# Dynamic Programming

A. Madhur[1]

Notes from the pioneer work of Richard Bellman on *Dynamic Programming*, 1954.

# 1 Multistage Allocation Process

Before proceeding, in what follows, a few relevant useful mathematical results are stated.

(A) Let $g, h$ be two continuous functions of $x$, such that $x \geq 0$ and $g(0) = 0 = h(0)$. If $m(x) = \underset{0 \leq r \leq x}{\text{Max}} \text{Max}\ (|g(r)|, |h(r)|)$ and $s = \text{Max}(a, b)$ (where $0 \leq a < 1$ and $0 \leq b < 1$), then,

$$\sum_{n=0}^{\infty} m(s^n x) < \infty \text{ for all } x \geq 0. \tag{A.1}$$

This is an important result, as will be seen later.

## 1.1 Two Stage Allocation Process

We start with a brief description of allocation process and the mechanism of it.

Consider a quantity $x$, such that it is desired, to partition it, into two parts, let them be $y$ and $x - y$. Further, assuming there exist two processes; let's denote them as $g$ and $h$ with their respecive outputs be $g(y)$ and $h(x - y)$. Then the cumulative return be $f_1(x) \equiv g(y) + h(x - y)$. The objective is to maximize $f_1(x)$. The implicit assumptions here are primarily that such a maximum exists and secondly, the only variable which can be controlled is the partition of the initial resource. Thus the problem can be summarized as maximizing $f_1(x)$ by finding the partition for the initial resource $x$. This is called a one-stage allocation process.

Let $\delta$ and $\epsilon$ be two non-negative real numbers such that $0 \leq \delta, \epsilon < 1$. After the first allocation, consider the resource[2] $y + (x - y)$ gets depreciated and the what remains of it is $\delta y + \epsilon(x - y)$. Designating this as the new initial resource $x_1$, we have,

$$x_1 = \delta y + \epsilon(x - y) = y_1 + (x_1 - y_1) \ \ (\equiv y_k(x_1) \text{ say}). \tag{1.1}$$

However please notice here that the equation(1.1) does **NOT** mean that $\delta y = y_1$ and $\epsilon(x - y) = (x_1 - y_1)$. Proceeding as before, partitioning the resource $x_1$ into $y_1$ and $y_1 - x_1$ and generating the output, we have,

$$f_2(x) \equiv g(y_1) + h(x_1 - y_1).$$

Note that the constitution of $R_2$ is remarkably different from $R_1$, since the input parameters being the same for both $R_1$ and $R_2$. The cumulative return after this second-allocation process is,

$$f_2(x) \equiv g(y) + h(x - y) + g(y_1) + h(x_1 - y_1). \tag{1.2}$$

This is what is called two-stage allocation process. Formally

$$f_1(x) = \underset{0 \geq y \geq x}{\text{Max}} \left\{ g(y) + h(x - y) \right\}.$$

The output after the second allocation,

$$f_2(x) \equiv \underset{0 \leq y < x}{\text{Max}} \left\{ g(y) + h(x - y) + f_1(\delta x + \epsilon(x - y)) \right\},$$

or,

$$f_2(x) \equiv \underset{0 \leq y < x}{\text{Max}} \left\{ f_1(x) + f_1(\delta x + \epsilon(x - y)) \right\}. \tag{1.3}$$

[1] a.madhur@yahoo.com

[2] It is important here to note that the nature of this type of resource is different form the generally held view of single use, exhaustible resource.

Note that the first $f_1(x)$ $(= g(y) + h(x-y))$ is a constant term while the second term is what varies and carries all the causal information from previous executions. Hence, this form (which is by deliberate design, as would be shown later) is amenable for extrapolating the behaviour for $n^{\text{th}}$ stage, i.e.,

$$f_n(x) = \underset{0 \le y < x}{\text{Max}} \{f_1(x) + f_{n-1}(\delta x + \epsilon(x-y))\} \tag{1.4}$$

for $n \geq 2$.

Here, according to Bellman, the recurrence relation given by equation(1.4), is sufficient enough to provide both the optimal division of resource and their optimal utilization, with $y_n(x)$ is given as,

$$\begin{aligned}
&\bar{y}_1 = y_{n-1}(a\bar{y} + b(x - \bar{y})) \\
&\bar{y}_2 = y_{n-2}(a\bar{y_1} + b(x_1 - \bar{y}_1)) \\
&. \\
&. \\
&. \\
&\bar{y}_{n-1} = y_1(a\bar{y}_{n-2} + b(x_{n-2} - \bar{y}_{n-2})).
\end{aligned}$$

There are a few subtle nuances, which have been implicitly considered to be true, in the description above.

1. **Resources**: The resource which concerns with this specific avenue in the study of optimality, does not have a extinguishing feature after use. Often, these are given as an abstract continuation of real world resources, such as a dice throw after a few predefined specific throws. Or a robot moving in an $xy$-plane with obstacles towards a predefined target, where the movement choices for the robot gets constrained after making a move.
2. **Partition of Resources**: This is the one of the most basic tenets of dynamic programming. In general practice of computer programming, one usually adheres to all possible choices and then tries to limit them, according to the constraints given in the problem or more generally, by making a specific obeservation in the recurrence relation of optimal output; since the both the output and the partition are not mutually exclusive events.
3. **Nature of Problem**: The reason why even the straight forward definition of such a problem is not possible in a universal sense, is that, the conditions and constraints of the definitions, are dependent on the problem itself. Thus and thereby, it will be seen that the nature of the problem has often been inexorably entwined with the constraints of the problem.

---

However, notwithstanding with the construction of the formulation presented above; approaching the same problem construct in a slightly different way. As let $x$ be the initial resource, and its first partition be $r$ and $x - r$. Then the cumulative output is

$$\mathcal{O}_1(x) = g(r) + h(x - r) \tag{B.1}$$

after the first stage, please note that the *maximum* of the RHS is taken. Proceeding with the depreceated assets,

$$\mathcal{O}_2(x) = O_1(x) + \mathcal{O}_1(\delta r + \epsilon(x - r))$$

and consequently after the third stage,

$$\mathcal{O}_3(x) = O_1(x) + \mathcal{O}_1(\delta^2 r + \epsilon^2(x - r))$$

and likewise after $n^{\text{th}}-$stage,

$$\mathcal{O}_n(x) = O_1(x) + \mathcal{O}_1(\delta^{n-1} r + \epsilon^{n-1}(x - r)).$$

Generalizing this,

$$\boxed{\mathrm{O}_n(x) = \mathcal{O}_\phi(\delta^{n-1} r + \epsilon^{n-1}(x - r)) + \text{some initial constant}} \tag{B.2}$$

where $\mathcal{O}_\phi(x) = g(r) + h(x - r)$ as $x = r + (x - r)$ being the optimal partition.

---

In what follows below, several ready-made constructions are described. Each belongs to a certain class of optimization problems. However since most of the systems defined in natural sciences are a combination of some basic systems (or so we would like to believe), therefore, possibilities of variance, by combining a few known basic structures, in degrees of incorporations, are too many.

## 1.2 Multi-Dimensional Maximization Problem

Consider the problem of, determining the maximum for the following function,

$$F(x_1, x_2, \ldots x_N) = \sum_{i=1}^{N} g_i(x_i) \tag{1.5}$$

with the following constraints

(a) $x_1 + x_2 + \cdots + x_N = s$ where $s \geq 0$ and $N \in \mathbb{N}$; and

(b) $x_i \geq 0$.

The following recurrence relation is outcome for such a formulation, provided $f_N(s) = \underset{\{x_i\}}{\text{Max}}\, F(x_1, x_2, \ldots, x_N)$;

$$f_N(s) = \underset{0 \leq x \leq s}{\text{Max}} \{g_N(x) + f_{N-1}(c - x)\} \tag{1.6}$$

for all $N \in \mathbb{N}$ and $N \geq 2$.

Please notice the difference between the formulations, definitions of this problem with the "two-stage allocation problem". The division of resource is under constraint here, i.e. $\sum_{i=1}^{N} x_i = s$, here the equality, $=$ has a very dedicated meaning, when compared to the sign of $\leq$. Instead of depreciation by certain factor, here, the concerning portion of the resource is simply getting irrevocably exhausted.

Since the process is causal, as dictated by the problem, thus at the time $f_1(s) = g_1(s)$, $f_0(s)$ becomes meaningless.

## 1.3 Multi-Dimensional Minimization Problem

This concerns with the minimizing the function given below,

$$F(x_1, x_2, \ldots, x_N) = \sum_{k=1}^{N} g_k(x_k - r_k) + \sum_{k=1}^{N} h_k(x_k - x_{k-1}) \tag{1.7}$$

with the sequence $\{x_k\}$ is to be determined for the given sequence $\{r_k\}$.

For $R \in \{1, 2, 3, \ldots, N\}$ and $x_0 = s$, let the $f_R(s)$ be the minimum over all $x_R, x_{R+1}, \ldots, x_N$ of the function

$$F_R = \sum_{k=R}^{N} g_k(x_k - r_k) + \sum_{k=R}^{N} h_k(x_k - x_{k-1}) \tag{1.8}$$

where $x_{R-1} = s$. Thus

$$f_N(s) = \underset{x}{\text{Min}} \{g_N(x - r_N) + h_N(x - s)\} \text{ and} \tag{1.9}$$

$$f_R(s) = \underset{x}{\text{Min}} \{g_R(x - r_R) + h_R(x - s) + f_{R+1}(s)\} \tag{1.10}$$

for $R = 1, 2, \ldots, N - 1$.

---

---

---

However there is subtle prestidigitation in the aforementioned logic. First, we had some qunatity $x$ which we wish to allocate to different resources with their respective costs. We constrainted the number of processes to two. And then we supposed that there exist such a partition, let's say it is $y$. That is, we have already figured out the solution of the problem.

However taking another approach; given,

$$R_1(x, y) = g(y) + h(x - y),$$
$$\text{this is a recurrence relation, viz., } \Rightarrow R_2(x, y, y_1) = R_1(x, y) + \text{ remaining quantity}$$

### 1.4 $N$-Stage Allocation Process

Along the same line, an $N$-stage allocation process could be defined;

$$g(y) + h(y-x) + \left\{ \sum_{i=1}^{N-1} g(y_i) + \sum_{i=1}^{N-1} h(y_i - x_i) \right\} \equiv R_N(x, y, y_1, \ldots, y_{N-1}). \tag{1.11}$$

However, there is a subtle point regarding this depiction, which might not be readily visible to the unfamiliar eye. In all of this construction, the values of reduction coefficients, i.e., $a$ and $b$ remain unchanged. That is, they are independent of the stage processing (on a personal note, I found this assumption quite amusing, though it falls perfectly well within the bandwidth of practicality, −as stated by Bellman in the preface (and as we see later)− as to provide a comprehensive solution system). This gives the following,

$$\begin{aligned} x_1 &= ay + b(x-y) \\ x_2 &= ay_1 + b(x_1 - y_1) \\ &\cdot \\ &\cdot \\ &\cdot \\ x_{N-1} &= ay_{N-2} + b(x_{N-2} - y_{N-2}), \end{aligned} \tag{1.12}$$

where $0 \leq y_i \leq x_i$ for $i \in [0, N-1]$.

## 2 Solution Procedure

### 2.1 Functional Approach

Considering a functional formulation of equation(1.11),

$$f_\eta(x) = \underset{\{x_i, y_i\}}{\text{Max}} R_\eta(x, y, \ldots, y_{\eta-1}) \tag{2.1}$$

for $\eta = 2, 3, \ldots$, along with

$$f_1(x) = \underset{0 \leq y \leq x}{\text{Max}} \{g(y) + h(x-y)\}. \tag{2.2}$$

n The next step is to get an equation for $f_2(x)$ in terms of $f_1(x)$. Consider, $f_1 \equiv f_1(x)$ and $f_2 \equiv f_2(x)$, then,

$$\begin{aligned} f_1 &= g(y) + h(x-y) \text{ and} \\ f_2 &= f_1 + g(y_1) + h(x_1 - y_1), \end{aligned}$$

where $x_1 = ay + b(x-y) \Rightarrow x_1 = y_1 + (x_1 - y_1)$. The equation

$$f_2 = g(y) + h(x-y) + f_1(x_1) \tag{2.3}$$

which Bellman gives with the with the following reasoning,

> return from second stage =
> Maximum {return from first stage + $f_1$(remaining)}.

This observation gets generalized as,

$$f_\eta(x) = \underset{0 \leq y \leq x}{\text{Max}} \{g(y) + h(x-y) + f_{\eta-1}(ay + b(x-y))\}. \tag{2.4}$$

And then, from Bellman, stage-wise evalution yields not only $f_k(x)$ but also $y_k$, the optimum allocation. This, in my opinion, is bizzare, as one encounters later, for the time being, proceeding according to the Bellman's

evaluation of aforementioned quantities, we have,

$$\begin{aligned}
&\bar{y} = y_N(x)\\
&\bar{y}_1 = y_{N-1}(a\bar{y} + b(x - \bar{y}))\\
&\bar{y}_2 = y_{N-1}(a\bar{y}_1 + b(x_1 - \bar{y}_1))\\
&.\\
&.\\
&.\\
&\bar{y}_{N-1} = y_1(a\bar{y}_{N-2} + b(x_{N-2} - \bar{y}_{N-2}))
\end{aligned}$$

---

---

**Resource Allocation Problem**

The owner of a chain of four grocery stores has purchased six boxes of strawberries. The table below gives the profit for each store against the allocation of (the number of) strawberries boxes.

| Number of Boxes | Store 1 | Store 2 | Store 3 | Stores 4 |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 1 | 4 | 2 | 6 | 2 |
| 2 | 6 | 4 | 8 | 3 |
| 3 | 7 | 6 | 8 | 4 |
| 4 | 7 | 8 | 8 | 4 |
| 5 | 7 | 9 | 8 | 4 |
| 6 | 7 | 10 | 8 | 4 |

Now if $\zeta_\circ \equiv \zeta(x_i, y_i)\rfloor_{i=0} = \{g(y_i) - h(x_i - y_i)\}\rfloor_{i=0}$, then,

$$f_2 = \sigma \cdot f_1 + \delta \cdot \zeta_\circ \tag{2.5}$$

**Theorem 2.1.** *Some Theorem*

*Proof.* proof of some theorem □

# 3 Acknowledgement

All material herein, which have been used as a source, is from Richard Bellman and his research. My interest is from the view-point of pure math; and some semblance of in-vogue combinatorial optimality.