A new algorithm for finding numerical solutions of optimal feedback control

BAO-ZHU GUO†
Academy of Mathematics and Systems Science, Academia Sinica, Beijing 100080,
People’s Republic of China and School of Computational and Applied Mathematics,
University of the Witwatersrand, Wits 2050, Johannesburg, South Africa

AND

BING SUN
Department of Mathematics, Beijing Institute of Technology, Beijing 100081, People’s
Republic of China and School of Computational and Applied Mathematics, University of the
Witwatersrand, Wits 2050, Johannesburg, South Africa

[Received on 15 March 2007; revised on 17 October 2007]

A new algorithm for finding numerical solutions of optimal feedback control based on dynamic programming is developed. The algorithm is based on two observations: (1) the value function of the optimal control problem considered is the viscosity solution of the associated Hamilton–Jacobi–Bellman (HJB) equation and (2) the appearance of the gradient of the value function in the HJB equation is in the form of directional derivative. The algorithm proposes a discretization method for seeking optimal control–trajectory pairs based on a finite-difference scheme in time through solving the HJB equation and state equation. We apply the algorithm to a simple optimal control problem, which can be solved analytically. The consistence of the numerical solution obtained to its analytical counterpart indicates the effectiveness of the algorithm.

Keywords: optimal feedback control; viscosity solution; dynamic programming; numerical solution; exponential stability.

1. Introduction

Determination of optimal feedback control law has been one of the main problems in modern control theory (Ho, 2005). Among the most celebrated results are the Bellman’s dynamic programming method and the Pontryagin’s maximum principle. The former can give the optimal control law in closed-loop form, but it is analytically effective only to linear systems with quadratic performances. Whereas the latter can be used to deal with non-linear systems, it provides only a necessary condition that an optimal control must satisfy. Nevertheless, it is believed that the condition is also sufficient in most practical applications in view of the existence and uniqueness of optimal control in such cases. However, there are still two important issues that hinder the application of the Pontryagin’s maximum principle to non-linear systems. First, one has to solve a two-point boundary-value problem of a system of non-linear differential equations and second, the solution obtained is usually in open-loop form, implying that it cannot be used directly as a feedback law.

†Email: bzguo@iss.ac.cn

© The author 2009. Published by Oxford University Press on behalf of the Institute of Mathematics and its Applications. All rights reserved.
It appears that numerical methods are the only feasible way to deal with non-linear systems when using Pontryagin’s maximum principle. Up to now both direct and indirect methods have been developed to solve optimal control problems (Pesch, 2003). The indirect method solves the two-point boundary-value problem using the powerful multiple shooting method. It requires a good ‘initial guess’ to start the iterative process, which is not an easy task as that means a good understanding of the physical essence of the system. On the other hand, the direct method starts with a partial or full discretization process of the system equation. The process converts the original problem into a large-scale non-linear programming problem subject to restrictions. The conversion naturally introduces the problem of reliability or accuracy of the method. Moreover, when the degrees of discretization and parameterization become high, the method becomes extremely difficult to manage (Bryson, 1996) due to the complexity of the computation involved. It is further noted that the numerical solutions offered by both methods are still in open-loop form.

On the other hand, it has been realized since Pontryagin’s time that the dynamic programming method which leads to the Hamilton–Jacobi–Bellman (HJB) equation that governs the value function of optimal control problem would give the feedback law provided that the HJB equation is solvable. Thanks to the notion of viscosity solution for the HJB equation introduced in the 1980s (Crandall, 1997), it is now possible to conclude the existence and uniqueness of the HJB equations. One remarkable feature of the viscosity solution is that the value function of optimal control problem is usually also the unique viscosity solution of the associated HJB equation.

While the viscosity solution associated with the HJB equation can lead to optimal control in closed-loop form, it remains that one often has to find the solution numerically rather than analytically. In this paper, we develop a new algorithm for finding numerical solutions of the optimal feedback control and optimal trajectory, which is extracted from our numerical experiments for practical optimal control problems performed in Guo & Sun (2005, 2007). The algorithm is based on the HJB equation and the viscosity solution theory. It is constructive and applicable to finite-dimensional systems (Guo & Sun, 2007) as well as infinite-dimensional ones (Guo & Sun, 2005).

The paper is organized as follows. Some preliminaries related to the theoretical background of the algorithm is provided in Section 2. In Section 3, the new algorithm is constructed step by step for general finite-dimensional optimal control problems. An illustrative example is provided in Section 4, which indicates the effectiveness of the new algorithm.

2. Preliminary

Consider the control system of the following:

\[
\begin{aligned}
\dot{y}(t) &= f(y(t), u(t)), \quad t \in (0, T], \quad T > 0, \\
y(0) &= z,
\end{aligned}
\]

(2.1)

where \(y(\cdot) \in \mathbb{R}^n\) is the state, \(u(\cdot) \in \mathcal{U}[0, T] = L^\infty([0, T]; U)\) is the admissible control with compact set \(U \subset \mathbb{R}^m\) and \(z\) is the initial value. Assume that \(f: \mathbb{R}^n \times U \to \mathbb{R}^n\) is continuous in its variables.

Given a running cost \(L(t, y, u)\) and a terminal cost \(\psi(y)\), the optimal control problem for the system (2.1) is to seek an optimal control \(u^*(\cdot) \in \mathcal{U}[0, T]\), such that

\[
J(u^*(\cdot)) = \inf_{u(\cdot) \in \mathcal{U}[0, T]} J(u(\cdot)),
\]

(2.2)
where $J$ is the cost functional given by

$$J(u(\cdot)) = \int_0^T L(\tau, y(\tau), u(\tau)) d\tau + \psi(y(T)). \quad (2.3)$$

The dynamic programming principle due to Bellman is fundamental in modern optimal control theory. Instead of considering optimal control problem (2.1–2.2), the principle proposes to deal with a family of optimal control problems initiating from (2.1–2.2). That is, consider the optimal control problem for the following system for any $(t, x) \in [0, T] \times \mathbb{R}^n$:

$$\begin{aligned}
    y_t'(s) &= f(y_t, s, u), \quad s \in (t, T], \\
    y_t(t) &= x
\end{aligned} \quad (2.4)$$

with the cost functional

$$J_{t,x}(u(\cdot)) = \int_t^T L(\tau, y_{t,x}(\tau), u(\tau)) d\tau + \psi(y_{t,x}(T)). \quad (2.5)$$

Define the value function

$$V(t, x) = \inf_{u(\cdot) \in \mathcal{U}[t, T]} J_{t,x}(u(\cdot)), \quad \forall (t, x) \in [0, T] \times \mathbb{R}^n, \quad (2.6)$$

with the terminal value

$$V(T, x) = \psi(x), \quad \text{for all } x \in \mathbb{R}^n. \quad (2.7)$$

It is well known that if $V$ is smooth enough, say $V \in C^1([0, T] \times \mathbb{R}^n)$, then $V$ satisfies the following HJB equation (Bardi & Capuzzo-Dolcetta, 1997):

$$\begin{aligned}
    V_t(t, x) + \inf_{u \in \mathcal{U}} \{ f(x, u) \cdot \nabla_x V(t, x) + L(t, x, u) \} &= 0, \quad (t, x) \in [0, T] \times \mathbb{R}^n, \\
    V(T, x) &= \psi(x), \quad x \in \mathbb{R}^n
\end{aligned} \quad (2.8)$$

where $\nabla_x V(t, x)$ stands for the gradient of $V$ in $x$.

The following two propositions show the important role of the value function in characterizing the optimal feedback law (Bardi & Capuzzo-Dolcetta, 1997).

**Proposition 1** Let $V \in C^1([0, T] \times \mathbb{R}^n)$ be the value function. Then, if there exists a control $u^*(\cdot) \in \mathcal{U}[0, T]$ such that

$$f(y^*(t), u^*(t)) \cdot \nabla_x V(t, y^*(t)) + L(t, y^*(t), u^*(t))
\begin{equation}
= \inf_{u \in \mathcal{U}} \{ f(y^*(t), u) \cdot \nabla_x V(t, y^*(t)) + L(t, y^*(t), u) \}, \quad (2.9)
\end{equation}$$

then $u^*(\cdot)$ is an optimal control, where $y^*$ is the state corresponding to $u^*$.

As usual, we denote the optimal control as (Bardi & Capuzzo-Dolcetta, 1997)

$$u^*(t) \in \arg \inf_{u \in \mathcal{U}} \{ f(y^*(t), u) \cdot \nabla_x V(t, y^*(t)) + L(t, y^*(t), u) \} \quad (2.10)$$

for almost all $t \in [0, T]$. 
PROPOSITION 2 Let \( V(t, x) \in C^1([0, T] \times \mathbb{R}^n) \) be the value function. Then, \((u^*(\cdot), y^*(\cdot))\) is an optimal control–trajectory pair in feedback form if and only if

\[
V_t(t, y^*(t)) + f(y^*(t), u^*(t)) \cdot \nabla_x V(t, y^*(t)) + L(t, y^*(t), u^*(t)) = 0
\]

(2.11)

for almost all \( t \in [0, T] \).

By Propositions 1 and 2, we have Theorem 1, which leads to the construction of the feedback law via the value function.

THEOREM 1 Let \( V(t, x) \in C^1([0, T] \times \mathbb{R}^n) \) be the value function. Suppose \( u(t, x) \) satisfies

\[
f(x, u(t, x)) \cdot \nabla_x V(t, x) + L(t, x, u(t, x)) = \inf_{u \in U} \{ f(x, u) \cdot \nabla_x V(t, x) + L(t, x, u) \}.
\]

Then,

\[
u^*_z(t) = u(t, y^*_z(t))
\]

is the feedback law of the optimal control problem (2.1–2.2), where \( y^*_z(t) \) satisfies the following:

\[
y^*_z'(t) = f(y^*_z(t), u(t, y^*_z(t))), \quad y^*_z(0) = z, \quad t \in [0, T].
\]

We see from Theorem 1 that in order to find the feedback control law, not only the value function \( V \) itself but also its gradient \( \nabla_x V \) is needed.

Equation (2.8) generally has no classical solution regardless of the smoothness of the functions \( f \) and \( L \). Fortunately, under some basic assumptions on \( f \) and \( L \), the value function \( V \) is the unique viscosity solution to (2.8). However, it is usually not possible to find analytical solution of (2.8) for general non-linear functions \( f \) and \( L \). It therefore becomes very important to solve (2.8) numerically, particularly in applications. Actually, some difference schemes have been proposed to find the viscosity solutions (Fleming & Soner, 1993; Huang et al., 2000, 2004; Wang et al., 2003).

Once a viscosity solution of (2.8) is obtained numerically, we are able to construct a numerical solution of the feedback law by Theorem 1.

3. Algorithm of finding optimal feedback law

In this section, we follow Theorem 1 to construct an algorithm for finding numerical solutions of optimal feedback control and optimal trajectory pair. The algorithm consists of two steps of discretization. The first one is to discretize the HJB equation (2.8) for the feedback law and the second is to discretize the state equation (2.1) for the optimal trajectory.

In the last two decades, many different approximation schemes have been developed for the numerical solution of (2.8) such as the upwind finite-difference scheme (Wang et al., 2000), the method of vanishing viscosity (Crandall & Lions, 1984) and the parallel algorithm based on the domain decomposition technique (Falcone et al., 1994), name just a few. As for numerical solution of the state equation (2.1), there are numerous classical methods available such as the Euler method, the Runge–Kutta method, the Hamming algorithm and so on (Li & Feng, 1996).

Note that when we use Theorem 1 to find the optimal feedback law, it is the directional derivative \( \nabla_x V \cdot f \) not the gradient \( \nabla_x V \) itself that is needed. This fact greatly facilitates our search for numerical
solutions. The key step is to approximate $\nabla_x V \cdot f$ by its natural definition as following:

$$\nabla_x V(t, x) \cdot f(x, u) = \left[ \nabla_x V(t, x) \cdot \eta \frac{f(x, u)}{1 + \|f(x, u)\|} \right] \frac{1 + \|f(x, u)\|}{\eta} \approx \frac{V(t, x + \eta \frac{f(x, u)}{1 + \|f(x, u)\|}) - V(t, x)}{\eta} (1 + \|f(x, u)\|),$$

(3.12)

where $\eta > 0$ is a small number and $\| \cdot \|$ denotes the Euclidean norm in $\mathbb{R}^n$.

It is pointed out that the above approximation brings obvious advantages to our algorithm presented in this paper. If we try to work out the viscosity solution of (2.8) first, we will most likely be under ‘the curse of dimensionality’ for high-dimensional problems since we have to obtain data for all grids in the whole region. Perhaps that is why the numerical experiments about the viscosity solution of (2.8) studied in many literatures are most limited to 1D or 2D problems (e.g. Wang et al., 2000). On the other hand, since our scheme searches the optimal control only along the direction of $f$ not in the whole region, the new algorithm involves much less data. This idea is also applicable to infinite-dimensional systems (Guo & Sun, 2005). To the best of our knowledge, there has been no effort along this direction to find optimal control by dynamic programming.

Based on above discussion, we now construct the algorithm for the numerical solutions of the optimal feedback control–trajectory pairs.

**Step 1: Initial partition on time and space.** Select two positive integers $N$ and $M$. Let $t_j = T + j \Delta t, j = 0, 1, \ldots, N$, be a backward partition of $[0, T]$, where $\Delta t = -T/N$. For any initial given $\tilde{u} \in U$, let initial state $x_0 = z$ and

$$x_i = x_{i-1} + \eta \frac{f(x_{i-1}, \tilde{u})}{1 + \|f(x_{i-1}, \tilde{u})\|}, \quad i = 1, 2, \ldots, M.$$  

(3.13)

**Step 2: Approximate (2.8).** This is done by the finite-difference scheme in time and space mesh approximation via (3.12):

$$\begin{align*}
\frac{V_i^{j+1} - V_i^j}{\Delta t} + \frac{V_i^{j+1} - V_i^j}{\eta} \cdot (1 + \|f_i^j\|) + L_i^j &= 0, \\
\arg \inf_{u \in U} \left\{ \frac{V_i^{j+1} - V_i^j}{\eta} \cdot (1 + \|f(x_i, u)\|) + L(t_{j+1}, x_i, u) \right\}
\end{align*}$$

(3.14)

for $i = 0, 1, \ldots, M$ and $j = 0, 1, \ldots, N - 1$, where $V_i^j = V(t_j, x_i), f_i^j = f(x_i, u_i^j)$ and $L_i^j = L(t_j, x_i, u_i^j)$.

It is assumed that

$$\frac{|\Delta t|}{\eta} \left( 1 + \max_{i, j} \|f_i^j\| \right) \leq 1,$$

(3.15)

which is a sufficient condition for the stability of the finite-difference scheme (Li & Feng, 1996).
Step 3: Initialization of value function and control. Let
\[ V_i^0 = \psi(x_i), \]
\[ u_i^0 \in \arg \inf_{u \in U} \left\{ \frac{V_{i+1}^0 - V_i^0}{\eta} \cdot (1 + \|f(x_i, u)\|) + L(T, x_i, u) \right\}, \quad i = 0, 1, \ldots, M. \]

Step 4: Iteration for the HJB equation. By (3.14) and Step 3, we obtain all \( \{V_i^j\}_{i=0}^M \) and \( \{u_i^j\}_{i=0}^M \):
\[ \begin{cases} 
V_i^{j+1} = \left(1 + \frac{\Delta t}{\eta}(1 + \|f_i^j\|)\right)V_i^j - \frac{\Delta t}{\eta}(1 + \|f_i^j\|)V_i^{j+1} - \Delta tL_i^j, \\
u_i^{j+1} \in \arg \inf_{u \in U} \left\{ \frac{V_{i+1}^{j+1} - V_i^{j+1}}{\eta} \cdot (1 + \|f(x_i, u)\|) + L(t_{j+1}, x_i, u) \right\}, 
\end{cases} \]
(3.16)
where \((u_0^N, y^0) = (u_0^N, y(0)) = (u(0), z)\) is the first optimal feedback control–trajectory pair.

Step 5: Iteration for state equation. Solve the state equation
\[ \frac{y^{j+1} - y^j}{-\Delta T} = f(y^j, u(t_{N-j})) \]
to obtain \(y^1 = y(t_{N-1})\). Replace \((\tilde{u}, z)\) in Step 1 by \((u(0), y^1)\) and goto Steps 1 and 3 and then Step 4 to obtain \(u_0^{N-1}\). Then, \((u_0^{N-1}, y^1) = (u(t_{N-1}), y(t_{N-1}))\) is the second optimal feedback control–trajectory pair.

Step 6: Continuation of iteration. Once \((u(t_{N-j}), y^j) = (u(t_{N-j}), y(t_{N-j}))\) is known, solve the state equation
\[ \frac{y^{j+1} - y^j}{-\Delta T} = f(y^j, u(t_{N-j})) \]
to obtain \(y^{j+1} = y(t_{N-j-1})\). Replace \((\tilde{u}, z)\) in Step 1 by \((u(t_{N-j}), y^{j+1})\) and goto Steps 1 and 3 and then Step 4 to obtain \(u_0^{N-j-1}\). Then, \((u_0^{N-j-1}, y^{j+1}) = (u(t_{N-j-1}), y(t_{N-j-1}))\) is the \((j + 2)\)th optimal feedback control–trajectory pair. Continue the iteration to obtain all \((u(t_{N-j}), y(t_{N-j}))\) for \(j = 0, 1, \ldots, N\).

After Steps 1–6, we finally get all the desired optimal feedback control–trajectory pairs:
\[ (u(t_{N-j}), y(t_{N-j})), \quad j = 0, 1, \ldots, N. \]

It should be pointed out the above algorithm has generalization for infinite-dimensional systems (see Guo & Sun, 2005).

It is worth noting that the focus of the above algorithm is not to solve the HJB equation, not even to obtain the value function itself. This is different from most literatures (e.g. see Huang et al., 2000, 2004; Wang et al., 2003) in the field, which focus on solving the HJB equations. Our ultimate aim is to find the numerical solutions of both the optimal feedback control and the corresponding optimal
trajectory. The whole algorithm consists of solving the state equation one time and the HJB equation $N$

According to our numerical experiments reported in Guo & Sun (2005, 2007) and the example presented in Section 4, the mesh-point sequence in space generated by the recursive relation (3.13) does not bring oscillation of the space variable in subregions of the given space even when $f$ changes its sign. This is because (3.12) is the more proper definition of the directional derivative, which allows us to search the optimal control along its natural direction $f$.

If only the solution of the HJB equation (2.8) is concerned, for instance computing the values of $V$ in the polyhedral $\prod_{i=1}^{n}[a_i, b_i]$ of $\mathbb{R}^n$, we have to produce the monotone mesh-point sequence in space generated by the recursive relation (3.13). To this purpose, we have to change the searching direction forcefully. In this case, we suggest to use the following approximation for directional derivative instead of using the approximation of (3.12). Specifically, for every fixed $(x, u), x = (x^1, x^2, \ldots, x^n), f = (f^1, f^2, \ldots, f^n),$

$$\nabla_x V(t, x) \cdot f(x, u) = \sum_{p=1}^{n} \left[ V(x) - V(t, x) \cdot \eta \frac{f^p(x, u) \text{sgn}(f^p(x, u))}{1 + \|f^p(x, u)\|} \right] 1 + \|f^p(x, u)\| \text{sgn}(f^p(x, u)) \tag{3.17}$$

$$\approx \sum_{p=1}^{n} V(t, x + \eta \frac{f^p(x, u) \text{sgn}(f^p(x, u))}{1 + \|f^p(x, u)\|} I_p) - V(t, x) 1 + \|f^p(x, u)\| \text{sgn}(f^p(x, u)),$$

where

$$\text{sgn}(\varphi) = \begin{cases} 1, & \text{if } \varphi \geq 0, \\ -1, & \text{if } \varphi < 0, \end{cases} \tag{3.18}$$

$I_p$ is the $n$-dimensional unit vector with the $p$th component 1 and $V_{x^p}$ is the partial derivative of the value function $V$ with respect to $x^p$.

In this way, the possible oscillation caused by sign change of the function $f$ can be avoided when we use the formula (3.13) to perform the spatial mesh partition. Accordingly, the mesh-point sequence in space generated by the recursive relation (3.13) now becomes

$$x^p_i = x^p_{i-1} + \eta \frac{\text{sgn}(f^p(x_{i-1}, \tilde{u})) f^p(x_{i-1}, \tilde{u})}{1 + \|f^p(x_{i-1}, \tilde{u})\|}, \quad p = 1, 2, \ldots, n, \quad i = 1, 2, \ldots, M, \tag{3.19}$$

where $x_k = (x^1_k, x^2_k, \ldots, x^n_k), k = i - 1, i$.

4. An example

In this section, we provide an illustrative example to show the effectiveness of our new algorithm. Both the value function and the feedback law have closed-form solutions in this example. This allows us to evaluate the numerical solutions obtained by our algorithm. Note that in this example, the value function is not the classical solution; rather it is the viscosity solution to its associated HJB equation (Yong, 1992).

Consider the state equation governed by

$$\begin{cases} y'(t) = u(t), & t \in (0, 1), \\ y(0) = z, \end{cases} \tag{4.20}$$
for which the state space is $\mathbb{R}$. The control constraint $U = [-1, 1]$ and hence $U[0, 1] = L^\infty([0, 1]; U)$. The cost functional is
\[
J(u(\cdot)) = -y^2(1).
\]
(4.21)

Now, we introduce the value function of the problem. For any $(t, x) \in [0, 1) \times \mathbb{R}$, consider the following system:
\[
\begin{align*}
    y_{t,x}'(s) &= u(s), & s & \in [t, 1], \\
    y_{t,x}(t) &= x.
\end{align*}
\]
(4.22)

Accordingly, the cost functional is
\[
J_{t,x}(u(\cdot)) = -y_{t,x}^2(t).
\]

Define the value function
\[
V(t, x) = \inf_{u(\cdot) \in U(t, 1)} J_{t,x}(u(\cdot)).
\]
(4.23)

A simple calculation gives that
\[
V(t, x) = \begin{cases}
-|x + (1 - t)|^2, & x \geq 0, \\
-|x - (1 - t)|^2, & x \leq 0,
\end{cases}
\]
\[= -|[x] + (1 - t)|^2, \quad x \in \mathbb{R}, \quad t \in [0, 1].
\]
(4.24)

It is seen that $V$ is continuous in its variable but not in $C^1([0, 1] \times \mathbb{R})$. The associated HJB equation is
\[
\begin{cases}
V_t(t, x) + \inf_{|u| \leq 1} \{V_x(t, x) \cdot u = 0, \quad (t, x) \in [0, 1) \times \mathbb{R}, \\
V(1, x) = -x^2, \quad x \in \mathbb{R}.
\end{cases}
\]
(4.25)

From Yong (1992), we know that the value function $V$ is the unique viscosity solution to the HJB equation (4.25) and the optimal feedback control–trajectory pair is given analytically as
\[
u_{t,x}^*(s) = \text{sign}(y_{t,x}^*(s)),
\]
(4.26)

where
\[
\text{sign}(y) = \begin{cases}
1, & y > 0, \\
\{\pm 1\}, & y = 0, \\
-1, & y < 0,
\end{cases}
\]

and
\[
y_{t,x}^*(s) = \begin{cases}
x + (s - t), & x \geq 0, \\
x - (s - t), & x \leq 0.
\end{cases}
\]
(4.27)

We apply the algorithm in Section 3 step by step and obtain the numerical solutions of the optimal feedback control–trajectory pairs for different initial values from $-1$ to $1$ with step size $0.2$. The parameters are taken as $N = M = 100$ and $\eta = 0.02$. All computations are performed in Visual C++ 6.0 and numerical results are plotted by MATLAB 6.0.

Figure 1 shows the obtained numerical solutions of the optimal trajectories, where the solid lines denote the analytical solutions (4.27). It is seen that the analytical solutions and numerical solutions
FIG. 1. Numerical solutions of the optimal trajectories.

FIG. 2. Numerical solutions of the optimal feedback laws.

completely coincide with each other. Figure 2 displays the computed numerical solutions of the optimal feedback laws where solid lines denote the analytical solutions (4.26). They also completely match with the analytical solutions (4.26).

Acknowledgements

The authors would like to thank an anonymous referee for indicating possible oscillations due to sign change of $f$ of the mesh-point sequence in space generated by the recursive relation (3.13) in computation.
of the HJB equation. This prompts us to use the modified mesh-point consequence (3.19) produced by (3.17).

**Funding**

National Natural Science Foundation of China; National Research Foundation of South Africa.

**REFERENCES**


