日本管理会計学会誌
管理会計学2003年 第12巻 第1号

論 文

Extension of Revenue Accounting:
Using Markov Processes and Dynamic Programming

Yuji Ijiri*
Naoyuki Kaneda†

Abstract

E-Commerce Age needs revenue accounting, oriented toward serving information needs of managers and investors in planning and controlling a firm's sales activities and their financial consequences. We wish to show the revenue accounting proposed in Glover and Ijiri (2002) extended to Markov processes and dynamic programming to gain insight into their processes. In this paper, Markov process was used as a way of capturing the customer transitions and related impact of the corporate profit. We incorporate the possibility of the firm having alternative policies under which transition probabilities and payoffs may be altered, along with an algorithm for an optimal selection of the policies that maximize the long-term profit of the firm.

Keywords

Revenue accounting, Markov processes, Dynamic programming, E-commerce

収益会計の拡張：
マルコフ過程と動的計画法の応用

井尻雄士*
金田直之†

<論文要旨>
Ｅーコマースの時代には、経営管理者や投資家が企業の売上に関して計画・制御するための「収益会計」が必要となる。本研究の目的は、Glover and Ijiri (2002)によって提起された「収益会計」（revenue accounting ）の概念をマルコフ過程と動的計画法を用いて拡張することにある。マルコフ過程は、E-commerce における buyer と browser の推移とその企業利益に与える影響を分析するために用いられる。企業が値引きや広告などの方策をとった場合、推移確率や利益が影響されるが、その可能性も考慮した形で長期の企業利益を最大化する方法を提示する。

<キーワード>
収益会計、マルコフ過程、動的計画法、E-コマース

2003年 3月 7日 受付
2003年 6月24日 受理
*カーネーギーメロン大学産業経営学大学院
†筑波大学社会工学系

Submitted 7, March 2003.
Accepted 24, June 2003.
*Graduate School of Industrial Administration, Carnegie Mellon University
†Institute of Policy and Planning Sciences, University of Tsukuba

1. Introduction

1.1 Revenue Accounting in Contrast to Cost Accounting:

During the last decade, we have seen a shift from product-orientation in the Industrial Age to customer-orientation in the E-Commerce Age. As the Industrial Age needed cost accounting, the E-Commerce Age now needs revenue accounting, oriented toward serving information needs of managers and investors in planning and controlling a firm's sales activities and their financial consequences. Glover and Ijiri (2002) developed a conceptual framework for revenue accounting including tentative postulates of revenue accounting and an analytical framework focusing on revenue mileposts, revenue momentum and sustainability measurements, and intangibles capitalization. Traditional accounting has a large network of cost accounts involving many processes and departments. Yet when it comes to revenues, accounting starts with revenue realization and ends with cash collection, with not many layers of accounts as we see in cost accounting.

In particular, Glover and Ijiri emphasized "revenue milestones" and capture the transition of customers among many states probabilistically. Here a Markov process was used as a way of capturing the customer transitions and related impact of the corporate profit.

1.2. Markov Processes with Payoffs

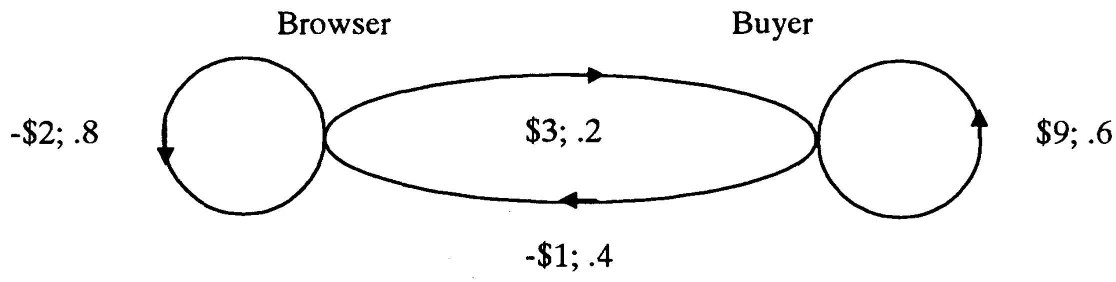
Glover and Ijiri (2002) discusses revenue mileposts and a customer transition between the "browser" state and the "buyer" state by means of a transition matrix. Furthermore, taking advantage of Howard's (1960) model that incorporated a payoff matrix, after each transition of the customer, a payoff amount is assigned depending upon from which state i to which state j the customer moved, including the case i = j, the customer staying at the same state. Providing that the transition matrix is regular, the output of the analysis is that, after a large number of transitions, the probability that the customer is in state i converges to a constant and the payoff the firm can expect from the customer in each transition converges to a constant. In the following, we shall limit our attention to only regular Markov matrices. Non-regular ones are either cyclic or non-ergodic, both of which can be analyzed building upon a set of regular matrices.

1.3. A Browser-Buyer Example

Here, we quote from Glover and Ijiri (2002) with minor modification. "If a customer was a browser in the previous period and is also a browser in the current period, designated by "browser/browser", the cost to the firm is $2 (a -$2 payoff) in the current period, while if s/he was a browser in the previous period and a buyer in the current period (browser/buyer), the benefit to the firm is $3 (a payoff of $3) in the current period. On the other hand, if a customer was a buyer in the previous period and is a browser in the current period (buyer/browser), the cost to the firm is $1 (a payoff of -$1), while if a customer was a buyer in the previous period and is also a buyer in the current period (buyer/buyer), the benefit to the firm is $9 (a payoff of $9). These payoffs along with transition probabilities, which will be explained shortly, are depicted in Figure 1 and summarized in Table 1 in the form of a payoff matrix and a transition matrix.

We now move on to the transition matrix in Table 2. If a customer was a browser in the previous period, there is a .8 probability that s/he will stay as a browser (with no purchase) in the current period and a .2 probability that s/he will become a buyer (with a purchase) in the current period. On the other hand, if a customer was a buyer in the previous period, there is a .4 probability that s/he will become a browser (with no purchase) in the current period and a .6 probability that s/he will stay as a buyer (with a purchase) in the current period (Glover and Ijiri 2002, p. 46.)"

Figure 1: Transition Diagram with Payoffs and Probabilities



The expected payoff given the customer was a browser in the previous period is computed as -$2*.8 +$3*.2 = -$1 and the same for a buyer is -$1*.4 + $9*.6 = $5, as shown in the last column. (Only the expected payoff will be needed in the future computations and not the payoff matrix.)

Table 1: Payoff Matrix and Transition Matrix

		Payoff Matrix		Transition Matrix		Expected
		Browser	Buyer	Browser	Buyer	Payoff
Previous	Browser	-$2	$3	.8	.2	-$1
Period	Buyer	-$1	$9	.4	.6	$5

2. An Extension to Dynamic Programming

2.1. Advertising and Discounting Options

We now want to go further to incorporate, as shown in Howard (1960), the possibility of the firm having alternative policies under which transition probabilities and payoffs may be altered, along with an algorithm for an optimal selection of the policies that maximize the long-term profit of the firm.

Suppose that the firm has an advertising plan that changes the transition probability for a browser from the current (.8 .2) to (.7 .3) but the expected payoff will be worsened from the current -$1 to -$3 as a result of the advertising cost. Similarly, the firm has a discounting plan that changes the transition probability for a buyer from (.4 .6) to (.3 .7), thus improving the repeat purchase rate and increasing the expected payoff from $5 to $6 as a result of price-cut and increased demand. (This is an obvious winner compared with the status quo, since the transition probability to the favorable state is improved without sacrificing, actually improving, the payoff.) The two plans need not be introduced simultaneously. Furthermore, the algorithm allows many different plans to be considered for each state, say 2 plans for a browser and 3 plans for a buyer. For simpler illustration, we shall consider only one new plan for each state along with the current plan as shown in Table 2. The firm wishes to maximize, not the immediate payoff for the current period but, the long-term profit that takes into account the impact on the future transition and the future payoffs.

Table 2: Advertising and Discounting Options

State (Choice)	Options	Transition Probabilities T	Expected Immediate Payoffs w
1. Browser	a: Non-advertising	(.8 .2)	-$1
(a or b)	b: Advertising	(.7 .3)	-$3
2. Buyer	c: Non-discounting	(.4 .6)	$5
(c or d)	d: Discounting	(.3 .7)	$6

2.2. The Policy Iteration Process

In the interest of quickly showing the policy iteration process to get an optimal solution, we shall show the steps in the simplest term, deferring explanations to Sections 2.4.~2.8. The process has 4 components as shown in (1) below, where the policy iteration takes place between **Step 1** and **Step 2** until a certain condition is met, at which time the process ends.

(1) **Start ----> Step 1 <===> Step 2 ----> End**

a) Start by Setting T, w, and v: We shall use T^ to mean the matrix consisting of all optional transition vectors and w^ to mean the expected immediate payoff vector of all optional payoffs in Table 2, namely,

$$T\hat{} = \begin{bmatrix} .8 & .2 \\ .7 & .3 \\ .4 & .6 \\ .3 & .7 \end{bmatrix} \text{ and } w\hat{} = \begin{bmatrix} -1 \\ -3 \\ 5 \\ 6 \end{bmatrix}$$

. Then the starting transition matrix T and the expected immediate payoffs w are created from T^ and w^ by choosing, for each given state, the alternative that has the highest expected immediate payoffs. If there is a tie between two best options for the same state, we choose the one that appeared in the previous policy iteration, if any. Otherwise, we choose any option with the highest value.

For browser, non-advertising (-$1) beats advertising (-$3) and for buyer, discounting ($6) beats non-discounting ($5). Hence, T and w looks as shown in (2). Here, we also use v to mean the expected total payoff vector, which includes the expected immediate payoffs and all the expected payoffs to occur in the future. We set v = w initially.

(2) $$T = \begin{bmatrix} .8 & .2 \\ .3 & .7 \end{bmatrix} \text{ and } w = v = \begin{bmatrix} -1 \\ 6 \end{bmatrix}.$$

b) Step 1, Update v: The policy iteration starts by creating I-T and then replacing the last column of I-T with a unit column vector e = $\begin{bmatrix}1\\1\end{bmatrix}$ (the reason is to be explained later). This matrix is called U and we take the inverse of U as shown below.

(3) $$\text{I-T} = \begin{bmatrix}.2 & -.2\\ -.3 & .3\end{bmatrix}, \text{ hence } U = \begin{bmatrix}.2 & 1\\ -.3 & 1\end{bmatrix} \text{ and } U^{-1} = \begin{bmatrix}2 & -2\\ .6 & .4\end{bmatrix}.$$

Using v' to mean the updated value of v for use in the next policy iteration, we determine v' from v and U^{-1} by:

(4) $$v' = U^{-1}v = \begin{bmatrix}2 & -2\\ .6 & .4\end{bmatrix}\begin{bmatrix}-1\\ 6\end{bmatrix} = \begin{bmatrix}-14\\ 1.8\end{bmatrix}.$$

c) Step 2, Update T and w: Here, we denote by v'^0 a vector obtained from v' by replacing the last element in v' with a zero which is signified by the superscript 0, thus obtaining $\begin{bmatrix}-14\\ 0\end{bmatrix}$ for the above example. Finally, we derive the test quantity u as:

(5) $$u = w\hat{} + T\hat{}v'^0 = \begin{bmatrix}-1\\ -3\\ 5\\ 6\end{bmatrix} + \begin{bmatrix}.8 & .2\\ .7 & .3\\ .4 & .6\\ .3 & .7\end{bmatrix}\begin{bmatrix}-14\\ 0\end{bmatrix} = \begin{bmatrix}-12.2\\ -12.8\\ -.6\\ 1.8\end{bmatrix}.$$

Keeping in mind that the first two rows are for options used for the browsers and the last two rows, for the buyers, we update T and w by selecting the best values in u for each state. Thus, for browsers, non-advertising (-$12.2) is better than advertising (-$12.8); and for buyers discounting ($1.8) is better than non-discounting (-$0.6). We create T' and w', i.e. updated T and w, using choices made in (5) from T^ and w^ for the next round of the policy iteration.

d) End if T'=T and w' = w: If T' = T and w' = w, then we stop the policy iteration. Otherwise, we go back to **Step 1**. In this example, choosing non-advertising and discounting was exactly what we did in the previous round, thus T' = T and w' = w. This signals the fact that no more improvements are available and the policy iteration stops here.

2.3. A Modified Example

While the policy iteration ended after the first try, this is not always the case. To illustrate, it is interesting to see what happens if we reduce the advertising cost by $1, thus improving the expected immediate payoff associated with the advertising option from -$3 to -$2, shown by a * in Table 3.

Table 3: Advertising and Discounting Options (Modified)

State (Choice)	Options	Transition Probabilities T	Expected Immediate Payoffs w
1. Browser	a: Non-advertising	(.8 .2)	-$1
(a or b)	b: Advertising	(.7 .3)	-$2* (modified)
2. Buyer	c: Non-discounting	(.4 .6)	$5
(c or d)	d: Discounting	(.3 .7)	$6

Then, the option selected initially is the same as before, non-advertising (-$1) and discounting ($6). T' (=T) and w' (=w) are thus unchanged. Hence, all derivations stay the same until the iteration process comes to (5), whose second element of w, marked by a *, is changed from -$3 to -$2 as shown below:

(6) $$u = w\hat{} + T\hat{}v'^{0} = \begin{bmatrix} -1 \\ -2^{*} \\ 5 \\ 6 \end{bmatrix} + \begin{bmatrix} .8 & .2 \\ .7 & .3 \\ .4 & .6 \\ .3 & .7 \end{bmatrix} \begin{bmatrix} -14 \\ 0 \end{bmatrix} = \begin{bmatrix} -12.2 \\ -11.8 \\ -.6 \\ 1.8 \end{bmatrix}.$$

As a result of this change, the next policy iteration changes from the previous "non-advertising and discounting" to "advertising and discounting" as advertising (-$11.8) now beats non-advertising (-$12.2). Then, the iteration generates:

(7) $$T = \begin{bmatrix} .7 & .3 \\ .3 & .7 \end{bmatrix} \text{ and } w = \begin{bmatrix} -2 \\ 6 \end{bmatrix}.$$

(8) $$I\text{-}T = \begin{bmatrix} .3 & -.3 \\ -.3 & .3 \end{bmatrix}, \text{ hence } U = \begin{bmatrix} .3 & 1 \\ -.3 & 1 \end{bmatrix} \text{ and } U^{-1} = \begin{bmatrix} 5/3 & -5/3 \\ .5 & .5 \end{bmatrix}.$$

We then obtain an updated v' from v and U^{-1} by:

(9) $$v' = U^{-1}v = \begin{bmatrix} 5/3 & -5/3 \\ .5 & .5 \end{bmatrix} \begin{bmatrix} -2 \\ 6 \end{bmatrix} = \begin{bmatrix} -40/3 \\ 2 \end{bmatrix}.$$

We now compute u as:

(10) $$u = w\hat{} + T\hat{}v'^{0} = \begin{bmatrix} -1 \\ -2 \\ 5 \\ 6 \end{bmatrix} + \begin{bmatrix} .8 & .2 \\ .7 & .3 \\ .4 & .6 \\ .3 & .7 \end{bmatrix} \begin{bmatrix} -40/3 \\ 0 \end{bmatrix} = \begin{bmatrix} -35/3 \\ -34/3 \\ -1/3 \\ 2 \end{bmatrix}.$$

This shows that "advertising and discounting" should be chosen. But this is the same as the options chosen in the previous policy iteration. Hence, the iteration stops. (It is suggested to try other variations in payoffs.)

2.4. Explanations for the Iteration Process

Explanations of the policy iteration process and a proof that the above process does yield an optimum solution is given in Howard (1960, chapters 2~4). We simplified the explanations of the policy iteration process by incorporating all key elements of iteration in "composite" vectors and matrices. We must now, however, explain the reason for the insertion of a unit vector in T's and setting of the last element of v equal to zero.

For this purpose, we use as an example the particular T and w given earlier by:

(2) $$T = \begin{bmatrix} .8 & .2 \\ .3 & .7 \end{bmatrix} \text{ and } w = v = \begin{bmatrix} -1 \\ 6 \end{bmatrix}.$$

We define v(n) to be the value of v if the Markov iteration process is to terminate n periods from now, and apply the backward Markov iteration starting with n = 0.

A caution at this point might be in order since we have two kinds of iterations involved here. The policy iteration process discussed earlier in Section 2.2 changes the values of T, w, and v at its each iteration. The Markov iteration process to be discussed here involves iterations under given T, w, and v, chosen at a particular round of the policy iteration.

2.5. Backward Markov Iterations

Table 4 below shows how v(n) changes as the Markov process moves backwards, along with Figure 2 which depicts the data in the second and the third columns of Table 4. At n = 0, the system has ended and has no more payoffs to generate, hence $v(0) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$. This means that at n = 1, only the expected immediate payoff $w = v(1) = \begin{bmatrix} -1 \\ 6 \end{bmatrix}$ is available. At n = 2, the system will have Tv(1) from the operation in n = 1 plus the expected immediate payoffs of w, thus $v(2) = w + Tv(1) = \begin{bmatrix} -1 \\ 6 \end{bmatrix} + \begin{bmatrix} .8 & .2 \\ .3 & .7 \end{bmatrix} \begin{bmatrix} -1 \\ 6 \end{bmatrix} = \begin{bmatrix} -.6 \\ 9.9 \end{bmatrix}$, or in general we have:

(11) $$v(n) = w + Tv(n-1).$$

Let us now examine Table 4 which consists of 3 groups of two columns each, setting aside the column for n. The first and second columns show the values of v(n) for n = 0, 1, 2, ..., 12 for browsers and buyers and it can be easily verified that the above numbers derived for n = 0, 1, and 2 agree with those in the table. Thus from the last row, if the system has 12 more periods before its termination, a browser at that time has the expected total payoffs of \$16.001 and a buyer, \$29.998, for the firm. (See also Figure 2 below.)

Skipping the next two columns of Table 4 for now, we move to the last two columns of the table that show the amount of increment over the previous period computed for browsers and for buyers. As clearly shown in the table, the increment converges to $1.80 regardless of whether the customer starts out as a browser or as a buyer. The convergence is assured by the property of regular Markov processes, no matter which state the customer starts from. We let g to mean this limit amount of payoffs $1.80 that will be repeated in each period regardless of the state, namely it is the "state-independent payoff."

Table 4: Values of v_t and Δv_t after Each Backward Markov Iteration

	---- Value v(n) -----		---- Relative Value $v^0(n)$ ----		---$\Delta v(n)=v(n)-v(n-1)\rightarrow g$ ---	
n	Browser	Buyer	Browser	Buyer	Browser	Buyer
0	0	0	0	0		
1	-1	6	-7	0	-1	6
2	-0.6	9.9	-10.5	0	0.4	3.9
3	0.5	12.75	-12.25	0	1.1	2.85
4	1.95	15.075	-13.125	0	1.45	2.325
5	3.575	17.138	-13.563	0	1.625	2.063
6	5.288	19.069	-13.781	0	1.713	1.931
7	7.044	20.934	-13.891	0	1.756	1.866
8	8.822	22.767	-13.945	0	1.778	1.833
9	10.611	24.584	-13.973	0	1.789	1.816
10	12.405	26.392	-13.986	0	1.795	1.808
11	14.203	28.196	-13.993	0	1.797	1.804
12	16.001	29.998	-13.997	0	1.799	1.802
	Limit		h = -14.000		g = 1.800	g = 1.800

2.6. Relative Values and Asymptotes

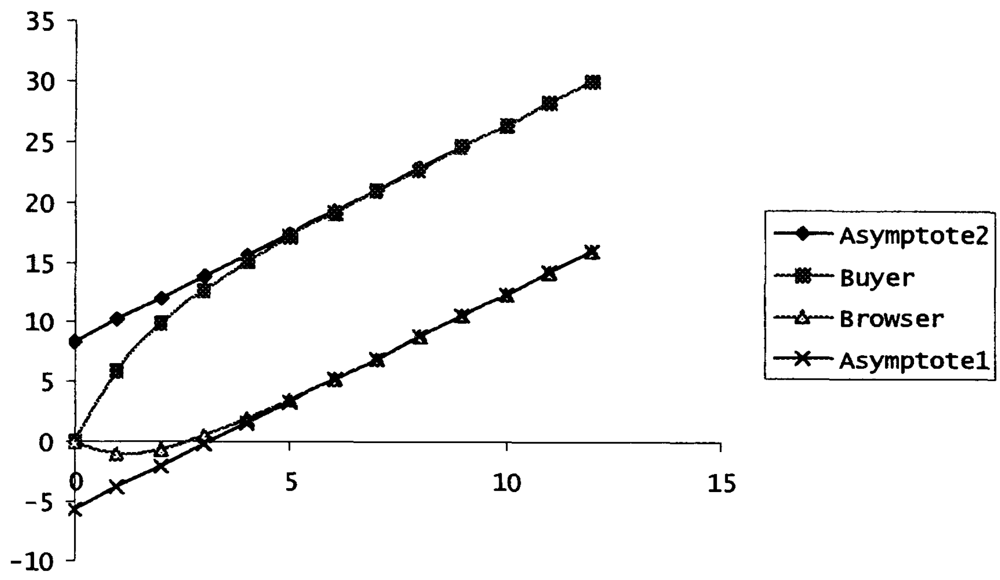
The middle 2 columns of Table 4 which are yet to be explained are the column of $v_1^0(n) = v_1(n) - v_2(n)$ for browsers and $v_2^0(n) = v_2(n) - v_2(n) = 0$ for buyers. By the definition of the superscript 0, values are expressed relative to the value of the last state. For optimization purposes, it is not necessary to keep track of v(n)'s for each and every state in the system. We can set v_i for any state i equal to zero, expressing payoffs for all other states as "relative payoffs," compared to the selected state's payoffs. For example, we can set the payoffs for buyers equal to zero and state the relative payoffs for browsers, which is what is shown in the middle 2 columns in Table 4. As evident from the table, this relative payoffs, denoted by h, for browsers converge to -$14.00.

We also note from Table 4 that v(n) approaches the two asymptotes shown in Figure 2. The lower line is for browsers and the upper line is for buyers. The asymptotes intercept with the y-axis at -5.6 for browsers and at 8.4 for buyers. The slope of both asymptotes are 1.8 per each increment in n, the Markov iteration number. The two asymptotes are expressed as Equation (12) below which form good approximations to v(n) for any large n.

(12) $$v(n) \cong nge + v = \begin{bmatrix} 1.8n \\ 1.8n \end{bmatrix} + \begin{bmatrix} -5.6 \\ 8.4 \end{bmatrix},$$

where e is a unit column vector and v is a vector of constants indicating the intercepts of the asymptotes.

Figure 2: Values of v(n) for n = 0, 1, ..., 12 in Table 4 and Their Asymptotes



The optimization routine attempts to maximize (12), as after a large enough number of iterations, any errors from this use of asymptotes diminishes. In this way, the optimization process involves only linear functions, hence the above policy iteration process does lead to the optimum solution.

The constants in (12) can be explained using the property of regular transition matrices that can be decomposed into the permanent part and the transient part. In particular, the transition matrix T in (12) can be stated as:

(13) $$T^n = \begin{bmatrix} .6 & .4 \\ .6 & .4 \end{bmatrix} + (.5)^n \begin{bmatrix} .4 & -.4 \\ -.6 & .6 \end{bmatrix}.$$

Note that for n =0, $T^0 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ and for n=1, $T^1 = \begin{bmatrix} .6 & .4 \\ .6 & .4 \end{bmatrix} + (.5)^1 \begin{bmatrix} .4 & -.4 \\ -.6 & .6 \end{bmatrix} = \begin{bmatrix} .8 & .2 \\ .3 & .7 \end{bmatrix}$ as it should be. If we multiply T^n by the expected immediate payoff vector w, we obtain:

(14) $$T^n w = \begin{bmatrix} .6 & .4 \\ .6 & .4 \end{bmatrix} \begin{bmatrix} -1 \\ 6 \end{bmatrix} + (.5)^n \begin{bmatrix} .4 & -.4 \\ -.6 & .6 \end{bmatrix} \begin{bmatrix} -1 \\ 6 \end{bmatrix} = \begin{bmatrix} 1.8 \\ 1.8 \end{bmatrix} + (.5)^n \begin{bmatrix} -2.8 \\ 4.2 \end{bmatrix}.$$

The transient component summed from n = 0 to ∞ is $(.5^0 + .5^1 + ...) \begin{bmatrix} -2.8 \\ 4.2 \end{bmatrix} = 2 \begin{bmatrix} -2.8 \\ 4.2 \end{bmatrix} = \begin{bmatrix} -5.6 \\ 8.4 \end{bmatrix}$, hence the meaning of the constants in (12) becomes clear.

2.7. Composite Matrix U and Composite Vector v^g

We now explain the reason for the composite matrix U used in Section 2.2. Assuming that n is large, we equate Equations (11) and (12) as follows.

(11) $v(n) = w + Tv(n\text{-}1)$,

(12) $v(n) = nge + v$, which also means,

(15) $v(n\text{-}1) = (n\text{-}1)ge + v$.

Hence putting (15) into the right-hand side of (11), we obtain:

(16) $v(n) = w + T[(n\text{-}1)ge + v] = w + [(n\text{-}1)g]Te + Tv = w + (n\text{-}1)ge + Tv$,

for Te = e because T is a transition matrix whose each row sums to 1.

Equating (12) and (16), we have:

(17) $nge + v = w + (n\text{-}1)ge + Tv$,

thus, transferring (n-1)ge + Tv to the left and simplifying, we obtain:

(18) $(I\text{-}T)v + ge = w$.

We now show that:

(19) $(I\text{-}T)v + ge = Uv^g$,

where U is (I-T) whose last column is replaced by a vector of all 1's and v^g is the vector v whose last element is replaced by g. Remembering that we focus on the relative values in v, relative to the value of the last element which is, therefore, set equal to zero. The equality in (19) should be clear from the 3x3 matrix below:

(20) $$\begin{bmatrix} a11 & a12 & a13 \\ a21 & a22 & a23 \\ a31 & a32 & a33 \end{bmatrix} \begin{bmatrix} v1 \\ v2 \\ 0 \end{bmatrix} + \begin{bmatrix} g \\ g \\ g \end{bmatrix} = \begin{bmatrix} a11 & a12 & 1 \\ a21 & a22 & 1 \\ a31 & a32 & 1 \end{bmatrix} \begin{bmatrix} v1 \\ v2 \\ g \end{bmatrix}.$$

Thus (18) is equivalent to:

(21) $Uv^g = w$.

In this way, when (21) is solved for v^g by multiplying both sides of (18) by U^{-1}, we see that the solution is obtained indeed by v^g with all solutions obtained in one matrix inversion:

(22) $v^g = U^{-1}w$.

We said earlier that the optimization process was carried out to maximize:

(12) $$v(n) = nge + v = \begin{bmatrix} 1.8n \\ 1.8n \end{bmatrix} + \begin{bmatrix} -5.6 \\ 8.4 \end{bmatrix},$$

but actually it was carried out to maximize:

(23) $$v(n) = nge + v = \begin{bmatrix} 1.8n \\ 1.8n \end{bmatrix} + \begin{bmatrix} -14 \\ 0 \end{bmatrix},$$

since as mentioned earlier in Section 2.6 the constant term that is common to all states, 8.4, is irrelevant to the optimization process. We have postponed this point until now to avoid the need to define another variable.

2.8. Interpretations

We can now interpret v = $\begin{bmatrix} -14 \\ 1.8 \end{bmatrix}$ obtained in (4) which is also shown in the last row of Table 4. Its first element (-$14) is h which is the relative payoff, the expected total payoffs if the system started from the first state, the browser in comparison to the buyer. The second element ($1.8) is g which we discussed earlier in Section 2.5 and indicates the state-independent payoff per each transition after the system has been run for a sufficiently large number of times.

Equation (5) then shows the test quantity u computed as the sum of the immediate payoff w and the total expected payoffs in the future.

Also after changing the payoff under advertising from -$3 to -$2, we find in (9) that the expected total payoff after starting in the first state is -$40/3, while the state-independent payoffs is $1.8. Then, the test quantity is computed in (10) and the best options are chosen as "advertising and discounting." This choice is repeated at the next round and the iteration stops as we saw before.

2.9. Optimization and Sensitivity Analyses

The above analysis indicates that Markov processes have been enriched greatly by the introduction of payoffs, options, and dynamic processes. Sensitivity analyses can be carried out as an aid to policy and strategy decisions. The value data such as v and g gives management a base to determine maximum or minimum offers they can make to let the system starts at a preferred state such as the "buyer" state instead of a less preferred state such as the "browser" state.

Although the above process does not consider the time value of money, it can easily be incorporated. For example, knowing the value of the state-independent payoff g per iteration and the interest rate r per iteration period, the present value of future cash flows may be determined as g/r focusing only on the state-independent component. It can then be fine-tuned by incorporating the state-dependent values v.

2.10. The Duality

We now consider the duality of the dynamic programming system discussed in the above. It is clear that the above optimization method can be applied intact to the transition matrix along with options and payoffs. Options that may be considered include an introduction of a higher quality process that produces more preferred output, a cost saving process that improves yield, sub-contracting, etc.

To search for an optimum method of going backwards does make sense if the underlying production process is time-reversible. Even in cases where this is not the case, a backward search for optimum can make sense when we wish to find a minimum cost to produce a given amount of output, to allocate revenues back to the original process that are responsible for their production, or to reverseengineer, for example, a product mix that will meet certain conditions. This indeed explains the enormous power of dynamic programming.

The policy iteration method discussed above obviously assumes that options represented by each row of T^ and w^ can be picked up independent of each other and assembled in a meaningful way. For example, in our illustration using the browser-buyer example, we had non-advertising and advertising options for browsers and non-discounting and discounting options for buyers. We assumed that the choice of option we make for browsers does not affect the choice we make for buyers. This may not always be the case in practice, since, for example, a decision to advertise may not be implemented just for browsers and a decision to discount may not be implemented just for buyers. If this is the case, some creative ways of rethinking about options and states in the transition matrix may be required.

3. Conclusions

In this way, we have extended revenue accounting proposed in Glover and Ijiri (2002) into Markov processes and dynamic programming models. The application of Markov processes and dynamic programming is a convenient tool to find the solutions that maximize the expected total payoffs. As already mentioned in section 2.1, the algorithm introduced in this paper is applicable for multiple options for each state. The algorithm also allows different number of plans to be considered for each state. In other words, we may apply this methods to the situation where we have more than three states and more than three different options. In this paper, we described the simplest example for the illustration purposes.

References

Glover, J.C. and Y. Ijiri, 2000, "'Revenue Accounting' in the Age of E-Commerce: A Framework for Conceptual, Analytical, and Exchange Rate Considerations," *Journal of International Financial Management and Accounting*, 13 (1), pp. 32-72.

Howard, R.A., 1960, *Dynamic Programming and Markov Processes*, MIT Technology Press and John Wiley & Sons, New York.