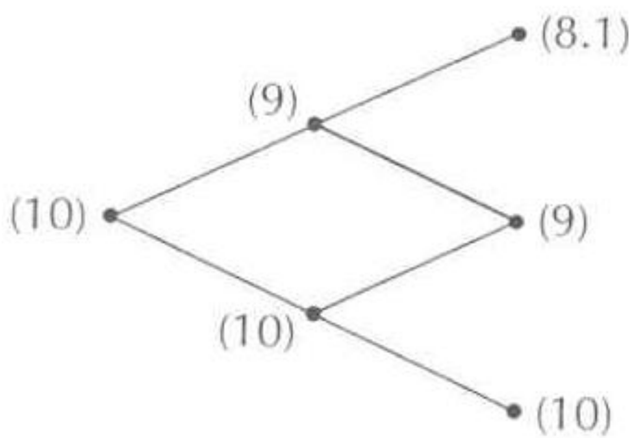
Chapter 5 Applied Interest Rate Analysis (3)

- **Dynamic cash flow process**
  - Many investment problems are not one time decisions. They require ongoing *dynamic* management.
  - A dynamic cash flow process can be described by a cash flow stream $\mathbf{x} = (x_0, \ldots, x_t, \ldots, x_n)$, where x_t, depends on the management actions at times 0, 1,…, t.
  - A dynamic cash process can be represented by a graph (tree).
  - In this graph, *nodes* represent different possible *states* of the process. Each node is defined at a given time period (*stage*).
  - *Arcs* connect nodes between one time and the next.
  - This is a special case of *dynamic programming*.

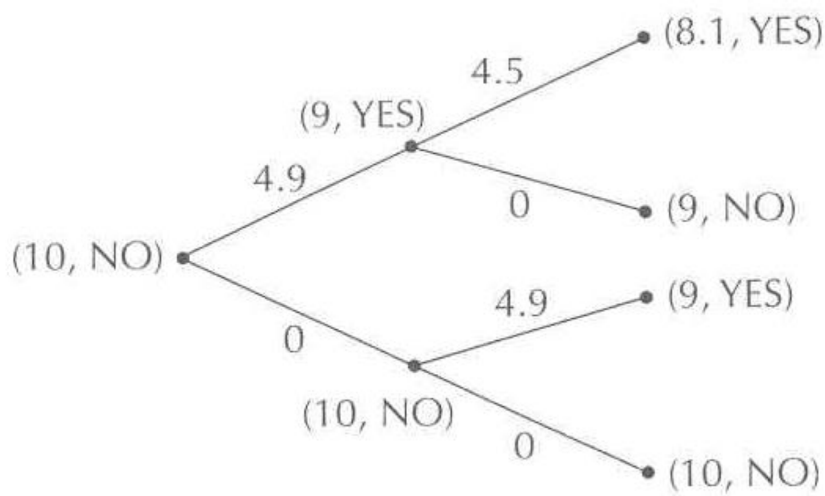
- **Example (investing in an oil well)**
  - A binomial tree is a tree such that exactly two branches leave each node (excluding end nodes).
  - E.g., in an oil well, in each period the management must decide whether to pump 10% of the oil reserve starting with a reserve of 10 M barrels.
  - The state of the system (shown above the nodes) is the number of barrels of oil reserve and whether a new crew of workers has been hired.



- A binomial lattice is a binomial tree where "intermediate" nodes at each stage can be combined.
- E.g., the graph below is a binomial lattice representing the oil well situation where a crew can be hired at no cost.

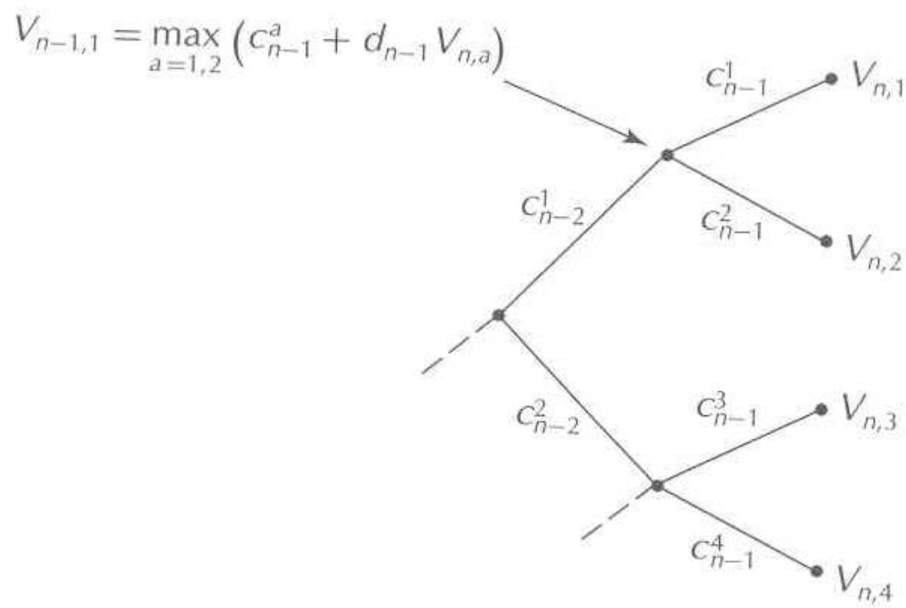


- Cash flows resulting from a given decision are shown above the arc representing the decision.
- E.g., suppose that the cost of hiring a new crew is $100 K (think of this as a recruitment cost). Suppose also that profit from oil production is $5/barrel.

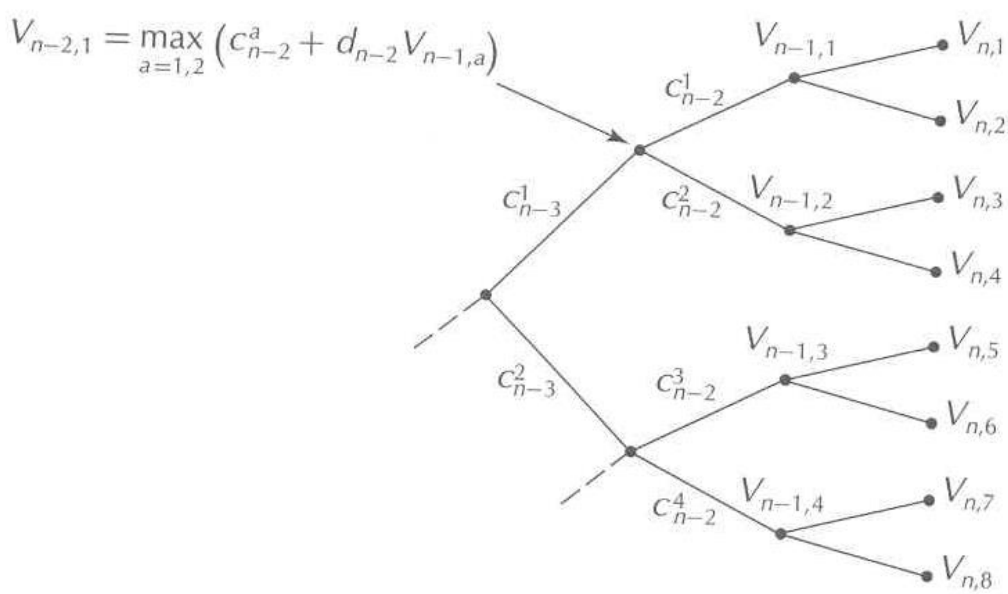


- There could be a final reward or a salvage value associated with a process termination. This is placed on the graph to the right of the final nodes.

- **Dynamic Programming**
  - Once the tree of a cash flow process has been developed, the "optimal path" can be determined by enumerating all possible paths.
  - However, this process is computationally inefficient due to the *curse of dimensionality*.
  - Dynamic programming (DP) is a computational procedure to search for the optimal path efficiently.
  - DP recursively finds the optimal path from each node in the graph to termination.
  - E.g., the optimal decision and present value at time $n-1$, at every node, are determined as follows.



- At time $n-2$, the optimal present value and decisions are determined as follows.



- In general, Denote the value associated with node i at time k by V_{ki}.
- Suppose that a time k the set of decisions is A.

- Let c_{ki}^{a} be the cash flow associated with decision $a \in A$ at node i and time k.
- Then, the DP *optimality equation* is as follows

$$V_{ki} = \max_{a \in A}(c_{ki}^{a} + d_k V_{k+1,a(i)}),$$

where $a(i)$ is the node at time k+1 that the process moves to if decision a is taken, and d_k is the discount factor.
- DP starts with the terminal values, V_{ni}, which are known. It iterates until the *optimal value* V_{k0}, and decisions (path), are found.

- **Dynamic cash flow process example**

> **Example 5.4 (Fishing problem)** Suppose that you own both a lake and a fishing boat as an investment package. You plan to profit by taking fish from the lake. Each season you decide either to fish or not to fish. If you do not fish, the fish population in the lake will flourish, and in fact it will double by the start of the next season. If you do fish, you will extract 70% of the fish that were in the lake at the beginning of the season. The fish that were not caught (and some before they are caught) will reproduce, and the fish population at the beginning of the next season will be the same as at the beginning of the current season. So corresponding to whether you abstain or fish, the fish population will either double or remain the same, and you get either nothing or 70% of the beginning-season fish population. The initial fish population is 10 tons. Your profit is \$1 per ton. The interest rate is constant at 25%, which means that the discount factor is .8 each year. Unfortunately you have only three seasons to fish. The management problem is that of determining in which of those seasons you should fish.
>
> The situation can be described by the binomial lattice shown in Figure 5.8. The nodes are marked with the fish population. A lattice, rather than a tree, is appropriate because only the fish population in the lake is relevant at any time. The manner by which that population was achieved has no effect on future cash flows. The value on a branch indicates the catch (and hence the cash flow) associated with that branch. Horizontal branches correspond to no fishing and no catch, whereas downward directed branches correspond to fishing.
>
> The problem is solved by working backward. We assign the value of 0 to each of the final nodes, since once we are there we can no longer fish. Then at each of the nodes one step from the end we determine the maximum possible cash flow. (Clearly, we fish in every case.) This determines the cash flow received that season, and we

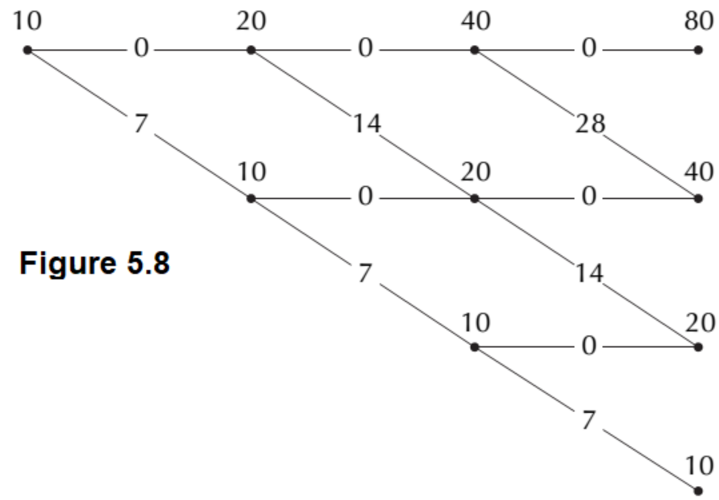


Figure 5.8

assume that we obtain that cash at the beginning of the season. Hence we do *not* discount the profit. The value obtained is the (running) present value, as viewed from that time. These values are indicated on a copy of the lattice in Figure 5.9.

Next we back up one time period and calculate the maximum present values at that time. For example, for the node just to the right of the initial node, we have

$$V = \max\,(.8 \times 28, \quad 14 + .8 \times 14).$$

The maximum is attained by the second choice, corresponding to the downward branch, and hence $V = 14 + .8 \times 14 = 25.2$. The discount rate of $1/1.25 = .8$ is applicable at every stage since the spot rate curve is flat. (See Section 4.6.) Finally, a similar calculation is carried out for the initial node. The value there gives the maximum present value. The optimal path is the path determined by the optimal choices we discovered in the procedure. The optimal path for this example is indicated in Figure 5.9 by the heavy line. In words, the solution is not to fish the first season (to let the fish population increase) and then fish the next two seasons (to harvest the population).

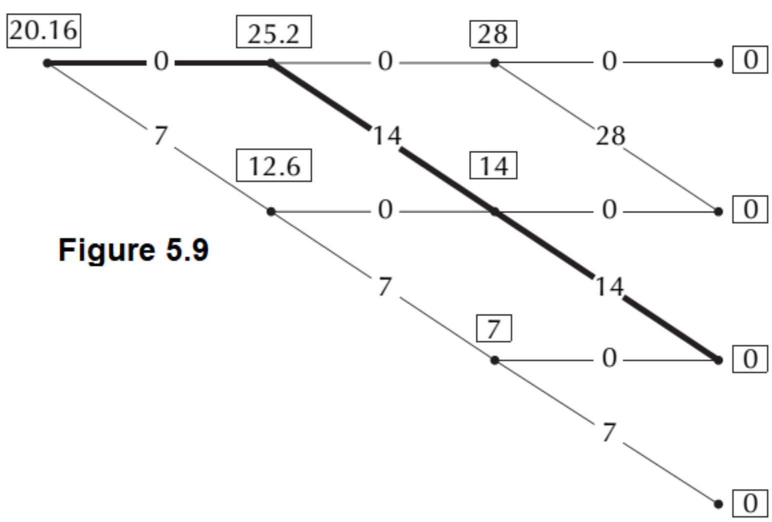


Figure 5.9