Optimal Decision Making Systems in Manufacturing

Roger J. Wallace

School of Management Information Systems, Deakin University, Burwood, Vic., 3125, Australia,
E-mail: rwallace@deakin.edu.au

ABSTRACT: *In this introductory paper, the author identifies new optimal decision making problems in manufacturing. These arise from certain multi-stage allocation processes in production, and entail maximizing the probability of reaching nominated production targets under risk. It is established that such problems can be modelled by particular dynamic programming difference systems. These systems are investigated for various values of the process parameters. Several conclusions are reached, and future research directions are indicated. The main outcome is a cost-effective approach to practical problems in manufacturing.*

Keywords: Optimal decision making, manufacturing, dynamic programming, difference systems.

INTRODUCTION

Background

Optimal decision making systems are mathematical *models* of business problems in *minimization* or *maximization*. For over 50 years, *techniques* for the *solution* and *implementation* of such models have been regularly introduced in the literature, and then applied commercially. For example:

- *Linear programming*, in allocation of scarce resources. Chapters 14 and 15 of Albright et al. (1999) discuss the landmark 1947 researches of G.B. Dantzig, and describe many current applications of this tool—in areas as diverse as plate glass production at Libbey-Owens-Ford, bond selection on Wall Street and costing at Monsanto; see also the paper on timetabling by Birbas et al. (1997).
- *CPM (Critical Path Method)* and *PERT (Program Evaluation and Review Technique)*, in project management. These related methods were developed, independently and simultaneously, by various workers in the late 1950s. Chapter 11 of Taylor (1999) outlines the beginnings of *CPM* and *PERT*, and provides examples of their recent use—in team training at IBM, the Benfield repair project at Sasol and facility relocation at Rockwell; see also the findings of Phillips (1996) on network flow.
- *Dynamic programming*, in sequential decision processes. *Dynamic* programming is quite different from *linear* programming, above. Denardo (1982) describes R.E. Bellman's original 1952 paper, and Chapters 20 and 21 of Winston (1991) relate how dynamic programming has solved many modern-day problems—particularly in equipment replacement at Phillips Petroleum and dynamic lot-sizing in large-scale US production centres; see also the text of Esogbue (1989) on optimal resource systems.

Synopsis

In this introductory paper, the author identifies a new *class* of optimal decision making problems in *manufacturing*. These problems arise from certain *multi-stage* allocation processes in production, and entail *maximizing* the *probability* of reaching nominated production targets *under risk*. It is established that such problems can be modelled by a particular class of dynamic programming *difference systems*. These systems are investigated for various values of the process parameters. Several conclusions are reached, and future research directions are indicated.

The author's technique augments other approaches to related problems, notably those of Iwamoto (1985) and Jakubowski et al. (1985) (in *allocation processes*) and Ratz and Russell (1987) (in *random walks*); see also the findings of Wallace (1984, 1987a, 1987b, 1999, 2000a, 2000b) (in *optimal sequential search*). The main outcome here is a *cost-effective* approach to practical problems in manufacturing.

THE PROBLEMS

Background

In a certain factory, *all* machines are designed to perform the *one* routine *task*, and *each* machine is one of *two* different *types* (Type A or Type B). Every day, the factory manager chooses some of these machines to become components of two *teams* (Team A and Team B), that will then, *separately*, perform the task. Team A consists *solely* of Type A machines; Team B comprises*only* Type B machines.

- Production issues and other considerations require that (a) the *total* number of machines in the two teams *combined* (N) be *constant*, and that (b) *both* machine types be represented. Therefore, each day, the manager has a choice of $N - 1$ allocation *alternatives* $A(1)$, $A(2)$, $A(3)$, $\cdots$, $A(i)$, $\cdots$, $A(N - 1)$; in which, with alternative $A(i)$ $(0 < i < N)$, there are i machines in Team A, and $N - i$ in Team B.
- The nature of the task is such that, for each team, *all* its components must perform the task satisfactorily for there to be an output. The *gain* then is equal to the *number* of components in the team. Any other outcome is termed a *foul*, and yields a gain of *zero*.
- Historically, a known *proportion* α of Type A machines has performed the task satisfactorily, as has a known (larger) *proportion* β of Type B machines. Moreover, it is assumed that, in regard to task performance, components of Team A are *identical*, as are those from Team B. It is also assumed that the performance of any component of either team is *independent* of that of any *other* team component, and of that of any component of the *other* team.

Objective

The manager's *objective* is to determine the so-called *optimal policy*; namely, that particular *sequence* of daily allocation alternatives which *maximizes* the *probability* of reaching a prescribed number (n) of units of gain, *before* both teams foul simultaneously.

Methodology

In the next section, it will be established that the manager's *problems* can be modelled by a particular *class* of dynamic programming *difference systems*. Subsequently, it will be shown *how* these systems can be analyzed to achieve the manager's *objective*.

THE MODEL

Henceforth, for given α, β $(0 < \alpha < \beta < 1)$ and prescribed integers N, n $(N \geq 3,\ n \geq 0)$, let M denote the *integer part* of $(N - 1)/2$, let $P(\alpha, \beta; N; n)$ denote the aforementioned *maximum probability*, and abbreviate $P(\alpha, \beta; N; n)$ to $P(n)$, without loss of generality.

Theorem 1

(a) $P(0) = 1$, whereas $P(1)$, $P(2)$, $P(3)$, $\cdots$, $P(M)$ are given by the following *initial conditions* (1a):

For fixed n in $0 < n \leq M$,

$$P(n) = \max_i \begin{cases} \alpha^i (1 - \beta^{N-i})\, P(n - i), & \text{for all } i \text{ in } 0 < i \leq n; \\ (1 - \alpha^i)\, \beta^{N-i} P(n - N + i), & \text{for all } i \text{ in } N - n \leq i < N. \end{cases} \tag{1a}$$

(b) $P(M + 1)$, $P(M + 2)$, $P(M + 3)$, $\cdots$, $P(N - 1)$ are determined by the following *initial conditions* (1b):

For fixed n in $M < n < N$,

$$P(n) = \max_i \begin{cases} \alpha^i (1-\beta^{N-i})\, P(n-i), & \text{for all } i \text{ in } 0 < i < N-n; \\ \alpha^i (1-\beta^{N-i})\, P(n-i) + (1-\alpha^i)\,\beta^{N-i} P(n-N+i), & \text{for all } i \text{ in } N-n \le i \le n; \\ (1-\alpha^i)\,\beta^{N-i}\, P(n-N+i), & \text{for all } i \text{ in } n < i < N. \end{cases} \tag{1b}$$

(c) $P(N)$, $P(N+1)$, $P(N+2)$, ··· satisfy the following dynamic programming *difference equation* (1c):

For fixed $n \ge N$,

$$P(n) = \max_i \{ \alpha^i (1-\beta^{N-i})\, P(n-i) + (1-\alpha^i)\,\beta^{N-i} P(n-N+i) + \alpha^i \beta^{N-i} P(n-N) \}, \quad \text{for all } i \text{ in } 0 < i < N. \tag{1c}$$

Proof

It will prove expedient to consider *separately* the cases N *odd* and N *even*.

Case (i): N odd

It is obvious that $P(0) = 1$. To prove (1a, b, c), first recall that, with *alternative* $A(i)$ ($0 < i < N$), there are i machines in Team A, and $N - i$ in Team B. Accordingly, associated with $A(i)$, there are only *three* (mutually exclusive) ways of obtaining a *non-zero* gain:

- When *all* components of Team A perform the task satisfactorily, *and*, simultaneously, *at least one* component of Team B does *not*. Denote this *event* by $E_A(i)$. With $E_A(i)$, therefore, there is a gain of i units, and the probability of $E_A(i)$ occurring is $\alpha^i(1-\beta^{N-i})$.
- When, conversely, *all* components of Team B perform the task satisfactorily, *and*, simultaneously, *at least one* component of Team A does *not*. Denote this event by $E_B(i)$. With $E_B(i)$, therefore, there is a gain of $N - i$ units, and the probability of $E_B(i)$ occurring is $(1-\alpha^i)\beta^{N-i}$. (Note that, here, with N *odd*, it is *never* the case that $i = N - i$; hence, $E_A(i)$ and $E_B(i)$ *never* coincide.)
- When *all* components of Team A perform the task satisfactorily, *and*, simultaneously, *all* components of Team B *also* do. Denote this event by $E_{AB}(i)$. With $E_{AB}(i)$, therefore, there is a gain of N units, and the probability of $E_{AB}(i)$ occurring is $\alpha^i\beta^{N-i}$.

Next, for fixed i in $0 < i < N$, let $p(i, j)$ denote the *probability* of gaining j ($0 < j \le N$) units of gain, by choice of alternative $A(i)$. From the conclusions on $E_A(i)$, $E_B(i)$ and $E_{AB}(i)$, above, it follows that:

For all i in $0 < i < N$ and all j in $0 < j \le N$, $p(i, j) = 0$, except that

- $p(i, i) = \alpha^i(1-\beta^{N-i})$. (2a)
- $p(i, N-i) = (1-\alpha^i)\,\beta^{N-i}$. (2b)
- $p(i, N) = \alpha^i\beta^{N-i}$. (2c)

Next, recall Bellman's *principle of optimality* (see Bellman (1957)). This will now be used to prove initial conditions (1a), and then initial conditions (1b) and difference equation (1c):

For fixed n in $0 < n \le M$,

$$P(n) = \max_i \{ \sum_{j=1}^{n} p(i,j)\, P(n-j) \}, \qquad \text{for all } i \text{ in } 0 < i < N;$$

$$= \max_i \begin{cases} \alpha^i (1-\beta^{N-i})\, P(n-i), & \text{for all } i \text{ in } 0 < i \le n; \\ (1-\alpha^i)\beta^{N-i}\, P(n-N+i), & \text{for all } i \text{ in } N-n \le i < N; \end{cases}$$

because of (2a, b). This establishes (1a). Moreover, the proofs of (1b, c) parallel that of (1a). Accordingly, this completes the *first* part of the proof of Theorem 1.

Case (ii): N even

With *N even*, however, there is, associated with $A(i)$, a *fourth* mutually exclusive way of obtaining a *non-zero* gain:

- When $i = N - i$. That is, when $i = M + 1$, *and all* components of Team *A* perform the task satisfactorily, *and*, simultaneously, *at least one* component of Team *B* does *not*—OR—when, conversely, $i = M + 1$, *and all* components of Team *B* perform the task satisfactorily, *and*, simultaneously, *at least one* component of Team *A* does *not*. Denote this event by $E_{A+B}(M+1)$. With $E_{A+B}(M+1)$, therefore, there is a gain of $M + 1$ units, and the probability of $E_{A+B}(M+1)$ occurring is $\alpha^{M+1}(1-\beta^{M+1}) + (1-\alpha^{M+1})\beta^{M+1}$.

With *N even*, therefore, event $E_{A+B}(M+1)$ arises *in addition to* the three previously mentioned events $E_A(i)$, $E_B(i)$ (both now with $i \neq M + 1$) and $E_{AB}(i)$. Accordingly, *here*:

For all i in $0 < i < N$ and all j in $0 < j \leq N$, findings (2a, b, c) *again* result (with $i \neq M + 1$ for both (2a, b)), *along with* the following conclusion:

- $p(M+1, M+1) = \alpha^{M+1}(1-\beta^{M+1}) + (1-\alpha^{M+1})\beta^{M+1}$.

The proofs of (1a, b, c) for *N even* parallel those for *N odd*. Accordingly, this completes the *second* (and *final*) part of the proof of Theorem 1. ❑

An example to illustrate Theorem 1 follows shortly, but first note that, henceforth, $A^*(n)$ will denote the (unique) *alternative* that produces $P(n)$.

Example 1

Let $N = 3$, $\alpha = 0.4$ and $\beta = 0.5$. Accordingly, $M = 1$ (and $P(0) = 1$); hence, (1a, b, c) are, respectively:

$$P(1) = \max \left\{ \begin{array}{l} \alpha(1-\beta^2)P(0) \\ \beta(1-\alpha^2)P(0) \end{array} \right. = \cdots = \max \left\{ \begin{array}{l} 0.3000 \\ \underline{\mathbf{0.4200}} \end{array} \right. = \underline{\mathbf{0.4200}};$$

$$P(2) = \max \left\{ \begin{array}{l} \alpha(1-\beta^2)P(1) + (1-\alpha)\beta^2 P(0) \\ \beta(1-\alpha^2)P(1) + (1-\beta)\alpha^2 P(0) \end{array} \right. = \cdots = \max \left\{ \begin{array}{l} \underline{\mathbf{0.2760}} \\ 0.2564 \end{array} \right. = \underline{\mathbf{0.2760}};$$

and, for fixed $n \geq 3$,

$$P(n) = \max \left\{ \begin{array}{l} \alpha(1-\beta^2)P(n-1) + (1-\alpha)\beta^2 P(n-2) + \alpha\beta^2 P(n-3) \\ \beta(1-\alpha^2)P(n-1) + (1-\beta)\alpha^2 P(n-2) + \beta\alpha^2 P(n-3) \end{array} \right.$$

The last result yields, in particular, the following findings:

$$P(3) = \max \left\{ \begin{array}{l} \alpha(1-\beta^2)P(2) + (1-\alpha)\beta^2 P(1) + \alpha\beta^2 P(0) \\ \beta(1-\alpha^2)P(2) + (1-\beta)\alpha^2 P(1) + \beta\alpha^2 P(0) \end{array} \right. = \cdots = \max \left\{ \begin{array}{l} \underline{\mathbf{0.2458}} \\ 0.2295 \end{array} \right. = \underline{\mathbf{0.2458}};$$

$$P(4) = \max \left\{ \begin{array}{l} \alpha(1-\beta^2)P(3) + (1-\alpha)\beta^2 P(2) + \alpha\beta^2 P(1) \\ \beta(1-\alpha^2)P(3) + (1-\beta)\alpha^2 P(2) + \beta\alpha^2 P(1) \end{array} \right. = \cdots = \max \left\{ \begin{array}{l} 0.1571 \\ \underline{\mathbf{0.1589}} \end{array} \right. = \underline{\mathbf{0.1589}}.$$

Accordingly, here, $A^*(1) = \boldsymbol{A(2)}$, $A^*(2) = \boldsymbol{A(1)}$, $A^*(3) = \boldsymbol{A(1)}$ and $A^*(4) = \boldsymbol{A(2)}$. (*In contrast*, it can be shown that, if $N = 3$, $\alpha = 0.1$ and $\beta = 0.3$ then $A^*(1) = \boldsymbol{A(2)}$, $A^*(2) = \boldsymbol{A(1)}$, $A^*(3) = \boldsymbol{A(1)}$ and $A^*(4) = \boldsymbol{A(1)}$ (*not* $\boldsymbol{A(2)}$). Furthermore, it can be established that, if $N = 4$, $\alpha = 0.4$ and $\beta = 0.5$ then—in contrast *again*—$A^*(1) = \boldsymbol{A(2)}$, $A^*(2) = \boldsymbol{A(2)}$ (*not* $\boldsymbol{A(1)}$), $A^*(3) = \boldsymbol{A(1)}$ and $A^*(4) = \boldsymbol{A(1)}$ (*not* $\boldsymbol{A(2)}$).)

In this section, it has been established (in Theorem 1) that the manager's *problems* can be modelled by a class of dynamic programming *difference systems* (1a, b, c). In the next section, it will be shown *how* these systems can now be analyzed to *implement* the model, and so achieve the manager's *objective*.

IMPLEMENTATION OF THE MODEL

Optimal Policy 1

To achieve the *objective*, the manager should, first, choose *alternative* $A^*(n)$. Thereupon, either (a) *both* teams foul, simultaneously (and so there are no more decisions to be made), or (b) $A^*(n)$ produces either $i^* = i$ or $N - i$ or N ($0 < i < N$) units of gain. In the latter situation, the manager should next choose $A^*(n - i^*)$, and then proceed as before. However, *whichever* be this next choice, the *maximum* probability—of thereby successfully accruing a *total* of n units of gain—is $P(n)$.

Example 2

Let $N = 3$, $\alpha = 0.4$, $\beta = 0.5$ and $n = 4$. Next, recall Example 1, which illustrates that the manager should, first, choose alternative $A^*(4)$ (= $\boldsymbol{A(2)}$). Thereupon, either (a) *both* teams foul, simultaneously, or (b) $A(2)$ produces either $i^* = 2$ or 1 or 3 units of gain. If $i^* = 2$ then the manager should next choose alternative $A^*(2)$ (= $\boldsymbol{A(1)}$, *instead*). If $i^* = 1$ then the manager should next choose alternative $A^*(3)$ (= $\boldsymbol{A(1)}$, *instead*). However, if $i^* = 3$ then the manager should next choose alternative $A^*(1)$ (= $\boldsymbol{A(2)}$, *again*). However, *whichever* be this next choice, the *maximum* probability—of thereby successfully accruing a *total* of 4 units of gain—is $P(4)$ (= $\underline{0.1589}$ = $\underline{\textbf{15.89 \%}}$).

CONCLUSIONS AND FUTURE RESEARCH DIRECTIONS

To achieve the *objective*, the manager should use Theorem 1 to determine the first *n optimal alternatives* $A^*(k)$, $0 < k \leq n$ (recall Example 1), and then proceed as described in Optimal Policy 1 (recall Example 2).

This requires the manager to, first, generate *each* of the $A^*(k)$, $0 < k \leq n$—*individually*—by systematically using (1a, b, c) to determine *each* of the associated *probabilities* $P(k)$, $0 < k \leq n$. It would be advantageous, however, *if*—for particular α, β and N—*difference equation* (1c) could be *solved*, subject to initial conditions (1a, b), thereby providing a *closed form* expression—a *formula*—for $P(n)$.

A formula for $P(n)$ would necessarily mean a *companion* formula for $A^*(n)$. This would, therefore, avoid the need to individually generate these $A^*(k)$, $0 < k \leq n$; and, as well, would provide valuable insights into their underlying *structural patterns* (and those of the $P(k)$, $0 < k \leq n$).

Furthermore, the discussed model easily *generalizes* to situations where there are *more than two* teams. Moreover, it is anticipated that equations (1a, b, c) may well model commercial (and scientific) problems *other* than these *manufacturing* ones, discussed above. The author will report, elsewhere, on these research directions, and on related investigations in the field of dynamic programming difference systems.

REFERENCES

Albright S.C., Winston W.L. and Zappe C. (1999). *Data Analysis and Decision Making with Microsoft Excel*. Brooks/Cole, Pacific Grove, CA.

Bellman R.E. (1957). *Dynamic Programming*. Princeton University Press, Princeton, N.J.

Birbas T., Daskalaki S. and Housos E. (1997). Timetabling for Greek High Schools. *J. Opl. Res. Soc.* **48**, pp. 1191-1200.

Denardo E.V. (1982). *Dynamic Programming: Models and Applications*. Prentice-Hall, Englewood Cliffs, N.J.

Esogbue A.D. (1989). *Dynamic Programming for Optimal Resource Systems Analysis*. Prentice-Hall, Englewood Cliffs, N.J.

Iwamoto S. (1985). On Bellman's Allocation Processes. *J. Math. Anal. Appl.* **111**, pp. 65-89.

Jakubowski A., Kulikowski R. and Wagner D. (1985). Interactive System for Negotiations on Allocation of Research Funds. *Annu. Rev. Autom. Program.* **12** (Part 1), pp. 202-207.

Phillips, S. Jr. (1996). Project Management Duration/Resource Tradeoff Analysis: An Application of the Cut Search Approach. *J. Opl. Res. Soc.* **47**, pp. 697-701.

Ratz J. and Russell D. (1987). An Extremal Problem Related to Probability. *Aequationes Math.* **34**, pp. 316-324.

Taylor B.W. III (1999). *Introduction to Management Science*. Prentice-Hall, Upper Saddle River, N.J., 6th ed.

Wallace R.J. (1984). Sequential Search for Zeroes of Derivatives. In *General Inequalities 4*: Proceedings of the Fourth International Conference, Oberwolfach, 1983 (ed. W. Walter), Birkhauser, Basel **71**, pp. 151-167.

Wallace R.J. (1987a). The Maximal Solution of a Restricted Subadditive Inequality in Numerical Analysis. *Aequationes Math.* **33**, pp. 183-193.

Wallace R.J. (1987b). An Even Order Search Problem. In *General Inequalities 5*: Proceedings of the Fifth International Conference, Oberwolfach, 1986 (ed. W. Walter), Birkhauser, Basel **80**, pp. 183-202.

Wallace R.J. (1999). The General Solution of a Periodic Difference System. *Wyz. Szkola Ped. Krakow. Roczn. Nauk.-Dydakt. Prace Matematyczne XVI* **207**, pp. 105-119.

Wallace R.J. (2000a). Optimal Sequential Search of Order $2(2^m - 1)$. In *Communications in Difference Equations*: Proceedings of the Fourth International Conference on Difference Equations and Applications, Poznan, 1998 (eds. S. Elaydi, G. Ladas, J. Popenda and J. Rakowski), Gordon and Breach, N.J., to appear.

Wallace R.J. (2000b). A Family of Restricted Subadditive Recursions, submitted.

Winston W.L. (1991). *Operations Research: Applications and Algorithms*. PWS-KENT, Boston, MA, 2nd ed.