Yokohama Publishers
ISSN 1349-8169 ONLINE JOURNAL

Pacific Journal of Optimization
Vol. 12, No. 2, pp. 379–397, April 2016

ITERATIVE UPWIND FINITE DIFFERENCE METHOD WITH COMPLETED RICHARDSON EXTRAPOLATION FOR STATE-CONSTRAINED HJB EQUATIONS

Hartono, L.S. Jennings and S. Wang*

Abstract: In this work, we develop an iterative method in conjunction with an upwind finite difference discretization scheme for solving a Hamilton-Jacobi-Bellman (HJB) equation governing a class of state-constrained optimal feedback control problems. We prove that the method is stable. We also propose an algorithm for computational domain reduction and a completed Richardson extrapolation technique to improve the accuracy of numerical solutions from the method. Numerical results will be presented to demonstrate the accuracy and efficiency of the method.

Key words: *optimal control, Hamilton-Jacobi-Bellman equation, finite difference method, penalty function, Richardson extrapolation.*

Mathematics Subject Classification: *65M06, 49L25, 90C39.*

1 Introduction

Many real-world problems in classic and financial engineering are governed by the following optimal problem:

Problem 1.1.

$$\begin{aligned} \min_{u(t)\in U} \quad & J(u) = \int_s^{t_f} L(x(t),u(t),t)\,dt + \phi(x(t_f)) \\ \text{subject to} \quad & \dot{x} = f(x(t),u(t),t), \qquad \text{for all } \ t \in (s,t_f], \\ & x(s) = y, \end{aligned}$$

where $x \in \mathbb{R}^n$ and $u \in U \subset \mathbb{R}^m$ are the state and control, U denotes the set of all admissible controls, $t_f > 0$ is a constant, $(s,y) \in [0,t_f) \times \mathbb{R}^n$, $y \in \mathbb{R}^n$ is a given point. $L : \mathbb{R}^{n+m+1} \to \mathbb{R}$, $\phi : \mathbb{R}^n \to \mathbb{R}$ and $f : \mathbb{R}^{n+m+1} \to \mathbb{R}^n$ are known functions.

If $s = 0$ and $y = x_0$ is fixed then Problem 1.1 reduces to an open-loop optimal control problem. The solution to the open-loop problem constitutes an optimal control u along the optimal trajectory x from the initial state x_0. However, this solution is neither robust nor stable with respect to perturbations in the state x. This is because if perturbations exist in the state x such that it is off the optimal trajectory, then the corresponding optimal

*S. Wang's work was partially supported by the AOARD Project #15IOA095 from the US Air Force.

control solution is no longer available, as the open-loop solution becomes non-optimal. On the other hand, the solution to the closed-loop problem is robust or stable because it is defined over a time-space region that contains all possible optimal trajectories in the region. Hence, the corresponding optimal control can still be determined without resolving the problem even if disturbances are present in the state x. Therefore, optimal closed-loop or feedback controllers are much preferred in practice than open-loop ones. However, the above problem is usually not analytically solvable except for some trivial cases. Therefore, numerical approximations are always sought in practice which requires efficient, stable and accurate numerical methods.

Before further discussion, we firstly need to transform the problem into a first order partial differential equation called the Hamilton-Jacobi-Bellman (HJB) equation. By defining the value function

$$V(s,y) = \inf_{u\in U} J(s,y,u)$$

and using the Dynamic Programming approach, it is well known that the above optimal feedback control problem can be written as the following HJB equation

$$\frac{\partial V}{\partial t} + \inf_{u\in U}\left[\nabla V \cdot f(x,u,t) + L(x,u,t)\right] = 0 \tag{1.1}$$

with terminal condition

$$V(t_f,x) = \phi(x(t_f)). \tag{1.2}$$

In this equation there are two unknowns, the value function V and the optimal control u.

Generally, the solution to HJB equation is continuous but nonsmooth. To deal with this nonsmooth solution, the concept of viscosity solution was introduced by Lions et al [5], [6] and [19]. In the presence of state constraints, Soner [29] broadened the definition of the viscosity solution to the constrained viscosity solution. More detail about viscosity solutions of HJB equations can be found in [3] and [7].

Although the HJB equation system (1.1)–(1.2) has theoretically a unique solution under certain assumptions, it is in general impossible to solve (1.1) analytically. In practice, numerical approximations to the solution of (1.1) is always sought as mentioned above. Therefore, numerical methods are crucial for the accurate approximation of (1.1). There are many numerical methods available in the literature for solving unconstrained HJB equations, such as those in [1–3, 8, 9, 13, 14, 26, 35, 36] to name but a few. Among these methods, we are interested in the upwind finite difference method introduced in [35] due to its simplicity. For one dimensional problems, we let the spatial and time intervals be partitioned uniformly into M and N subintervals, respectively, with respective mesh sizes Δx and Δt. Then, the application of the method in [35] to (1.1) yields the following discrete system:

$$\frac{V_i^{k+1}-V_i^k}{\Delta t} + \frac{1+\operatorname{sign} f_i^k}{2} f_i^k \frac{V_{i+1}^k - V_i^k}{\Delta x} + \frac{1-\operatorname{sign} f_i^k}{2} f_i^k \frac{V_{i-1}^k - V_i^k}{\Delta x} + L_i^k = 0$$

$$u_i^{k+1} = \arg\inf_u \left(f(x_i,u,t_{k+1}) \frac{V_i^{k+1} - V_{i-1}^{k+1}}{\Delta x} + L(x_i,u,t_{k+1}) \right)$$

for $k = 1,\ldots,N$ and $i = k,\ldots,M-k$ along with terminal condition (1.2), where $\operatorname{sign} f_i^k$ denotes the sign of f_i^k,

$$f_i^k = f(x_i,t_k,u_i^k), \quad L_i^k = L(x_i,t_k,u_i^k), \quad V_i^k \approx V(x_i,t_k), \quad u_i^k \approx u(x_i,t_k),$$

where x_i and t_k denote the mesh points in the x- and t-directions respectively. In the above discretization scheme the upwind technique is used, that is if $f_i^k > 0$ the scheme switches to the forward-difference scheme and to backward-difference scheme for the opposite sign.

However, there is a drawback, namely trapezoidal propagation of the spatial domain with each time step as discussed in [27, 35]. This is because from the discretization one can see that, for given i and k, V_i^{k+1} is determined by the three previous values V_{i-1}^k, V_i^k and V_{i+1}^k. This propagation causes a large initial spatial region so that it leads to expensive computation due to a greater number of computed grid points. To address this problem and to improve the speed and accuracy of the method, we introduce an Iterative Upwind Finite Difference Method (inspired by Luus [20]) in combination with Completed Richardson Extrapolation ([24]).

Note that although Problem 1.1 contains constraints on the control variable u, it is not state-constrained. Clearly, if state constraints are present in Problem 1.1, the resulting optimal feedback control problem is much harder to solve numerically than Problem 1.1. To our best knowledge, there are essentially no numerical methods in the open literature for the approximation of the viscosity solutions Problem 1.1 with state constraints. On the other hand, there are many such problems in practice as state constraints are very often imposed for an optimal feedback control problem. In this work, we will develop an iterative upwind finite difference method for an HJB equation governing a class of state-constrained optimal feedback control problem, based on the above upwind finite difference scheme. The iteration part of the method applies to the doubling of the number of discrete x-values in order to gain better accuracy. Adjustments from iteration to iteration are designed to create efficiencies. In order to iterate without a trapezoidal propagation of the spatial domain in each time step, we impose artificial boundary conditions to be explained later.

As is commonly known, Richardson extrapolation is a technique for improving the order of accuracy of numerical results. The main idea behind this technique is as follows. If the rate of convergence of a discretization method with grid refinement is known and if discrete solutions on two systematically refined grids (coarse and fine grids) are available, then this information can be used to provide higher-order solution on the coarse grid. As a result, it is easily implemented as a postprocessor to solutions regardless of the methods or equations producing them (cf., for example, [23]). In this work, we will use this technique to improve the accuracy of the approximate solutions from our method.

2 Iterative Upwind Finite Difference Method

Let us consider the following state-constrained optimal feedback control problem:

Problem 2.1.

$$\begin{aligned} \min_{u(t)\in\Omega} \quad & J(u) = \int_s^{t_f} L(x(t), u(t), t)\, dt + \phi(x(t_f)) \\ \text{subject to} \quad & \dot{x} = f(x(t), u(t), t), \quad \text{for all } \ t \in (s, t_f], \\ & x(s) = y, \end{aligned}$$

where

$$\Omega = \{u(t) \in \mathbb{R}^q \mid g(x, u, t) \le 0\}, \quad \text{for all } \ t \in (0, t_f],$$

$t_f > 0$ is a constant, $(s, y) \in [0, t_f) \times \mathbb{R}^n$, $x \in \mathbb{R}^n$, $y \in \mathbb{R}^n$ is a given point and $L : \mathbb{R}^{n+q+1} \to \mathbb{R}$, $\phi : \mathbb{R}^n \to \mathbb{R}$, $f : \mathbb{R}^{n+q+1} \to \mathbb{R}^n$ and $g : \mathbb{R}^{n+q+1} \to \mathbb{R}^m$ are known functions.

Clearly, the set Ω is determined by the constraints on both x and u. To develop our numerical method for Problem 2.1, we first convert it to an unconstrained optimal control problem by incorporating linear penalty terms in the objective function as given below.

Problem 2.2.

$$\begin{aligned} \min_{u(t)\in\mathbb{R}^q} \quad & P(u,r) = J(u) + \int_s^{t_f} r^T(t) g_\rho(x(t),u(t),t)dt \\ \text{subject to} \quad & \dot{x} = f(x(t),u(t),t) \ \text{ for all } t \in (s,t_f], \\ & x(s) = y, \end{aligned}$$

where $r = (r_1(t), r_2(t), \ldots, r_m(t))^T$ is a vector satisfying $r_i(t) > 0$ for i=1,2,...,m and $g_\rho : \mathbb{R}^{n+q+1} \to \mathbb{R}$ is the smoothed version of $[g]_+ = ([g_1]_+, [g_2]_+, \ldots, [g_m]_+)^T$ defined as

$$g_\rho(x,u,t) = \begin{cases} 0 & \text{if } g < -\rho; \\ \frac{(g+\rho)^2}{4\rho} & \text{if } -\rho \le g \le \rho; \\ g & \text{if } g > \rho. \end{cases},$$

where ρ is a chosen positive constant and $[z]_+ := \max\{0, z\}$ for any z.

Penalty methods have been used very successfully for solving constrained optimization problems and for optimal control and constrained optimization problems (cf., for example, [12, 17, 18, 31, 33]). The r in Problem 2.2 represents a set of penalty constants which are usually chosen to be large positive numbers. Note that in this case smoothing the sharp corner of $[g]_+$ at zero is necessary with the aim of applying standard gradient-based optimization routines. Using a standard dynamic programming argument, it is easy to show that the HJB equation corresponding to Problem 2.2 is as follows:

$$\frac{\partial V}{\partial t} + \inf_u \left[\nabla V \cdot f(x,u,t) + L(x,u,t) + r g_\rho(x,u,t)\right] = 0 \tag{2.1}$$

with the terminal condition (1.2)

Remark 2.3. For the convergence analysis of the linear penalty on constrained viscosity solution of HJB equations, we refer to [3], [4] and [21].

Note that Richardson Extrapolation usually requires some assumptions such as smoothness and asymptotic range of the solution. Smoothness of the solution is important because the analysis of Richardson Extrapolation is based on Taylor series expansion. In order to have a smooth solution as required by Richardson Extrapolation method, we need to change equation (2.1) to the singularly perturbed convection-diffusion equation

$$\frac{\partial V}{\partial t} + \inf_{u\in\mathbb{R}^q} \left[\nabla V \cdot f(x,u,t) + L(x,u,t) + r g_\rho(x,u,t)\right] + \varepsilon \nabla^2 V = 0 \tag{2.2}$$

with the terminal condition (1.2). The difference between (2.1) and (2.2) is only the diffusion term $\varepsilon\nabla^2 V$, $\varepsilon > 0$, which represents a small perturbation parameter. As $\varepsilon \to 0$ the solution of (2.2) converges to the solution (2.1) (see [3]).

We now present our numerical method for (2.2) and (1.2). To simplify notation, let us consider an optimal control problem with one control u and one state variable $x \in [a,b]$. Extension to a multivariable optimal control problem can be easily done with some adjustments to notation. In addition, without loss of generality and for the intention of numerical tests later, we will set $s = 0$, $y = x_0$ and $t_f = 1$.

2.1 Discretization of HJB

We start with constructions of spatial discretization and time stages. We select a positive integer M and divide the space interval $[a, b]$ into M equal partitions so that

$$x_i = a + (i-1)\Delta x \quad \text{with} \quad \Delta x = \frac{b-a}{M}$$

for $i = 1, \ldots, M+1$.

In order that this spatial discretization always contains the initial point, an appropriate shifting might be necessary. Let $j = \arg\min_i |x_i - x_0|$ and make the adjustments

$$x_i \leftarrow x_i + (x_0 - x_j), \quad a \leftarrow a + (x_0 - x_j), \quad b \leftarrow b + (x_0 - x_j)$$

for $i = 1, \ldots, M$. Next, we impose a limit on control $u(t)$ where $t \in [0, 1]$, specifically, the lower bound u_{l} and the upper bound u_{u}. This constraint is usually determined by physical limits of the system control values.

The time interval $[0, 1]$ is then divided into N equal partitions with $\Delta t = -\frac{1}{N}$ so that

$$t_k = 1 + (k-1)\Delta t, \quad k = 1, \ldots, N+1$$

is the backward partition. This means that t_1 and t_{N+1} correspond to $t = 1$ and $t = 0$ respectively.

With notation $V_i^k \approx V(t_k, x_i)$ and $u_i^k \approx u(t_k, x_i)$ for the value function and control variable at point x_i and time t_k, we split the equation (2.2) into 2 equations and discretize it for $i = 1, \ldots, M+1$ as follows

$$\begin{aligned} \frac{V_i^{k+1} - V_i^k}{\Delta t} &+ \frac{1 + \operatorname{sign} f_i^k}{2} f_i^k \frac{V_{i+1}^k - V_i^k}{\Delta x} + \frac{1 - \operatorname{sign} f_i^k}{2} f_i^k \frac{V_i^k - V_{i-1}^k}{\Delta x} + L_i^k \\ &+ r\, g_{\rho,i}^k + \varepsilon \frac{V_{i-1}^k - 2V_i^k + V_{i+1}^k}{(\Delta x)^2} = 0 \end{aligned} \tag{2.3}$$

$$u_i^{k+1} = \arg\inf_u \left(f(x_i, u, t_{k+1}) \frac{V_i^{k+1} - V_{i-1}^{k+1}}{\Delta x} + L(x_i, u, t_{k+1}) + r g_\rho(x_i, u, t_{k+1}) \right),$$

where $f_i^k = f(x_i, t_k, u_i^k), L_i^k = L(x_i, t_k, u_i^k), g_{\rho,i}^k = g_\rho(x_i, t_k, u_i^k)$ and $\operatorname{sign} f_i^k$ denotes the sign of f at point x_i and time t_k. In (2.3) the spatial derivative $\frac{\partial V}{\partial x}$ is approximated by the first-order upwind finite difference scheme which has been widely used for convection-diffusion and various types of HJB equations [11, 16, 18, 28] Using $\eta_1 = \frac{\Delta t}{\Delta x}$ and $\eta_2 = \frac{\Delta t}{(\Delta x)^2}$, (2.3) can be rewritten as follows:

$$\begin{aligned} V_i^{k+1} &= (1 + \eta_1 |f_i^k| + 2\varepsilon\eta_2) V_i^k - \left[\frac{1 + \operatorname{sign} f_i^k}{2} \eta_1 f_i^k + \varepsilon\eta_2 \right] V_{i+1}^k \\ &+ \left[\frac{1 - \operatorname{sign} f_i^k}{2} \eta_1 f_i^k - \varepsilon\eta_2 \right] V_{i-1}^k - \Delta t (L_i^k + r\, g_{\rho_i}^k) \end{aligned} \tag{2.4}$$

for $i = 2, \ldots, M$. Because the Upwind Finite-Difference Method is an explicit method, we need to consider the stability of the scheme under some conditions on the step lengths Δt and Δx. This is given in the following theorem.

Theorem 2.4. *Under the condition*

$$N \geq \frac{\Delta x \, \|f\|_\infty + 2\varepsilon}{(\Delta x)^2}. \tag{2.5}$$

the scheme (2.4) is stable.

Proof. It is known from [30], that the scheme (2.4) is stable if and only if with $L + rg_\rho = 0$ it is also stable. The scheme (2.4) with $L + rg_\rho = 0$ is equivalent to

$$\begin{aligned} V_i^{k+1} &= \left[1 + \frac{1+\text{sign} f_i^k}{2}\eta_1 f_i^k - \frac{1-\text{sign} f_i^k}{2}\eta_1 f_i^k + 2\varepsilon\eta_2\right] V_i^k \\ &- \left[\frac{1+\text{sign} f_i^k}{2}\eta_1 f_i^k + \varepsilon\eta_2\right] V_{i+1}^k + \left[\frac{1-\text{sign} f_i^k}{2}\eta_1 f_i^k - \varepsilon\eta_2\right] V_{i-1}^k \end{aligned}$$

Introduce the discrete maximum norm $||\cdot||_\infty$ defined by

$$\|V^k\|_\infty = \max_{1\leq i\leq M+1} |V_i^k|$$

and let

$$\alpha_i^k = -\frac{1+\text{sign} f_i^k}{2}\eta_1 f_i^k - \varepsilon\eta_2, \quad \beta_i^k = \frac{1-\text{sign} f_i^k}{2}\eta_1 f_i^k - \varepsilon\eta_2.$$

Then, we have the following two cases.

1. If $f_i^k > 0$ then

$$\begin{aligned} \alpha_i^k &= -\eta_1 f_i^k - \varepsilon\eta_2 = -\frac{\Delta t}{\Delta x}\left(f_i^k + \frac{\varepsilon}{\Delta x}\right) > 0 \\ \beta_i^k &= -\varepsilon\eta_2 = -\frac{\Delta t}{\Delta x}\varepsilon > 0, \end{aligned}$$

   since $\Delta t < 0$.

2. If $f_i^k \leq 0$ then

$$\begin{aligned} \alpha_i^k &= -\varepsilon\eta_2 = -\frac{\Delta t}{\Delta x}\varepsilon > 0 \\ \beta_i^k &= \eta_1 f_i^k - \varepsilon\eta_2 = -\frac{\Delta t}{\Delta x}\left(|f_i^k| + \frac{\varepsilon}{\Delta x}\right) > 0. \end{aligned}$$

Therefore, under the condition $0 < \alpha_i^k + \beta_i^k \leq 1$, we have

$$\begin{aligned} |V_i^{k+1}| &= |(1-\alpha_i^k-\beta_i^k)V_i^k + \alpha_i^k V_{i+1}^k + \beta_i^k V_{i-1}^k| \\ &\leq (1-\alpha_i^k-\beta_i^k)|V_i^k| + \alpha_i^k|V_{i+1}^k| + \beta_i^k a|V_{i-1}^k| \\ &\leq \|V^k\|_\infty \end{aligned}$$

for $k = 0, 1, \ldots, N$. In particular, when $k+1 = N$, taking the maximum on both sides of the above with respect to i and using the recursive relationship, we get

$$\|V^{N+1}\|_\infty \leq \|V^N\|_\infty \leq \cdots \leq \|V^1\|_\infty.$$

Therefore, the scheme is stable under the condition $\alpha_i^k + \beta_i^k \leq 1$.

From their definitions it is to see that

$$\alpha_i^k + \beta_i^k = -\eta_1 |f_i^k| - 2\varepsilon\eta_2 = -\frac{\Delta t}{\Delta x^2}\left(|f_i^k|\Delta x + 2\varepsilon\right) \leq \frac{1}{N\Delta x^2}\left(||f||_\infty \Delta x + 2\varepsilon\right),$$

where $N = -1/\Delta t$ and $||f||_\infty$ denotes the discrete maximum norm of f. Therefore, a sufficient condition for the stability of the numerical scheme is given in (2.5). We have thus proved the theorem. □

Remark 2.5. Note that the stability condition (2.5) is the extension of the stability condition in [35] in the case for $\varepsilon \neq 0$.

Next, we set the initial value function according to

$$V_i^1 = \phi(x_i(1)),\ i = 1, \ldots, M+1$$

and initial control value for $i = 2, \ldots, M+1$

$$u_i^1 = \arg\min\left(f(t_1, x_i, u)\frac{V_i^1 - V_{i-1}^1}{\Delta x} + L(t_1, x_i, u) + r g_\rho(t_1, x_i, u)\right).$$

As mentioned before, the HJB problem (1.1)–(1.2) is a pure initial (or terminal) value problem and the application of the upwind finite difference scheme to it requires a proper trapezoidal region due to the propagation of the numerical scheme which is computationally very expensive. The use of the artificial diffusion term in (2.2) can help to remedy this problem because the effect of a trapezoidal propagation on the spatial domain for each time stage can be avoided by setting up some artificial boundary conditions for control and value function based on linear extrapolation of the closest known points. Theoretically, any artificial boundary conditions can be used as we expect that when ε is sufficiently small, the problem may display boundary layers so that the inaccuracy due the artificial boundary conditions is only inside the layers [22,34]. Therefore, the linear extrapolation to boundaries is chosen because it is simple to apply in computation. In this case, the boundary conditions behave like $u_{xx} = 0 = V_{xx}$ or $u_{yy} = 0 = V_{yy}$. Moreover, it gives freedom for the edge points to flip following the line directed by the values of two closest points. Thereby, for $i = 1$

$$u_1^1 = 2u_2^1 - u_3^1.$$

Next, we update the value function for $k = 1, \ldots, N$ and $i = 2, \ldots, M$ according to (2.4) and do extrapolations for both boundaries

$$V_1^{k+1} = 2V_2^{k+1} - V_3^{k+1}, \quad V_{M+1}^{k+1} = 2V_M^{k+1} - V_{M-1}^{k+1}.$$

Moreover, to update control we set for $k = 1, \ldots, N$ and $i = 2, \ldots, M+1$,

$$u_i^{k+1} = \arg\min_u\left(f(t_{k+1}, x_i, u)\frac{V_i^{k+1} - V_{i-1}^{k+1}}{\Delta x} + L(t_{k+1}, x_i, u) + r\, g_\rho(t_{k+1}, x_i, u)\right)$$

and for the left boundary

$$u_1^{k+1} = 2u_2^{k+1} - u_3^{k+1}.$$

So far we have obtained V_i^k and u_i^k for $i = 1, \ldots, M+1$ and $k = 1, \ldots, N+1$. These constitute the first iteration of the method.

2.2 Finding Optimal Trajectory and Control

To iterate, we first need to determine the optimal trajectory from the first iteration. Starting with the initial value, we integrate forward the state equation $\dot{x} = f(t, x, u)$ using the following predictor-corrector method. Let us name the resultant trajectory and control y_p and u_p for predictor, y_c and u_c for corrector with $y_p(1) = y_c(1) = x_0$ and $u_c(1) = u(x_0, t_{N+1})$ respectively.

The control value used during the integration is the optimal control value corresponding to the closest grid point to the resultant state as suggested in [32]. Thus, for $l = 2, \ldots, N+1$

$$\begin{aligned}
y_p(l) &= y_c(l-1) - \Delta t f(t_{-l+N+3}, y_p(l), u_c(l-1)) \\
u_p(l) &= u(x_{i^*}(l), t_{-l+N+3}) \\
&\quad \text{where } i^* = \arg\min_i |y_p(l) - x_i|,\ i = 1, \ldots, M+1 \\
y_c(l) &= y_c(l-1) - \tfrac{1}{2}\Delta t(f(t_{-l+N+3}, y_c(l-1), u_c(l-1)) \\
&+ f(t_{-l+N+3}, y_p(l), u_p(l))) \\
u_c(l) &= u(x_{i^*}, t_{-l+N+3}) \\
&\quad \text{where } i^* = \arg\min_i |y_c(l) - x_i|,\ i = 1, \ldots, M+1.
\end{aligned}$$

The resultant pair $(y_c(l), u_c(l))$ for $l = 1, \ldots, N+1$ makes an optimal trajectory and control for all time steps from the first iteration of the HJB. In addition, the value function along the optimal trajectory can be determined by the value function of the corresponding closest grid points. The penalty value and objective function value can also be evaluated by forward integration along the optimal trajectory of corresponding terms.

2.3 Region Size Reduction

Now, we determine a procedure for region reduction based on the optimal trajectory and control from previous iteration. This new region is applied to the next iteration in order to improve computational speed and accuracy. What we need, first, is the maximum and the minimum value of resultant control and trajectory. Thus, for $l = 1, \ldots, N+1$

$$\begin{aligned}
x_{\max} &= \max y_c(l), \quad x_{\min} = \min y_c(l) \\
u_{\max} &= \max u_c(l), \quad u_{\min} = \min u_c(l)
\end{aligned}$$

In view of the fact that for the next iteration the number of interval partitions M will be doubled, we set the region for the next iteration as follows:

$$a = x_{\min} - c\Delta x, \quad b = x_{\max} + c\Delta x, \quad M = 2M$$

and the lower and upper bound for the control

$$u_{\mathrm{l}} = \lfloor u_{\min} \rfloor, \quad u_{\mathrm{u}} = \lceil u_{\max} \rceil$$

where u_l and u_u are consecutively the lower bound and upper bound for the control and $\lfloor z \rfloor$ means rounding the elements of z to the nearest integer less than or equal to z, $\lceil z \rceil$ rounding the elements of z to the nearest integer greater than or equal to z and c some given positive integer. c is used to make the region larger so as to improve stability.

We repeat the above steps, i.e. discretization of the computation of the HJB equation, forward integration of optimal trajectory and region reduction until iteration has nearly

reached convergence. For that purpose, we may prescribe a lower bound for space interval shrinkage factor in percentage, i.e. the ratio of latter space interval length to former. The smaller the shrinkage factor is, the larger the reduction of the space interval length for the next iteration. The shrinkage factor close to 1 indicates that the length of space interval for the next iteration does not change much. By setting a lower bound for space interval shrinkage factor high, for instance 95%, it will ensure that the asymptotic range requirement for applying Richardson Extrapolation is satisfied. Afterwards, we run additional iteration with Completed Richardson Extrapolation as described in the next subsection on the region reduction from the last iteration. This improves the result accuracy from first-order to second-order.

3 Completed Richardson Extrapolation

The Completed Richardson Extrapolation proposed by Roache and Knupp in [25] is an extension of the original Richardson Extrapolation. They completed the method by giving higher-order solution not only on the coarse grid but on the entire fine grid. In particular, they presented application of the extrapolation on numerical solution of time-independent partial differential equations as examples. Furthermore, Richards [24] modified it in order to be used on time-dependent partial differential equation problems.

In short, the formulas for Completed Richardson Extrapolation are as follows. Let $\varphi_{c,i}$ and $\varphi_{f,j}$ denote respectively the first order approximate solution at node i on the coarse and j on the fine grid. The fine grid here is formed by bisecting the coarse grid such that the fine grid coincides with the coarse grid only when the indices are odd ($j = 2i - 1$) where $i = 1, 2, \ldots, N + 1$. Then the extrapolated second order approximate solution by the Completed Richardson Extrapolation, $\varphi_{RE,j}$, is determined by

$$\varphi_{RE,j} = 2\varphi_{f,j} - \varphi_{c,i} \text{ for } j = 2i - 1, \tag{3.1}$$

$$\varphi_{RE,j+1} = \varphi_{f,j+1} + 0.5(\varphi_{RE,j} - \varphi_{f,j} + \varphi_{RE,j+2} - \varphi_{f,j+2}) \text{ for } j + 1 \text{ even} \tag{3.2}$$

From the last iteration, we choose M as the number of coarse grid points so that

$$M_c = M, \quad \Delta x_c = \frac{b-a}{M_c}, \quad N_c > \frac{\Delta x_c \|f\|_\infty + 2\varepsilon_c}{(\Delta x_c)^2}, \quad \Delta t_c = -\frac{1}{N_c},$$

where ε_c is a chosen small positive number. Analoguously, we choose Δx_f and Δt_f so that $M_f = 2M_c$ and $N_f = 2N_c$ and $\varepsilon_f = \varepsilon_c$ for fine grids.

Remark 3.1. At this stage, an appropriate shifting in the spatial discretization might be necessary to include the starting point x_0.

Theorem 3.2. *The conditions $M_f = 2M_c, N_f = 2N_c$ and $\varepsilon_f = \varepsilon_c/2$ fulfill the stability condition in (2.5).*

Proof.

$$\begin{aligned} N_f = 2N_c &> 2\left[\frac{\Delta x_c \|f\|_\infty + 2\varepsilon_c}{(\Delta x_c)^2}\right] \\ &= 2\left[\frac{2\Delta x_f \|f\|_\infty + 4\varepsilon_f}{4(\Delta x_f)^2}\right] = \left[\frac{\Delta x_f \|f\|_\infty + 2\varepsilon_f}{(\Delta x_f)^2}\right]. \end{aligned} \tag{3.3}$$

□

Then, we run an extra iteration as before and update the value function (V_{RE}) and control (u_{RE}) for the fine grids according to (3.1) and (3.2). The resultant pair of matrices (V_{RE}, u_{RE}) is the solution of HJB equation which has a second order of accuracy.

Remark 3.3. We also comment that although it is required in Theorem 3.2 that $\varepsilon_f = \varepsilon_c/2$, in practice, the perturbation parameter ε is usually chosen to be much smaller than any mesh sizes. Therefore, in computations, we simple choose $\varepsilon_f = \varepsilon_c$ and the sufficient condition (3.3) still holds true.

4 Numerical Experiments

To test the effectiveness of this algorithm, we take two examples ranging from simple to more complex. The first example contains 1 state, 1 control and 1 mixed (state-control) inequality constraint from [37] and the second example from [31] has 2 states, 1 control and 1 purely state inequality constraint.

Example 4.1. The problem is to minimize

$$\min\left\{\int_0^1 (x^2+u^2-2u)\,dt + \tfrac{1}{2}(x(1))^2\right\}$$

subject to

$$\dot{x} = u \qquad\qquad x(0) = 0$$
$$g(x,u,t) = -(x^2+u^2-t^2-1) \le 0$$

The analytic optimal solution for this open-loop problem is $x^*(t) = t$ and $u^*(t) = 1$, so that the constraint is active for all $t \in [0,1]$. The value function for this solution is $-\frac{1}{6} \approx -0.166666667$.

The corresponding HJB initial-value problem for this example is

$$\begin{aligned} V_t + \min(uV_x + (x^2+u^2-2u) + rg_\rho(x,u,t)) &= 0 \\ V(1,x) &= \tfrac{1}{2}(x(1))^2 \end{aligned}$$

The numerical simulation is done using MATLAB R2010A and MATLAB Optimization Toolbox. We start with region $-1 \le x \le 2$ and $-2 \le u \le 2$ for the first iteration and then reduce it progressively according to our proposed method. The problem has been resolved for $\varepsilon = 10^{-10}$ and various values of M. The first four iterations are purely computed with Iterative Upwind Finite-Difference Method while the last iteration for $M = 256$ is the result of implementation of the Completed Richardson Extrapolation. The summary of our computations are given in Table 1.

The first, second and third columns are respectively the number of iterations, the number of spatial and time partitions. The penalty value and objective function value along the optimal trajectory for each iteration are shown in fourth and fifth columns. These values are evaluated by forward integration of corresponding terms along the optimal trajectory whereas the value function in sixth column is the value function at the initial point obtained from the Upwind Finite Difference Method. The discrepancy between the objective function

Table 1: Computational result for Example 4.1 ($c = r = 2, \varepsilon = 10^{-10}$).

it.	M	N	pen.	obj.	value	$[a, b]$	$[u_{\mathrm{l}}, u_{\mathrm{u}}]$	%x
1	16	11	0.2799	-0.2472	-0.1039	[-1.000, 2.000]	[-2, 2]	0.56
2	32	38	0.0429	-0.1792	-0.1466	[-0.375, 1.319]	[0, 2]	0.71
3	64	107	0.0039	-0.1646	-0.1570	[-0.106, 1.098]	[0, 2]	0.89
4	128	238	0.0107	-0.1688	-0.1600	[-0.038, 1.039]	[0, 2]	0.96
5	256	497	0.0040	-0.1659	-0.1623	[-0.017, 1.015]	[0, 2]	0.98

value and the value function in each iteration is caused by the use of state and control values of the corresponding closest grid points along the optimal trajectory in the evaluation of the objective function value. However, from iteration to iteration this discrepancy becomes smaller. This indicates that the use of the state and control values of the corresponding closest grid points is a reasonable choice. It can be seen also that in general the penalty and value function decrease as the number of iterations and M increase. Additional information related to the space and control interval used during the iteration are in the seventh and eighth columns. Last column contains the space interval shrinkage factor.

Tables 1, 2 and 3 indicate that the computed optimal control and state along the optimal trajectory converge to the analytic solution as the error decreases significantly.

Table 2: Computed error for u in the maximum and L_2 norm for Example 4.1.

	M				
Error	16	32	64	128	256
$\|\cdot\|_\infty$	0.1406	0.0296	0.0059	0.0068	0.0015
$\|\cdot\|_2$	0.2346	0.0776	0.0332	0.0420	0.0194

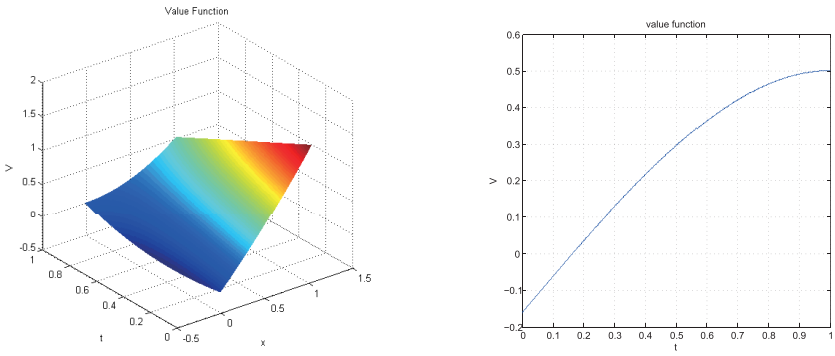
Table 3: Computed error for x in the maximum and L_2 norm for Example 4.1.

	M				
Error	16	32	64	128	256
$\|\cdot\|_\infty$	0.0562	0.0083	0.0013	0.0016	0.0004
$\|\cdot\|_2$	0.1049	0.0250	0.0069	0.0094	0.0072

The computed results for the last iteration are plotted in Figures 1–3. It can be seen that the value function and optimal control shown in Figures 1(a) and 2(a) are smooth in the solution domain. This shows the success of the linear extrapolation used.

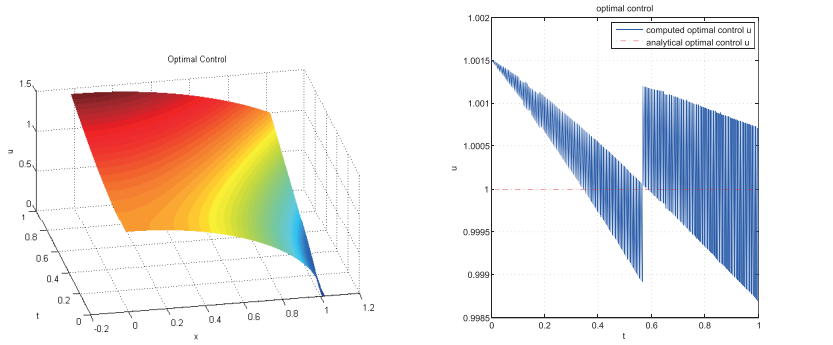
Example 4.2. The second problem is as follows.

$$\min_u J(u) = \int_0^1 (x_1^2 + x_2^2 + 0.005u^2)\, dt$$



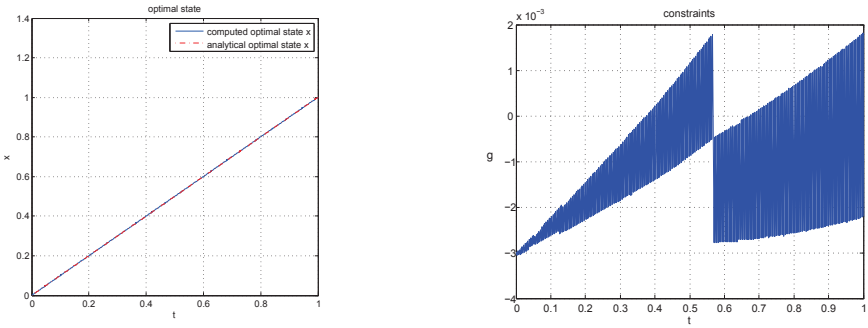
(a) Value function (b) Value function along optimal trajectory

Figure 1: Value functions for Example 4.1.



(a) Optimal control (b) Optimal control along optimal trajectory

Figure 2: Optimal controls for Example 4.1



(a)Optimal state (b) Constraint

Figure 3: Optimal state and constraint along optimal trajectory for Example 4.1

subject to the dynamics

$$\begin{aligned} \dot{x_1} &= x_2, \;\; x_1(0) = 0, \\ \dot{x_2} &= -x_2 + u, \;\; x_2(0) = -1, \end{aligned}$$

and the all-time state inequality constraint

$$h(t,x) = -8(t-0.5)^2 + 0.5 + x_2 \leq 0, \text{ for all } t \in [0,1]$$

Because the constraint is a purely state constraint, it does not give any direct information on how to choose control that satisfies the constraint. For this reason, we need to replace the constraint $h(x,t) \leq 0$ with an equivalent constraint

$$\psi h(x,t) + \varphi h'(x,u,t) \leq 0$$

where $\psi, \varphi > 0$ are two additional control variables, as mentioned in [10]. Here

$$h'(x,u,t) = \frac{dh}{dt} = \frac{\partial h}{\partial x} f(x,u,t) + \frac{\partial h}{\partial t}.$$

Further, to reduce the number of additional controls, we modify this new constraint as follows

$$\psi h(x,t) + (1-\psi) h'(x,u,t) \leq 0,$$

where $0 < \psi < 1$ is a constant close to 1. It is obvious that if $h(x,t_1) = 0$ then $h'(x,u,t_1) \leq 0$ so that $h(x,t) \leq 0$ for $t \in [t_1,t_2], t_2 > t_1$. On the other hand, when $h(x,t_1) \leq 0$, the inequality implies $h(x,t) < 0$ for $t \in [t_1,t_2], t_2 > t_1$ due to the dominance of the first term over the second term. For this numerical example we choose $\psi = 0.9$ and thus the constraint becomes

$$g(t,u,x) = 0.9\ h(t,x) + 0.1\ h'(t,u,x).$$

Therefore, the equivalent problem in HJB form for this problem is

$$\begin{aligned} V_t \quad + \quad & \min_u (x_2 V_{x_1} + (u - x_2) V_{x_2} + (x_1^2 + x_2^2 + 0.005u^2) + r g_\rho(t,x,u)) = 0, \\ & V(1,x) = 0. \end{aligned}$$

For the first iteration, we choose the region

$$-1 \leq x_1 \leq 1, \quad -3 \leq x_2 \leq 1, \quad -20 \leq u \leq 20.$$

For simplicity, we choose the same number of grid points, M, in both x_1 and x_2-directions. The computational results for various M are summarized in Tables 4 and 5 in which the first, second and third columns are respectively the number of iterations and the numbers of spatial and time partitions.

Like in the previous example, the values of the penalty parameter and the objective function along the optimal trajectory for each iteration are listed in the fourth and fifth columns whereas the value function at the initial point is listed in the sixth column of Table 4. We can see that from iteration to iteration, objective function value increases while at the same time the value function and penalty decrease significantly. The difference between the value function and objective function is getting smaller. The lower and upper bounds of the control used in each iteration are in the last column.

Table 5 provides information on the interval reduction for the state variables x_1 and x_2 in each iteration and the last two columns contain the spatial interval shrinkage factors, i.e. the ratio of two consecutive spatial interval sizes. From the table we also see the convergence of our method.

The computed results for $M = 128$ are plotted in Figures 4–6. In particular, Figure 4 contains the cross-sections of the value function and optimal control corresponding to

it.	M	N	pen.	obj.	value	$[u_{\mathrm{l}}, u_{\mathrm{u}}]$
1	8	46	0.2049	0.1763	0.4378	[-20,20]
2	16	92	0.0686	0.1869	0.2595	[-6,14]
3	32	268	0.0320	0.1897	0.2182	[-3,14]
4	64	728	0.0048	0.2021	0.2093	[-3,14]
5	128	1654	0.0055	0.2007	0.2072	[-3,14]

Table 4: Computational result for Example 4.2 ($c = r = 2, \rho = 10^{-2}$).

it.	M	N	$[a_1, b_1]$	$[a_2, b_2]$	$\%x_1$	$\%x_2$
1	8	46	[-1.000, 1.000]	[-3.000, 1.000]	0.60	0.83
2	16	92	[-0.708, 0.500]	[-2.000, 1.306]	0.46	0.61
3	32	268	[-0.402, 0.151]	[-1.413, 0.587]	0.60	0.66
4	64	728	[-0.297, 0.034]	[-1.125, 0.190]	0.91	0.84
5	128	1654	[-0.290, 0.010]	[-1.041, 0.067]	0.96	0.95

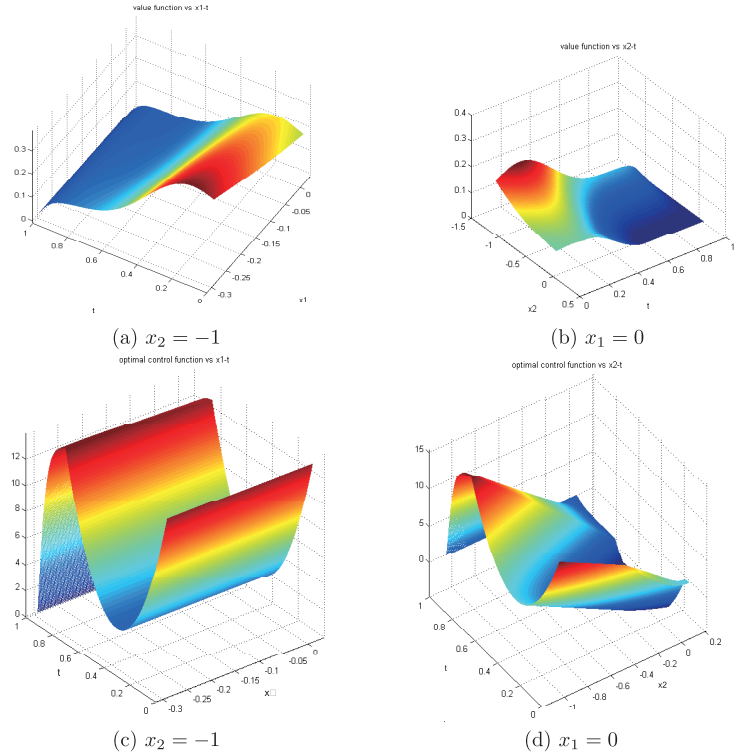
Table 5: Computational result for Example 4.2 ($c = r = 2, \rho = 10^{-2}$).

$x_2 = -1$ and $x_1 = 0$. The value function and optimal control along the optimal trajectory are depicted in Figure 5. Since the analytical solution to this example is unknown, we use the results from the open-loop optimal control solver, MISER 3.3 [15], as our reference solution. The optimal value of the objective function resulted from MISER 3.3 is 0.1736 (see [31]) which is slightly different from our optimal results in Table 4, i.e. 0.2072 and 0.2007 for value and objective functions respectively resulted from forward integration along the open-loop optimal trajectory. We also plot the optimal states from MISER 3.3 in Figure 5 which confirm that optimal solutions from both methods are close to each other. In addition, from Figure 5(d), it is clear that the optimal solution satisfies the constraint, i.e. x_2 is below the quadratic function $8(t - 0.5)^2 - 0.5$ for all t.

Figures 6(a) and 6(b) show the difference between the original constraint $h(t, x)$ and the modified one $g(t, u, x)$, in particular when $0 \leq t < 0.7$. During that period the constraint $h(t, x)$ is inactive, whereas $g(t, u, x)$ is active. However, this does not affect the computation that much as validated by the small value of penalty in Table 4 and Figure 6(c). From Figures 6(b) and 6(c) we see that, when $0 \leq t < 0.7$, the violation of the modified constraint introduces small positive penalty values. On the other hand when $t \geq 0.7$ the modified constraint is inactive and thus the penalty becomes zero.

5 Concluding Remarks

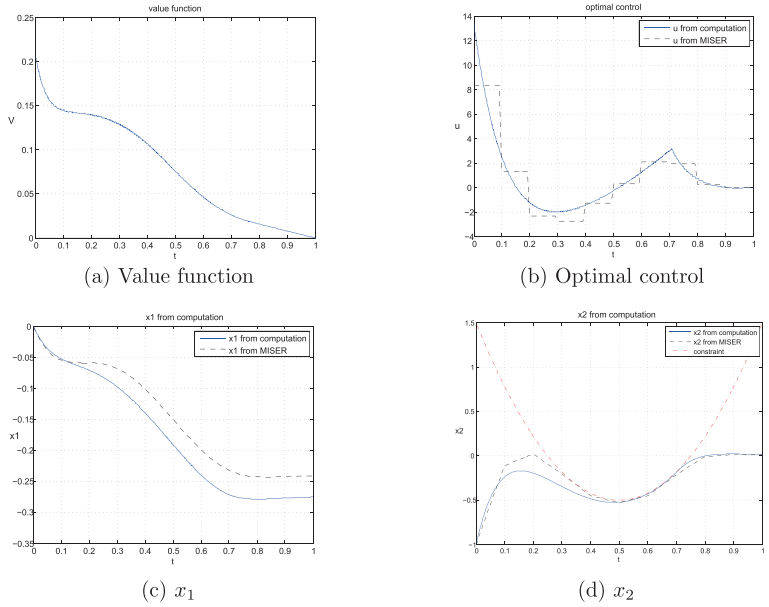
In this work we proposed the Iterative Upwind Finite-Difference Method for the approximation of constrained viscosity solutions to Hamilton-Jacobi-Bellman Equations. Stability of the numerical scheme is analysed. We have also proposed an algorithm for computational domain reduction and the completed Richardson extrapolation technique for increasing the accuracy of approximate solutions. Numerical experimental results have been presented to demonstrate the accuracy and efficiency of the method.



(a) $x_2 = -1$ (b) $x_1 = 0$

(c) $x_2 = -1$ (d) $x_1 = 0$

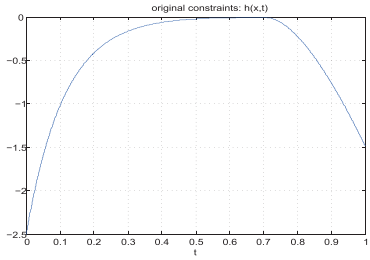
Figure 4: Cross-sections of optimal control for Example 4.2.



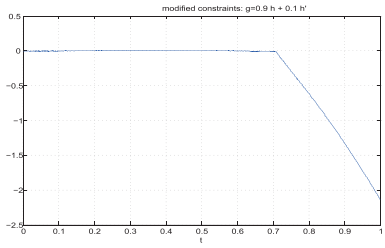
(a) Value function (b) Optimal control

(c) x_1 (d) x_2

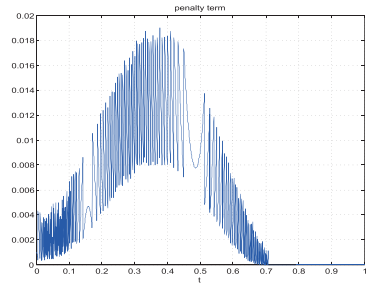
Figure 5: Value function, control and states along optimal trajectory for Example 4.2.



(a) Original constraint



(b) Modified constraint



(c)Penalty

Figure 6: Constraint, modified constraint and penalty along optimal trajectory for Example 4.2.

References

[1] H. Alwardi, S. Wang and L.S. Jennings and S. Richardson, An adaptive least-squares collocation radial basis function method for the HJB equation, *Journal of Global Optimization* 52 (2012) 305–322.

[2] H. Alwardi, S. Wang and L.S. Jennings, An adaptive domain decomposition method for the Hamilton-Jacobi-Bellman equation, *Journal of Global Optimization* 56 (2013) 1361–1373.

[3] M. Bardi and I. Capuzzo-Dolcetta, *Optimal Control and Viscosity Solutions of Hamilton-Jacobi-Bellman Equations*, Birkhauser, Boston-Basel-Berlin, 1997.

[4] O. Bokanowski, N. Forcadel and H. Zidani, Deterministic state-constrained optimal control problems without controllability assumptions, *ESAIM: Control, Optimisation and Calculus of Variations* 17 (2011) 995–1015.

[5] M.G. Crandall, L.C. Evans and P.L. Lions, Some properties of viscosity solutions of Hamilton-Jacobi equations, *Transactions of the American Mathematical Society* 282 (1984) 487–502.

[6] M. G. Crandall and P. L. Lions, Viscosity Solutions of Hamilton-Jacobi Equations, *Transactions of the American Mathematical Society* 277 (1983) 1–42.

[7] W.H. Fleming and H.M. Soner, *Controlled Markov Processes and Viscosity Solutions*, Springer–Verlag, New York, 1993.

[8] B.Z. Guo and B. Sun, A new algorithm for finding numerical solution of optimal feedback control, *IMA Journal of Mathematical Control and Information* 26 (2009) 95–104.

[9] B.Z. Guo and T.T. Wu, Approximation of optimal feedback control: a dynamic programming approach, *Journal of Global Optimization* 46 (2010) 395–422.

[10] R.F. Hartl, Suresh P. Sethi and R. G. Vickson, A Survey of the Maximum Principles for Optimal Control Problems with State Constraints, *SIAM Review* 37 (1995) 181 –218.

[11] C. Hirsch, *Numerical Computation of Internal and External Flows, Vol.2, Computational Methods for Inviscid and Viscous Flows*, John Wiley & Sons, 1990.

[12] C.C. Huang and S. Wang, A power penalty approach to a nonlinear complementarity problem, *Operations Research Letters* 38 (2010) 72–76.

[13] C.S. Huang, S. Wang and K.L. Teo, Solving Hamilton-Jacobi-Bellman Equations by a Modified Method of Characteristics, *Nonlinear Analysis* 40 (2000) 279–293.

[14] C.S. Huang, S. Wang, C.S. Chen and Z.C. Li, A radial basis collocation method for Hamilton-Jacobi-Bellman equations, *Automatica* 42 (2006) 2201–2207.

[15] L.S. Jennings, M.E. Fisher, K.L. Teo and C.J. Goh, *MISER 3 Optimal Control Software: Theory and User Manual. Version* 3, Department of Mathematics, University of Western Australia, WA, 2004.

[16] D.C. Lesmana and S. Wang, An upwind finite difference method for a nonlinear Black-Scholes equation governing European option valuation under transaction cost, *Applied Mathematics & Computation* 219 (2013) 8811–8828.

[17] W. Li and S. Wang, Penalty approach to the HJB equation arising in European stock option pricing with proportional transaction costs, *J. Optim. Theory Appl.* 143 (2009) 279–293.

[18] W. Li and S. Wang, Pricing American options under proportional transaction costs using a penalty approach and a finite difference scheme, *Journal of Industrial and Management Optimization* 9 (2013) 365–389.

[19] P.L. Lions, *Generalised Solutions of Hamilton-Jacobi equation*, Pitman Research Notes in Mathematics, 69, 1982.

[20] R. Luus, *Iterative Dynamic Programming*, Chapman and Hall/CRC Monographs and Surveys in Pure and Applied Mathematics, Boca Raton, 2000.

[21] P. Loreti and M.E. Tessitore, Approximation and regularity results on constrained viscosity solution of Hamilton-Jacobi-Bellman equations *Journal of Mathematical System, Estimation and Control* 4 (1994) 467–483.

[22] J.J.H. Miller, E.O. Riordan, G.I. Shishkin, *Fitted Numerical Methods for Singular Perturbation Problems: Error Estimates in the Maximum Norm for Linear Problems in One and Two Dimensions*, World Scientific, Singapore, 1996.

[23] W.L. Oberkampf and C.J. Roy, *Verification and Validation in Scientific Computing*, Cambridge University Press, Cambridge, 2010.

[24] S.A. Richards, Completed Richardson extrapolation in space and time, *Communications in Numerical Methods in Engineering* 13 (1997) 573–582.

[25] P.J. Roache and P.M. Knupp, Completed Richardson extrapolation, *Communications in Numerical Methods in Engineering* 9 (1993) 365–374.

[26] S. Richardson and S. Wang, Numerical solution of Hamilton-Jacobi-Bellman equations by an exponentially fitted finite volume method, *Optimization* 55 (2006) 121–140.

[27] S. Richardson and S. Wang, The viscosity approximation to the Hamilton-Jacobi-Bellman equation in optimal feedback control: upper bounds for extended domains, *Journal of Industrial and Management Optimization* 6 (2010) 161–175.

[28] H.G. Roos, M. Stynes and L. Tobiska, *Robust Numerical Methods for Singularly Perturbed Differential Equations: Convection-Diffusion-Reaction and Flow Problems*, Springer, 2008.

[29] H.M. Soner, Optimal control with state-space constraint I, *SIAM Journal on Control and Optimization* 24 (1986) 552–561.

[30] J.C. Strikwerda, *Finite Difference Schemes and Partial Differential Equations, Second Edition*, Society of Industrial and Applied Mathematics, Philadelphia, 2004.

[31] K.L. Teo and L.S. Jennings, Nonlinear optimal control problems with continuous state inequality constraints, *Journal of Optimization Theory and Applications* 63 (1989) 1–21.

[32] M.D. Tremblay and R. Luus, Optimization of non-steady-state operation of reactors, *The Canadian Journal of Chemical Engineering* 67 (1989) 494–502.

[33] K.L. Teo, V. Rehbock and L.S. Jennings, A new computational algorithm for functional inequality constrained optimization problems, *Automatica* 29 (1993) 789–792.

[34] S. Wang, A novel exponentially fitted triangular finite element method for an advection-diffusion problem with boundary layers, *Journal of Computational Physics* 134 (1997) 253–260.

[35] S. Wang, F. Gao and K.L. Teo An upwind finite-difference method for the approximation of viscosity solutions to Hamilton-Jacobi-Bellman equations, *IMA Journal of Mathematical Control and Information* 17 (2000) 167–178.

[36] S. Wang, L.S. Jennings and K.L. Teo, Numerical solution of Hamilton-Jacobi-Bellman equations by an upwind finite volume method, *Journal of Global Optimization* 27 (2003) 177–192.

[37] A.Q. Xing, Z.H. Chen, C.L. Wang and Y.Y. Yao, Exact penalty function approach to constrained optimal control problems, *Optimal Control Applications and Methods* 10 (1989) 173–180.

Manuscript received 14 October 2014
revised 19 May 2015
accepted for publication 23 June 2015

HARTONO
Mathematics Department, Sanata Dharma University
Yogyakarta, Indonesia
E-mail address: yghartono@usd.ac.id

L.S. JENNINGS
School of Mathematics & Statistics, The University of Western Australia
Perth, Australia
E-mail address: les.s.jennings@gmail.com

S. WANG
Department of Mathematics & Statistics, Curtin University
GPO Box U1987, Perth WA6845, Australia
E-mail address: song.wang@curtin.edu.au