# 1 Dynamic Programming: The Optimality Equation

We introduce the idea of dynamic programming and the principle of optimality. We give notation for state-structured models, and introduce ideas of feedback, open-loop, and closed-loop controls, a Markov decision process, and the idea that it can be useful to model things in terms of time to go.

## 1.1 Control as optimization over time

Optimization is a key tool in modelling. Sometimes it is important to solve a problem optimally. Other times either a near-optimal solution is good enough, or the real problem does not have a single criterion by which a solution can be judged. However, even then optimization is useful as a way to test thinking. If the 'optimal' solution is ridiculous it may suggest ways in which both modelling and thinking can be refined.

Control theory is concerned with dynamic systems and their **optimization over time**. It accounts for the fact that a dynamic system may evolve stochastically and that key variables may be unknown or imperfectly observed (as we see, for instance, in the UK economy).

This contrasts with optimization models in the IB course (such as those for LP and network flow models); these static and nothing was random or hidden. It is these three new features: dynamic and stochastic evolution, and imperfect state observation, that give rise to new types of optimization problem and which require new ways of thinking.

We could spend an entire lecture discussing the importance of control theory and tracing its development through the windmill, steam governor, and so on. Such 'classic control theory' is largely concerned with the question of stability, and there is much of this theory which we ignore, e.g., Nyquist criterion and dynamic lags.

## 1.2 The principle of optimality

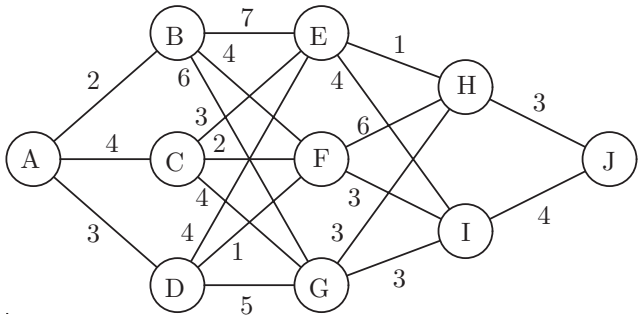
A key idea is that optimization over time can often be regarded as 'optimization in stages'. We trade off our desire to obtain the lowest possible cost at the present stage against the implication this would have for costs at future stages. The best action minimizes the sum of the cost incurred at the current stage and the least total cost that can be incurred from all subsequent stages, consequent on this decision. This is known as the Principle of Optimality.

**Definition 1.1 (Principle of Optimality)** *From any point on an optimal trajectory, the remaining trajectory is optimal for the corresponding problem initiated at that point.*

## 1.3 Example: the shortest path problem

Consider the 'stagecoach problem' in which a traveler wishes to minimize the length of a journey from town A to town J by first traveling to one of B, C or D and then onwards to one of E, F or G then onwards to one of H or I and the finally to J. Thus there are 4 'stages'. The arcs are marked with distances between towns.






Road system for stagecoach problem

**Solution.** Let $F(\mathrm{X})$ be the minimal distance required to reach J from X. Then clearly, $F(\mathrm{J}) = 0$, $F(\mathrm{H}) = 3$ and $F(\mathrm{I}) = 4$.

$$F(\mathrm{F}) = \min[\,6 + F(\mathrm{H}), 3 + F(\mathrm{I})\,] = 7\,,$$

and so on. Recursively, we obtain $F(\mathrm{A}) = 11$ and simultaneously an optimal route, i.e., A→D→F→I→J (although it is not unique). ■

The study of dynamic programming dates from Richard Bellman, who wrote the first book on the subject (1957) and gave it its name. A very large number of problems can be treated this way.

## 1.4 The optimality equation

**The optimality equation in the general case.** In **discrete-time** $t$ takes integer values, say $t = 0, 1, \ldots$. Suppose $u_t$ is a **control variable** whose value is to be chosen at time $t$. Let $U_{t-1} = (u_0, \ldots, u_{t-1})$ denote the partial sequence of controls (or decisions) taken over the first $t$ stages. Suppose the cost up to the **time horizon** $h$ is given by

$$\mathbf{C} = G(U_{h-1}) = G(u_0, u_1, \ldots, u_{h-1})\,.$$

Then the **principle of optimality** is expressed in the following theorem.

**Theorem 1.2 (The principle of optimality)** *Define the functions*

$$G(U_{t-1}, t) = \inf_{u_t, u_{t+1}, \ldots, u_{h-1}} G(U_{h-1})\,.$$

*Then these obey the recursion*

$$G(U_{t-1}, t) = \inf_{u_t} G(U_t, t+1) \quad t < h\,,$$

*with terminal evaluation* $G(U_{h-1}, h) = G(U_{h-1})$.

The proof is immediate from the definition of $G(U_{t-1}, t)$, i.e.,

$$G(U_{t-1}, t) = \inf_{u_t} \inf_{u_{t+1}, \ldots, u_{h-1}} G(u_0, \ldots, u_{t-1},\ u_t\ , u_{t+1}, \ldots, u_{h-1})\,.$$

**The state structured case.** The control variable $u_t$ is chosen on the basis of knowing $U_{t-1} = (u_0, \dots, u_{t-1})$, (which determines everything else). But a more economical representation of the past history is often sufficient. For example, we may not need to know the entire path that has been followed up to time $t$, but only the place to which it has taken us. The idea of a **state variable** $x \in \mathbb{R}^d$ is that its value at $t$, denoted $x_t$, is calculable from known quantities and obeys a **plant equation** (or law of motion)

$$x_{t+1} = a(x_t, u_t, t)\,.$$

Suppose we wish to minimize a cost function of the form

$$\mathbf{C} = \sum_{t=0}^{h-1} c(x_t, u_t, t) + \mathbf{C}_h(x_h)\,, \tag{1.1}$$

by choice of controls $\{u_0, \dots, u_{h-1}\}$. Define the cost from time $t$ onwards as,

$$\mathbf{C}_t = \sum_{\tau=t}^{h-1} c(x_\tau, u_\tau, \tau) + \mathbf{C}_h(x_h)\,, \tag{1.2}$$

and the minimal cost from time $t$ onwards as an optimization over $\{u_t, \dots, u_{h-1}\}$ conditional on $x_t = x$,

$$F(x,t) = \inf_{u_t,\dots,u_{h-1}} \mathbf{C}_t\,.$$

Here $F(x,t)$ is the minimal future cost from time $t$ onward, given that the state is $x$ at time $t$. Then by an inductive proof, one can show as in Theorem 1.2 that

$$F(x,t) = \inf_u [c(x,u,t) + F(a(x,u,t), t+1)]\,, \quad t < h\,, \tag{1.3}$$

with terminal condition $F(x,h) = \mathbf{C}_h(x)$. Here $x$ is a generic value of $x_t$. The minimizing $u$ in (1.3) is the optimal control $u(x,t)$ and values of $x_0, \dots, x_{t-1}$ are irrelevant. The **optimality equation** (1.3) is also called the **dynamic programming equation** (DP) or **Bellman equation**.

The DP equation defines an optimal control problem in what is called **feedback** or **closed loop** form, with $u_t = u(x_t, t)$. This is in contrast to the **open loop** formulation in which $\{u_0, \dots, u_{h-1}\}$ are to be determined all at once at time 0. A **policy** (or strategy) is a rule for choosing the value of the control variable under all possible circumstances as a function of the perceived circumstances. To summarise:

(i) The optimal $u_t$ is a function only of $x_t$ and $t$, i.e, $u_t = u(x_t, t)$.

(ii) The DP equation expresses the optimal $u_t$ in closed loop form. It is optimal whatever the past control policy may have been.

(iii) The DP equation is a backward recursion in time (from which we get the optimum at $h-1$, then $h-2$ and so on.) The later policy is decided first.

'*Life must be lived forward and understood backwards.*' (Kierkegaard)



## 1.5 Markov decision processes

Consider now stochastic evolution. Let $X_t = (x_0, \dots, x_t)$ and $U_t = (u_0, \dots, u_t)$ denote the $x$ and $u$ histories at time $t$. As above, state structure is characterised by the fact that the evolution of the process is described by a state variable $x$, having value $x_t$ at time $t$, with the following properties.

(a) *Markov dynamics:* (i.e., the stochastic version of the plant equation.)

$$P(x_{t+1} \mid X_t, U_t) = P(x_{t+1} \mid x_t, u_t)\,.$$

(b) *Decomposable cost*, (i.e., cost given by (1.1)).

These assumptions define state structure. For the moment we also require.

(c) *Perfect state observation:* The current value of the state is observable. That is, $x_t$ is known at the time at which $u_t$ must be chosen. So, letting $W_t$ denote the observed history at time $t$, we assume $W_t = (X_t, U_{t-1})$. Note that $\mathbf{C}$ is determined by $W_h$, so we might write $\mathbf{C} = \mathbf{C}(W_h)$.

These assumptions define what is known as a discrete-time **Markov decision process** (MDP). Many of our examples will be of this type. As above, the cost from time $t$ onwards is given by (1.2). Denote the minimal expected cost from time $t$ onwards by

$$F(W_t) = \inf_\pi E_\pi[\mathbf{C}_t \mid W_t]\,,$$

where $\pi$ denotes a policy, i.e., a rule for choosing the controls $u_0, \dots, u_{h-1}$. We can assert the following theorem.

**Theorem 1.3** *$F(W_t)$ is a function of $x_t$ and $t$ alone, say $F(x_t, t)$. It obeys the optimality equation*

$$F(x_t, t) = \inf_{u_t} \{c(x_t, u_t, t) + E[F(x_{t+1}, t+1) \mid x_t, u_t]\}\,, \quad t < h\,, \tag{1.4}$$

*with terminal condition*

$$F(x_h, h) = \mathbf{C}_h(x_h)\,.$$

*Moreover, a minimizing value of $u_t$ in* (1.4) *(which is also only a function $x_t$ and $t$) is optimal.*

Proof. The value of $F(W_h)$ is $\mathbf{C}_h(x_h)$, so the asserted reduction of $F$ is valid at time $h$. Assume it is valid at time $t+1$. The DP equation is then

$$F(W_t) = \inf_{u_t} \{c(x_t, u_t, t) + E[F(x_{t+1}, t+1) \mid X_t, U_t]\}\,. \tag{1.5}$$

But, by assumption (a), the right-hand side of (1.5) reduces to the right-hand member of (1.4). All the assertions then follow. ■