PDE for Finance Notes, Spring 2000 – Section 1

Notes by Robert V. Kohn, Courant Institute of Mathematical Sciences. For use only in connection with the NYU course PDE for Finance, G63.2706, Spring 2000.

Special note: My normal Thursday 4-5 office hours are cancelled this week due to jury duty. However I expect to be in the office evenings after 5:30.

Deterministic optimal control. This is the first of two sections on deterministic optimal control. Main topics: the method of dynamic programming; the value function as solution of the Hamilton-Jacobi-Bellman equation; verification theorems. We also discuss some examples, including: optimizing the utility of consumption; the eikonal equation $|\nabla u| = 1$ as the HJB equation for a geometric minimum-time problem; the Hopf-Lax solution formula for the dynamic programming solution of $u_t + H(\nabla u) = 0$. Section 2 will address further aspects of determistic optimal control, including Pontryagin's maximum principle and a bit about viscosity solutions. It will also present further examples, including one involving transaction costs.

This is "standard" material, however I am not following any single text. Macki and Strauss (on reserve) is a good undergraduate text, but its emphasis is very different from mine. Chapter 10 of Evans (on reserve) and Section 1 of the article by Bardi in the book by Capuzzo-Dolcetta and Lions (on reserve) have a viewpoint much like mine, but few examples. A.K. Dixit, *Optimization in Economic Theory* (Oxford Univ Press, 1990), is a charming and inexpensive book, and has a nice informal treatment of some relevant topics in Chapters 10 and 11 (xerox copy in the Green Box reserve).

****

What is optimal control? A typical problem of optimal control is this: we have a system whose *state* at any time t is described by a vector $y = y(s) \in R^n$. The system evolves in time, and we have the ability to influence its evolution through a vector-valued *control* $\alpha(s) \in R^m$. The evolution of the system is determined by an *ordinary differential equation*

$$\dot{y}(s) = f(y(s), \alpha(s)), \quad y(0) = x, \tag{1}$$

and our goal is to choose the function $\alpha(s)$ for $0 < s < T$ so as to maximize some utility or minimize some cost, e.g.

$$\max \int_0^T h(y(s), \alpha(s))\, ds + g(y(T)). \tag{2}$$

The problem is determined by specifying the dynamics f, the initial state x, the final time T, the "running utility" h and the "final utility" g. The problem is solved by finding the optimal control $\alpha(s)$ for $0 < s < T$ and the value of the maximum.

The mathematical and engineering literature often focuses on minimizing some sort of cost; the economic literature on maximizing utility. The two problems are mathematically equivalent.

One needs some hypotheses on f to be sure the solution of the ODE defining $y(s)$ exists and is unique. We do not make these explicit since the goal of these notes is to summarize the main ideas without getting caught up in fine points. See Evans for a mathematically careful treatment. Another technical point: it's possible (even easy) to formulate optimal control problems that have no solution. If the utility is bounded above, then for any $\epsilon > 0$ there's certainly a control $\alpha_\epsilon(s)$ achieving a value within ϵ of optimal. But the controls α_ϵ might not converge to a meaningful control as $\epsilon \to 0$. Note however that even if an optimal control doesn't exist, the optimal *value* (the maximum utility) is still well-defined.

An optimal control problem is evidently a special type of optimization problem. What's special is that we're dealing with functions of time, and decisions that must be made as time proceeds. Often the optimal control is described by a *feedback law*. Such a law determines the optimal control $\alpha(s)$ as having the form $\alpha(s) = F(y(s), s)$ for some function F (the feedback law).

Example 1: Here is a simple example which already has financial interest. (It's a deterministic version of Merton's famous example of optimal investment and consumption; we'll do the version with investment in a few weeks). Consider an individual whose wealth today is x, and who will live exactly T years. His task is to plan the rate of *consumption* of wealth $\alpha(s)$ for $0 < s < T$. All wealth not yet consumed earns interest at a fixed rate r. The state equation is thus

$$\dot{y} = ry - \alpha, \quad y(0) = x. \tag{3}$$

The control is $\alpha(s) \geq 0$, and the state is constrained by $y(s) \geq 0$ (he cannot consume wealth he doesn't have). The goal is

$$\max \int_0^T e^{-\rho s} h(\alpha(s))\, ds$$

where ρ is the discount rate and $h(\alpha)$ is the utility of consumption. (The function h, which must be given as part of the formulation of the problem, should be monotonically increasing and concave. A typical choice is $h(\alpha) = \alpha^\gamma$ with $0 < \gamma < 1$.) We have, for simplicity, assigned no utility to final-time wealth (a bequest), so the solution will naturally have $y(T) = 0$. Our goal is not strictly of the form (2) due to the presence of discounting; well, we omitted discounting from (2) only for the sake of simplicity.

The state constraint $y(s) \geq 0$ is awkward to deal with. In practice it tells us that if the investor ever runs out of wealth (i.e. if $y(s)$ ever reaches 0) then $\alpha = 0$ and $y = 0$ thereafter. This state constraint can be avoided by reformulating the goal as

$$\max \int_0^\tau e^{-\rho s} h(\alpha(s))\, ds$$

where τ is the first time y reaches 0 if this occurs before T, or $\tau = T$ if y is positive for all $s < T$. With this goal we need not impose the state constraint $y(s) \geq 0$.

Control theory is related to – but much more general than – the one-dimensional calculus of variations. A typical calculus of variations problem is

$$\max_{y(s)} \int_0^T W(s, y(s), \dot{y})\, ds$$

subject, perhaps, to endpoint conditions on $y(0)$ and $y(T)$. The example just formulated can be expressed in this form,

$$\max_{y(s)} \int_0^T h(ry - \dot{y})\, ds, \quad \text{subject to } y(0) = x,$$

except that we have additional constraints $ry(s) - \dot{y}(s) \geq 0$ and $y(s) \geq 0$ for all s.

We will shortly discuss the method of dynamic programming as a scheme for solving optimal control problems. The key to this method is to consider how the solution depends on the *initial time* and *initial state* as parameters. Thus rather than start arbitrarily at time 0, it is better to introduce a variable initial time t. And it is fruitful to consider the *value function* $u(x,t)$, the optimal value achievable using initial time t and initial state x. In the context of our basic framework (1) this means changing the state equation to

$$\dot{y}(s) = f(y(s), \alpha(s)), \quad y(t) = x.$$

The control $\alpha(s)$ is now to be determined for $t < s < T$, and the value function is

$$u(x,t) = \max \int_t^T h(y(s), \alpha(s))\, ds + g(y(T)).$$

In the context of Example 1 it means changing the state equation to

$$\dot{y} = ry - \alpha, \quad y(t) = x,$$

and the objective to

$$u(x,t) = \max \int_t^T e^{-\rho s} h(\alpha(s))\, ds.$$

(Warning: with this definition $u(x,t)$ is the utility of consumption discounted to time 0. The utility of consumption discounted to time t is $e^{\rho t} u(x,t)$.)

We started by formulating the "typical" optimal control problem (1)-(2). Now let's discuss some of the many variations on this theme, to get a better sense of the scope of the subject. We repeat for clarity the state equation:

$$\dot{y}(s) = f(y(s), \alpha(s)) \text{ for } t < s < T \text{ with initial data } y(t) = x.$$

Sometimes we may wish to emphasize the dependence of $y(s)$ on the initial value x, the initial time t, and the choice of control $\alpha(s), t < s < T$; in this case we write $y = y_{x,t,\alpha}(s)$. The control is typically restricted to take values in some specified set A, independent of s:

$$\alpha(s) \in A \text{ for all } s;$$

the set A must be specified along with the dynamics f. Sometimes it is natural to impose *state constraints*, i.e. to require that the state $y(s)$ stay in some specified set Y:

$$y_{x,t,\alpha}(s) \in Y \text{ for all } s;$$

when present, this requirement restricts the set of admissible controls $\alpha(s)$. Our basic example (2) is known as a **finite horizon** problem; its value function is

$$u(x,t) = \max_{\alpha} \left\{ \int_t^T h(y_{x,t,\alpha}(s), \alpha(s))\, ds + g(y_{x,t,\alpha}(T)) \right\}. \tag{4}$$

For the analogous **infinite horizon** problem it is customary to set the starting time to be 0, so the value function depends only on the spatial variable x:

$$u(x) = \max_{\alpha} \int_0^\infty e^{-\rho s} h(y_{x,0,\alpha}(s), \alpha(s))\, ds. \tag{5}$$

Discounting is important for the infinite-horizon problem, since without it the integral defining u could easily be infinite. (As already noted in our example, it is also often natural to include discounting in a finite-horizon problem.)

The **minimum time** problem is a little bit different. It minimizes the time it takes $y(s)$ to travel from x to some target set $\mathcal{T}$. The value function is thus

$$u(x) = \min_{\alpha} \{\text{time at which } y_{x,0,\alpha}(s) \text{ first arrives in } \mathcal{T}\}. \tag{6}$$

The minimum time problem is somewhat singular: if, for some x, the solution starting at x cannot arrive in $\mathcal{T}$ (no matter what the control) then the value is undefined. The **discounted minimum time** problem avoids this problem: its value function is

$$u(x) = \min_{\alpha} \int_0^{\tau(x,\alpha)} e^{-s}\, ds \tag{7}$$

where $\tau(x,\alpha)$ is the time that $y_{x,0,\alpha}(s)$ first arrives in $\mathcal{T}$, or infinity if it never arrives. Notice that the integral can be evaluated: the quantity being minimized is $\int_0^{\tau(x,\alpha)} e^{-s}\, ds = 1 - e^{-\tau(x,\alpha)}$. So we're still minimizing the arrival time, but the value function is $1 - \exp(-\text{arrival time})$ instead of the arrival time itself.

Example 2. Here is a simple example of a minimum-time problem, with the great advantages that (a) we can easily visualize everything, and (b) we know the solution in advance. In its simplest form the problem is: given a point x in R^n, and a set $\mathcal{T}$ not containing x, find the distance from x to $\mathcal{T}$. We recognize this as a minimum time problem, by reformulating it in terms of paths travelled with speed ≤ 1. The state equation is

$$dy/ds = \alpha(s), \quad y(0) = x,$$

and the controls are restricted by

$$|\alpha(s)| \leq 1.$$

The minimum arrival time

$$u(x) = \min_{\alpha} \{\text{time of arrival at } \mathcal{T}\}$$

is of course the distance from x to $\mathcal{T}$, and the optimal strategy is to travel with constant velocity (and unit speed) toward the point in $\mathcal{T}$ that is closest to x. We remark that $u(x) = \text{dist}\,(x, \mathcal{T})$ solves the differential equation

$$|\nabla u| = 1$$

in its natural domain $\Omega = R^n - \mathcal{T}$, with boundary condition $u = 0$ at $\partial\Omega$. This is an example of a (time-independent) Hamilton-Jacobi equation. The solution is typically not smooth: consider for example the case when Ω is a circle or a square. The optimal control is determined by a feedback law ("wherever you are right now, proceed at unit speed toward the nearest point on the target $\mathcal{T}$"). The non-smoothness of u reflects the fact that the feedback law is discontinuous, with nonuniqueness where ∇u is discontinuous. There is clearly nothing pathological about this example: non-smooth value functions, and discontinuous feedback laws, are commonplace in deterministic optimal control.

****

Dynamic programming. There are basically two systematic approaches to solving optimal control problems: one known as the Pontryagin Maximum Principle, the other known as Dynamic Programming. The two approaches are fundamentally equivalent, though in specific problems one may be easier to apply than the other. We shall emphasize dynamic programming, because (a) it extends more easily to the random case (time-dependent decision-making to optimize expected utility), and (b) it extends the familiar financial procedure of valuing an option by working backward through a tree.

The essence of dynamic programming is pop psychology: "today is the first day of the rest of your life." More: every day is the first day of the future thereafter. How to use this insight? One way is to make it the basis of a numerical solution scheme. Another way is to use it to derive a PDE for $u(x,t)$. These two ideas are of course closely related: our numerical solution scheme is in fact a crude numerical scheme for solving the PDE.

Let's start with the numerical scheme, concentrating on the finite-horizon problem (4), and keeping space one-dimensional for simplicity. Our goal is to compute (approximately) the value function $u(x,t)$. Of course any numerical scheme must work in discrete space and time, so t is a multiple of Δt, and x is a multiple of Δx. It's also natural to consider that the controls are discretized: $\alpha(s)$ is piecewise constant with mesh Δt. Now work backward in time:

First Consider the problem with initial time $t = T$. In this case the dynamics is irrelevant. So are the control and the running utility . Whatever the value of x, the associated value function is $g(x)$. In other words: $u(x,T) = g(x)$.

Next Consider the problem with initial time $t = T - \Delta t$. Approximate the dynamics as

$$y(s + \Delta t) = y(s) + f(y(s), \alpha(s))\Delta t.$$

Since there is just one time interval between the initial time t and the final time $T = t + \Delta t$, and since the control is piecewise constant, the unknown is now just a single vector $\alpha = \alpha(t)$ (not a function). It is determined by optimization. We may approximate the objective integral by a sum (dropping terms of higher order in Δt), leading to

$$u(x, T - \Delta t) = \max_{\alpha} \left\{ h(x,\alpha)\Delta t + g\left(x + f(x,\alpha)\Delta t\right) \right\}.$$

This must be evaluated for each x (i.e. every multiple of Δx), and the maximization over α must be done globally (we need the global optimum, not just a local optimum). For a real numerical scheme some further structure is needed here: we should solve a problem in a bounded spatial domain, and impose concavity hypotheses assuring that there are no local optima. For the present conceptual discussion let us ignore such practical issues and proceed. (One might worry that when the spatial dimension is greater than 1 this scheme is utterly impractical, since the number of grid points x to be considered at each time t is of order $(\Delta x)^{-n}$ in dimension n. This worry is well-founded: our scheme is impractical in higher dimensions. However there are good numerical schemes for multidimensional problems. One option is to solve the Hamilton-Jacobi-Bellman equation we'll derive presently, using a suitable finite-difference scheme.) At the end of this step we have computed $u(\cdot, T - \Delta t)$ as a function of space.

Next Consider the problem with initial time $t = T - 2\Delta t$. For any initial state $x = y(t)$, the possible controls are now represented by a pair of vectors $\alpha(t), \alpha(t + \Delta t)$. However we can still solve the problem by considering just the current control $\alpha = \alpha(t)$, since the optimal choice of $\alpha(t + \Delta t)$ has already been determined in the course of evaluating $u(x, T - \Delta t)$. Making crucial use of the fact that the "running utility" is an integral in time, we may determine the optimal value $u(x, T - 2\Delta t)$ by solving

$$u(x, T - 2\Delta t) = \max_{\alpha} \left\{ h(x, \alpha)\Delta t + u\left(x + f(x, \alpha)\Delta t, T - \Delta t\right) \right\}.$$

Here the unknown is just the control α to be used during the time interval from $T - 2\Delta t$ to $T - \Delta t$. The optimal α depends of course on x, and the optimization in α must be done for each choice of x separately. (Again, this is the conceptual but impractical version; numerical optimal control uses various workarounds to make it more practical.) At the end of this step we have computed $u(\cdot, T - 2\Delta t)$ as a function of space.

Continue The scheme continues, working backward time-step by time-step. Notice that for computing $u(x, T - (j + 1)\Delta t)$ we need only save the values of $u(x, T - j\Delta t)$. However if we wish to synthesize an optimal control starting at an arbitary point x and time $t = T - (j + 1)\Delta t$ we must save much more information: namely the feedback law $\alpha = F(y, s)$, obtained in the course of calculating $u(y, s)$ for $s > t$. (This is the optimal initial-time-period value of the control, when the initial state is y and the initial time is s). This information permits us to synthesize the optimal control and solve the state equation at the same time: starting from x at time t, the state evolves by

$$y_\alpha(s + \Delta t) = y_\alpha(s)) + f(y_\alpha(s), \alpha(s))\Delta t$$

with $\alpha(s)$ determined by

$$\alpha(s) = F\left(y_\alpha(s), s\right).$$

We remark that a similar philosophy can be used in many other settings. One example is this standard scheme for computing the shortest path between two nodes of a graph. Pick one of the nodes (call it an endpoint). Find all nodes that lie distance 1 from it, then all points that lie distance 2 from it, etc. Stop when the other endpoint appears in the set you come up with.

Students of math finance will have noticed by now that dynamic programming looks a lot like the binomial-tree method for valuing a European or American option. The resemblance is no coincidence. The biggest difference is that for the European option no optimization need be done at any point in the calculation; for the American option the optimization is simple – over just two alternatives, to exercise or not to exercise. This is due to the completeness of the underlying market model. In a multiperiod market that's not complete, there *is* an optimization to be done at each stage. We'll discuss an example of this type when we get to stochastic optimal control. (Students not familiar with option pricing: don't worry, the concepts in this paragraph will be developed when we need them.)

The discrete-time, discrete-space scheme described above can be viewed as a crude numerical scheme for solving the PDE satisfied by the value function. This is known as the **Hamilton-Jacobi-Bellman equation**. We shall derive it, in essence, by taking the formal limit $\Delta t \to 0$ in our numerical discussion. This viewpoint can be used for all the optimal control problems we've discussed (finite-horizon, infinite-horizon, least-time, with or without discounting) but to fix ideas we concentrate on the usual finite-horizon example

$$u(x,t) = \max_{\alpha} \left\{ \int_t^T h(y(s), \alpha(s))\, ds + g(y(T)) \right\}$$

where the controls are restricted by $\alpha(s) \in A$, and the state equation is

$$dy/ds = f(y(s), \alpha(s)) \text{ for } t < s < T \text{ and } y(t) = x.$$

(Space can be multidimensional here.) The Hamilton-Jacobi-Bellman equation in this case is

$$u_t + H(\nabla u, x) = 0 \quad \text{for } t < T \tag{8}$$

with

$$u(x,T) = g(x) \quad \text{at } t = T,$$

where H (the "Hamiltonian") is defined by

$$H(p,x) = \max_{a \in A} \{ f(x,a) \cdot p + h(x,a) \}. \tag{9}$$

(Note that p is a vector with the same dimensionality as x; a is a vector with the same dimensionality as α.)

To explain, we start with the **dynamic programming principle**, which was in fact the basis of our discrete scheme. It says:

$$u(x,t) = \max_{\alpha} \left\{ \int_t^{t'} h(y_{x,t,\alpha}(s), \alpha(s))\, ds + u(y_{x,t,\alpha}(t'), t') \right\} \tag{10}$$

whenever $t < t' < T$. The justification is easy, especially if we assume that an optimal control exists (this case captures the main idea; see Evans for a more careful proof, without this hypothesis). Suppose the optimal utility starting at x at time t is achieved by an optimal control $\alpha_{x,t}(s)$. Then the restriction of this control to any subinterval $t' < s < T$ must be optimal for its starting time t' and starting position $y_{x,t,\alpha}(t')$. Indeed, if it weren't then there would be a new control $\alpha'(s)$ which agreed with α for $t < s < t'$ but did better for $t' < s < T$. Since the utility is additive – the running utility is $\int_t^T h(y,\alpha)\,ds = \int_t^{t'} h(y,\alpha)\,ds + \int_{t'}^T h(y,\alpha)\,ds$ – this new control would be better for the entire time period, contradicting the optimality of α. Therefore in defining $u(x,t)$ as the optimal utility, we can restrict our attention to controls that are optimal from time t' on. This leads immediately to (10).

Now let us derive (heuristically) the Hamilton-Jacobi-Bellman equation. The basic idea is to apply the dynamic programming principle with $t' = t+\Delta t$ and let $\Delta t \to 0$. Our argument is heuristic because (i) we assume u is differentiable, and (ii) we assume the optimal control is adequately approximated by one that is constant for $t < s < t+\Delta t$. (Our goal, as usual, is to capture the central idea, referring to Evans for a more rigorous treatment.) Since Δt is small, the integral on the right hand side of (10) can be approximated by $h(x,a)\Delta t$, where $a \in A$ is the (constant) value of α for $t < s < t+\Delta t$. Using a similar approximation for the dynamics, the dynamic programming principle gives

$$u(x,t) \geq h(x,a)\Delta t + u(x + f(x,a)\Delta t, t+\Delta t) + \text{errors we wish to ignore}$$

with equality when a is chosen optimally. Using the first-order Taylor expansion of u this becomes

$$u(x,t) \geq h(x,a)\Delta t + u(x,t) + (\nabla u \cdot f(x,a) + u_t)\Delta t + \text{error terms}$$

with equality when a is optimal. In the limit $\Delta t \to 0$ this gives

$$0 = u_t + \max_{a\in A}\{\nabla u \cdot f(x,a) + h(x,a)\},$$

i.e. $u_t + H(\nabla u, x) = 0$ with H as asserted above. The final-time condition is obvious: if $t = T$ then the dynamics is irrelevant, and the total utility is just $g(x)$.

That was easy. Other classes of optimal control problems are treated similarly. Let's look at the minimum-time problem, where the state evolves by

$$dy/ds = f(y,\alpha), \quad y(t) = x,$$

and the controls are restricted by

$$\alpha(s) \in A \quad \text{for all } s$$

for some set A. The associated Hamilton-Jacobi-Bellman equation is

$$H(\nabla u, x) = -1 \quad \text{for } x \notin \mathcal{T}$$

with Hamiltonian

$$H(p,x) = \min_{a\in A}\{f(x,a)\cdot p\} = 0.$$

The boundary condition is

$$u = 0 \quad \text{for } x \in \mathcal{T}.$$

To see this, we argue essentially as before: the value function (the time it takes to arrive at $\mathcal{T}$) should satisfy

$$u(x) \leq \Delta t + u(x + f(x, a)\Delta t) + \text{error terms}$$

for any $a \in A$, with equality when a is optimal. Using Taylor expansion this becomes

$$u(x) \leq \Delta t + u(x) + \nabla u \cdot f(x, a)\Delta t + \text{error terms.}$$

Optimizing over a and letting $\Delta t \to 0$ we get

$$1 + \min_{a \in A}\{f(x, a) \cdot \nabla u\} = 0,$$

which is the desired equation.

Let us specialize this to Example 2. In that example the set A is the unit ball, and $f(y, \alpha) = \alpha$, so $H(p, x) = \min_{|a| \leq 1} p \cdot a = -|p|$ and the Hamilton-Jacobi equation becomes $|\nabla u| = 1$, as expected.

****

Solutions of the Hamilton-Jacobi-Bellman equation are not unique (at least, not when we understand "solution" in the naive almost-everywhere sense). For example, there are *many* Lipschitz continuous solutions of $|\nabla u| = 1$ in a square, with $u = 0$ at the boundary. If one were smooth we might prefer it – however there is *no* smooth solution. So, is the HJB equation really of any use?

The answer is yes, it's very useful, for three rather distinct reasons. The first is obvious; the second is elementary but not obvious; the third is subtle, representing a major mathematical achievement of the past 20 years:

(a) In deriving the HJB equation, we deduced a relation between the optimal control and the value of ∇u: briefly, $\alpha(s)$ achieves the optimum in the definition of $H(p, x)$ with $p = \nabla u(y(s), s)$ and $x = y(s)$. Thus the derivation of the HJB equation tells us the relation between the value function and the optimal control. In many settings, this argument permits us to deduce a feedback law once we know the value function.

(b) The argument used for the HJB equation can often be reorganized to show that a conjectured formula for the value function is correct. This sort of argument is called a *verification theorem.*

(c) There is a more sophisticated notion of "solution" of a Hamilton-Jacobi equation, namely the notion of a *viscosity solution.* Viscosity solutions exist, are unique, and can be computed by suitable numerical schemes. Moreover the value function of a dynamic programming problem is automatically a viscosity solution of the associated HJB equation. (Chapter 10 of Evans' book gives an excellent introduction to the theoretical side of this topic. The book *Level Set Methods* by J. Sethian, Cambridge Univ Press, provides a readable introduction to the numerical side, concentrating on the special class of HJB equations associated with geometric evolution problems – closely connected with our minimum time example.)

Point (a) should be clear, and it will be illuminated further by various examples later on. Point (c) is an interesting story, but beyond the scope of the present discussion. Our present intention is to concentrate on point (b). We focus as usual on the setting of the finite-horizon problem. As usual, $u(x,t)$ denotes the value function (the maximal value achievable starting from state x at time t). Our plan is to develop schemes for proving upper and lower bounds on u. If we do a really good job the upper and lower bounds will coalesce – in which case they will fully determine u.

There's always one type of bound that is easy. Since we're maximizing utility, these are the lower bounds. Any scheme for choosing the control – for example a conjectured feedback law specifying $\alpha(s)$ as a function of $y(s)$ – provides a lower bound $v(x,t) =$ the value achieved by this scheme. The inequality

$$v(x,t) \le u(x,t)$$

is obvious, since u is the maximal value obtainable using *any* control – including the ones used to define v.

The **verification theorem** provides the other bound. In its most basic form – specialized to the present setting – it says the following. Suppose $w(x,t)$ is defined (and continuously differentiable) for $t < T$, and it solves the Hamilton-Jacobi equation (8) with $w = g$ at $t = T$. Then w is an upper bound for the value function:

$$u(x,t) \le w(x,t).$$

To see why, consider any candidate control $\alpha(s)$ and the associated state $y = y_{x,\alpha}(s)$ starting from x at time t. The chain rule gives

$$\begin{aligned}
\frac{d}{ds} w(y(s),s) &= w_s(y(s),s) + \nabla w(y(s),s) \cdot \dot{y}(s) \\
&= w_s(y(s),s) + \nabla w(y(s),s) \cdot f(y(s),\alpha(s)) \\
&\le w_s + H(\nabla w, y) - h(y(s),\alpha(s)) \qquad (11) \\
&= -h(y(s),\alpha(s)),
\end{aligned}$$

using for (11) the relation

$$H(p,y) = \max_{a \in A} \{f(y,a) \cdot p + h(y,a)\} \ge f(y,\alpha) \cdot p + h(y,\alpha)$$

with $y = y(s)$, $\alpha = \alpha(s)$, and $p = \nabla w(y(s),s)$. Now integrate in time from t to T:

$$w(y(T),T) - w(x,t) \le -\int_t^T h(y(s),\alpha(s))\, ds.$$

Since $w(y(T),T) = g(y(T))$ this gives

$$g(y(T)) + \int_t^T h(y(s),\alpha(s))\, ds \le w(x,t).$$

The preceding argument applies to any control $\alpha(s)$. Maximizing the left hand side over all admissible controls, we have

$$u(x,t) \leq w(x,t)$$

as asserted.

We presented the task of finding lower and upper bounds as though they were distinct, but of course they are actually closely correlated. A smooth solution w of the Hamilton-Jacobi equation comes equipped with its own feedback law (as discussed in point (a) above). It is natural to consider the lower bound v obtained using the feedback law associated with w. I claim that this v is equal to w. To see this, follow the line of reasoning we used for the verification theorem, noticing that (11) holds with equality if α is determined by the feedback associated with w. Therefore integration gives

$$w(x,t) = g(y(T)) + \int_t^T h(y(s), \alpha(s))\, ds$$

and the right hand side is, by definition, $v(x,t)$. In conclusion: if w is a (continuously differentiable) solution of the HJB equation, satisfying the appropriate final-time condition too, then w *is in fact the value function* $u(x,t)$.

It sounds like a great scheme, and in many ways it is. There is however a small fly in the ointment. Sometimes the value function isn't continuously differentiable. (Consider, for example, the minimum time problem). In such a case our proof of the verification theorem remains OK for paths that avoid the locus of nonsmoothness – or cross it transversely. But there is a problem if the state should happen to hug the locus of nonsmoothness. Said more plainly: if $w(x,t)$ has discontinuous derivatives along some set Γ in space-time, and if a control makes $(y(s), s)$ move along Γ, then the first step in our verification argument

$$\frac{d}{ds} w(y(s), s) = w_s(y(s), s) + \nabla w(y(s), s) \cdot \dot{y}(s)$$

doesn't really make sense (for example, the right hand side is not well-defined). Typically this problem is overcome by using the fact that the verification argument has some extra freedom: it doesn't really require that w solve the HJB equation exactly. Rather, it requires only that w satisfy the *inequality* $w_t + H(\nabla w, t) \leq 0$.

To give an example where this extra freedom is useful consider our geometrical Example 2, with target $\mathcal{T}$ the complement of the unit square in R^2. The HJB equation is $|\nabla u| = 1$ in Ω=unit square, with $u = 0$ at $\partial\Omega$. The value function is defined as $u(x)$=minimum time of arrival to $\partial\Omega$ (among all paths with speed ≤ 1). Simple geometry tells us the solution is the distance function $\operatorname{dist}(x, \partial\Omega)$, whose graph is a pyramid. We wish to give an entirely PDE proof of this fact.

One inequality is always easy. In this case it is the relation $u(x) \leq \operatorname{dist}(x, \partial\Omega)$. This is clear, because the right hand side is associated with a specific control law (namely: travel straight toward the nearest point of the boundary, with unit speed). To get the other inequality, observe that if $w \leq 0$ at $\partial\Omega$ and $|\nabla w| \leq 1$ in Ω then

$$\frac{d}{ds} w(y(s)) \quad = \quad \nabla w(y(s)) \cdot \dot{y}(s)$$

$$
\begin{aligned}
&= \nabla w(y(s)) \cdot \alpha(s) \\
&\geq -|\nabla w(y(s))| \geq -1.
\end{aligned}
$$

(Here $y(s)$ solves the state equation $\dot{y} = \alpha$, with initial condition $y(0) = x$ and any admissible control $|\alpha(s)| \leq 1$.) If τ is the time of arrival at $\partial\Omega$ then integration gives

$$w(y(\tau)) - w(x) \geq \int_0^\tau (-1)\, ds.$$

Since $w(y(\tau)) \leq 0$ we conclude that

$$w(x) \leq \tau.$$

Minimizing the right hand side over all admissible controls gives

$$w(x) \leq u(x).$$

We're essentially done. We cannot set w equal to the distance function itself, because this choice isn't smooth enough. However we can choose w to be a slightly smoothed-out version of the distance function minus a small constant. It's easy to see that we can approach the distance function from below by such functions w. Therefore (using these w's and passing to a limit)

$$\text{dist}\,(x, \partial\Omega) \leq u(x),$$

completing our PDE argument that the value function is in this case the distance function.

It's time for a more financial example. Let's give the **solution to Example 1** for a power-law utility. The state equation is

$$\dot{y} = ry - \alpha, \quad y(t) = x$$

where x is the initial wealth and α is the consumption rate, restricted by $\alpha \geq 0$ (an explicit constraint on the controls). We consider the problem of finding

$$u(x,t) = \max_\alpha \int_t^T e^{-\rho s} \alpha^\gamma(s)\, ds,$$

which amounts to the utility of consumption with the power-law utility function $h(\alpha) = \alpha^\gamma$. Utility functions should be concave so we assume $0 < \gamma < 1$.

First, before doing any real work, let us show that the value function has the form

$$u(x,t) = g(t)x^\gamma$$

for some function $g(t)$. It suffices for this purpose to show that the value function has the homogeneity property

$$u(\lambda x, t) = \lambda^\gamma u(x,t), \tag{12}$$

for then we can take $g(t) = u(1,t)$. To see (12), suppose $\alpha(s)$ is optimal for starting point x, and let $y_x(s)$ be the resulting trajectory. We may consider the control $\lambda\alpha(s)$ for the trajectory that starts at λx, and it is easy to see that the associated trajectory is

$y_{\lambda x}(s) = \lambda y_x(s)$. Using the power-law form of the utility this comparison demonstrates that

$$u(\lambda x, t) \geq \lambda^\gamma u(x,t).$$

This relation with λ replaced by $1/\lambda$ and x replaced by λx gives

$$u(x,t) \geq \lambda^{-\gamma} u(\lambda x, t),$$

completing the proof of (12).

Now let's find the HJB equation. This is almost a matter of specializing the general calculation to the case at hand. But we didn't have a discount term before, so let's redo the argument to avoid any doubt. From the dynamic programming principle we have

$$u(x,t) \geq e^{-\rho t} a^\gamma \Delta t + u(x + (rx - a)\Delta t, t + \Delta t) + \text{error terms}$$

with equality when $a \geq 0$ is chosen optimally. Using the first-order Taylor expansion of u this becomes

$$u(x,t) \geq e^{-\rho t} a^\gamma \Delta t + u(x,t) + (u_x(rx - a) + u_t)\Delta t + \text{error terms}$$

with equality when a is optimal. In the limit $\Delta t \to 0$ this gives

$$u_t + \max_{a \geq 0} \{u_x(rx - a) + e^{-\rho t} a^\gamma\} = 0.$$

This is the desired HJB equation, to be solved for $t < T$. The final-time condition is of course $u = 0$ (since no utility is associated to final-time wealth).

It's obvious that $u_x > 0$. (This follows from the observation that $u(x,t) = g(t)x^\gamma$. Or it's easy to prove using the original problem formulation and a suitable comparison argument.) Therefore the optimal a is easy to find, by differentiation, and it is positive:

$$\gamma a^{\gamma - 1} = e^{\rho t} u_x.$$

This is the feedback law, determining the optimal control (once we know u_x). Remembering that $u(x,t) = g(t)x^\gamma$, we can write the feedback law as

$$a = \left[e^{\rho t} g(t)\right]^{1/(\gamma-1)} x$$

To find g (and therefore u) we substitute $u = g(t)x^\gamma$ into the HJB equation. This leads, after some arithmetic and cancelling a common factor of x^γ from all terms, to

$$\frac{dg}{dt} + r\gamma g + (1-\gamma) g (e^{\rho t} g)^{1/(\gamma-1)}.$$

This equation (with the end condition $g(T) = 0$) is entirely equivalent to the original HJB equation. It looks ugly, however it is not difficult to solve. First, multiply each term by $e^{\rho t}$ to see that $G(t) = e^{\rho t} g(t)$ solves

$$G_t + (r\gamma - \rho)G + (1-\gamma)G^{\gamma/(\gamma-1)} = 0.$$

Next, multiply by $(1-\gamma)^{-1}G^{\gamma/(1-\gamma)}$ to see that $H(t) = G^{1/(1-\gamma)}$ satisfies the linear equation

$$H_t - \mu H + 1 = 0 \quad \text{with } \mu = \tfrac{\rho - r\gamma}{1-\gamma}.$$

This is a linear equation! The solution satisfying $H(T) = 0$ is

$$H(t) = \lambda^{-1}\left(1 - e^{-\lambda(T-t)}\right).$$

Unraveling our changes of variables gives finally

$$g(t) = e^{-\rho t}\left[\frac{1-\gamma}{\rho - r\gamma}\left(1 - e^{-\frac{(\rho - r\gamma)(T-t)}{1-\gamma}}\right)\right]^{1-\gamma}.$$

We've solved the HJB equation. Have we actually found the value function and the optimal feedback (consumption) policy? Yes indeed. The verification theorem given above supplies the proof. (Well, it should be redone with discounting, and with the more precise formulation of the objective which integrates the utility only up to the first time τ when $y = 0$, if this occurs before T. These modifications require no really new ideas.) Nothing fancy is needed since $u(x,t)$ is smooth.

****

Now another class of examples. Consider what becomes of the finite-horizon problem when f (the right hand side of the state equation) and h (the running utility) are independent of x. The state equation is thus

$$dy/ds = f(\alpha(s)) \quad \text{with } y(t) = x$$

and the value function is

$$u(x,t) = \max_{\alpha(s)\in A}\left[\int_t^T h(\alpha(s))\,ds + g(y(T))\right].$$

The HJB equation has the form $u_t + H(\nabla u) = 0$ for $t < T$, with final-time condition $u = g$ at $t = T$ and with Hamiltonian

$$H(p) = \max_{a\in A}\{f(a)\cdot p + h(a)\}.$$

Notice that H is always a *convex* function of p, since the definition expresses it as a maximum of linear functions of p.

The function H does not determine f and h – different f's and h's can lead to the same HJB equation. But for any convex H there's an especially simple choice of an associated f and h, namely

$$f(a) = a, \quad h(a) = \min_p\{H(p) - a\cdot p\}. \tag{13}$$

(Comments on the latter formula: (a) h is concave, as it should be, since it is a minimum of linear functions; (b) $-h = \max_p\{a\cdot p - H(p)\}$ is the "Fenchel transform" of H). To justify this assertion we must show that when H is convex and f, h are given by (13), the resulting Hamiltonian

$$\max_{a \in A} \{f(a) \cdot p + h(a)\}$$

is equal to $H(p)$. The definition of h shows that

$$h(a) \leq H(p) - a \cdot p$$

for all p, with equality when p is chosen optimally (depending on a). We can rewrite this as

$$H(p) \geq a \cdot p + h(a)$$

for all a and p, with equality when a is chosen optimally (depending on p). Maximization over a gives the desired relation.

The preceding calculation is best understood in a more general context, as a fact about Fenchel transforms. For any function $F(z)$ defined for $z \in R^n$, its Fenchel transform is defined as

$$F^*(w) = \max_z \{w \cdot z - F(z)\}.$$

(We can effectively restrict the maximization to $z \in A$ by choosing $F(z) = \infty$ when $z \notin A$.) Here are two basic facts about the Fenchel transform:

- The double Fenchel transform $(F^*)^*(z)$ is the *convexification* of F, i.e. its graph is the convex hull of the graph of F.

- If F is convex then its double Fenchel transform $(F^*)^*$ is equal to F.

The proof of these facts follows the argument sketched briefly above. The fact that when H is convex the choice (13) leads to a HJB with Hamiltonian H is just an application of the second bullet.

Summarizing the above: if H is convex, we can define a "dynamic programming" solution of the HJB equation $u_t + H(\nabla u) = 0$ (with $u = g$ at $t = T$) as the solution of the finite-horizon dynamic programming problem associated with (13). This optimal control problem is easy to solve more or less explicitly. The key observation is that when $f(a) = a$ and $h(a)$ is concave, the optimal control is constant and the associated trajectory is a constant-velocity path. In fact, $f(a) = a$ means the control is the *velocity* of the path $y(s)$. The concavity of h gives

$$h[\text{average velocity}] \geq \text{average of } h[\text{velocity}].$$

Notice moreover that the average velocity of a path depends only on its endpoints, since

$$\frac{1}{T-t} \int_t^T \frac{dy}{ds} \, ds = \frac{1}{T-t} (y(T) - y(t)).$$

Thus replacing any path by one with the same endpoints and constant velocity can only improve the utility. We thus arrive at the **Hopf-Lax solution formula**: when $f(a) = a$

and $h(a)$ is concave, the solution of the associated dynamic programming problem (with final-time utility g) is

$$u(x,t) = \max_z \left\{ (T-t)h\left(\frac{z-x}{T-t}\right) + g(z) \right\}. \tag{14}$$

Here z represents the state at time T – the only remaining unknown – and $(z-x)/(T-t)$ is the velocity of the associated path starting at x at time t and ending at z at time T. We view (14) as specifying the (dynamic programming) solution of the associated HJB equation $u_t + H(\nabla u) = 0$ with final-time condition $u = g$ at $t = T$.

Let's bring this down to earth by considering the specific example: $f(a) = a$, $H(p) = \frac{1}{2}|p|^2$, and $h(p) = -\frac{1}{2}|a|^2$. Then the Hamilton-Jacobi-Bellman equation is

$$u_t + \frac{1}{2}|\nabla u|^2 = 0, \quad u(x,T) = g(x)$$

and the solution formula is

$$u(x,t) = \max_z \left\{ g(z) - \frac{|z-x|^2}{2(T-t)} \right\}.$$

An important fact is immediately evident: the Hamilton-Jacobi-Bellman equation has many (almost-everywhere) solutions, only one of which agrees with the solution formula. For example, suppose $g = 0$. Then the solution formula gives $u(x,t) = 0$, which does solve the Hamilton-Jacobi equation. However the PDE has lots of other solutions: for example the function

$$u(x,t) = \begin{cases} \frac{1}{2}(T-t) - |x| & \text{if } |x| \le \frac{1}{2}(T-t) \\ 0 & \text{otherwise.} \end{cases}$$

This example is easy to generalize, yielding infinitely many "solutions" of the PDE, all equal to 0 at $t = T$. This should not be a great surprise: we already noted the analogous nonuniqueness for the eikonal equation $|\nabla u| = 1$, the HJB equation associated with a geometrical minimum-time problem.

Another point to note: the Hamilton-Jacobi equation is *nonlinear*. If u_1 solves it with final data g_1 and u_2 solves it with final data g_2, we should not expect $u_1 + u_2$ to solve it with final data $g_1 + g_2$. When $H(p) = H(-p)$, for example when $H(p) = |p|^2/2$, one might imagine that if u is the correct solution with final data g then $-u$ is the correct solution with final data $-g$. But even this is false, at least if we understand "correct solution" as "dynamic programming solution" in the sense developed above. Indeed, when $H(p) = |p|^2/2$ and $g(y) = |y|$ the solution formula gives

$$u(x,t) = \frac{T-t}{2} + |x|$$

(the optimal z is $z = x + (T-t)x/|x|$), but when the final-time data is changed to $g(y) = -|y|$ the solution formula gives

$$u(x,t) = \begin{cases} (T-t)/2 - |x| & \text{if } |x| \ge (T-t) \\ -|x|^2/2(T-t) & \text{otherwise} \end{cases}$$

(the optimal z is $z = x - (T-t)x/|x|$ if $|x| \ge (T-t)$, $z = 0$ otherwise).