Adaptive spline interpolation for Hamilton–Jacobi–Bellman equations

Florian Bauer

Mathematisches Institut, Universität Bayreuth, 95440 Bayreuth, Germany

Lars Grüne [*]

Mathematisches Institut, Universität Bayreuth, 95440 Bayreuth, Germany

Willi Semmler

Center for Empirical Macroeconomics, Bielefeld
and New School University, New York

Abstract

We study the performace of adaptive spline interpolation in semi–Lagrangian discretization schemes for Hamilton–Jacobi–Bellman equations. We investigate the local approximation properties of cubic splines on locally refined grids by a theoretical analysis. Numerical examples show how this method performs in practice. Using those examples we also illustrate numerical stability issues.

Key words: Hamilton–Jacobi–Bellman equation; viscosity solution; optimal control; adaptive discretization; spline interpolation

1 Introduction

There are several methods for improving the efficiency of numerical methods for partial differential equations, in particular when dealing with the spatial discretization which is often the most expensive part. In this paper we investigate this problem for a semi–Lagrangian discretization scheme for Hamilton–Jacobi–Bellman equations from optimal control, see [12] for a detailed description of the scheme. This scheme essentially uses the classical

[*] corresponding author, phone +49-921-55-3281, fax +49-921-55-5361
Email address: `lars.gruene@uni-bayreuth.de` (Lars Grüne).

dynamic programming approach in optimal control, and in this context two main advanced space discretization approaches can be identified in the literature: adaptive space discretization techniques [9, 17, 18, 25, 30] which aim at distributing the grid points efficiently and high order interpolation techniques [10, 14, 16, 26, 27, 30] on equidistant grids, which use suitable spaces of high order interpolating functions for the spatial reconstruction.

Motivated by examples from mathematical economics in which adaptive low order approaches show very good results (see [21]), by the fact that in this application area spline interpolation is a popular technique (see, e.g., [10, 24, 27, 26]) and by some promising experimental results in [30] (using, however, different error indicators than in our case, for a comparative discussion see [21, Section 3]), here we thoroughly investigate the interplay between those techniques, i.e., an adaptive discretization technique using cubic splines. On the one hand we provide a theoretical analysis of the local approximation properties of cubic spline functions which, despite the fact that the basic approximation results date back to the 1960s, seemed to be missing up to now. On the other hand, we perform numerical tests for one dimensional examples which show the advantages and limitations of this approach.

Our main results are that (i) for smooth solutions adaptive spline approximations perform very well and can considerably improve the results obtained with both adaptive low order and nonadaptive high order methods while (ii) for nonsmooth solutions (note that we always understand the solutions to the Hamilton–Jacobi–Bellman equation in the viscosity solutions sense, see [2]) numerical instabilities may occur which can seriously affect and even destroy the convergence of the scheme. While these main results seem natural at a first glance given that splines provide a smooth approximation technique, a closer look reveals that neither of the results is trivial: the local approximation properties of splines are difficult to analyze, because splines are nonlocal by construction and the interpolation error does not only depend on the local grid size but on a subtle interconnection of all the grid points, see Theorem 2 and in particular the subsequent discussion. On the other hand, the unsatisfactory performance for nonsmooth functions comes unexpected because it is known that cubic splines provide a convergent interpolation technique even for merely continuous functions, see Remark 2(ii).

Our paper is organized as follows: In Section 2 we pose the problem and describe our basic numerical scheme. In Section 3 we recall some facts about error estimation and adaptive grid construction for Hamilton–Jacobi–Bellman equations while in Section 4 we summarize the basic approximation theory for splines including those aspects of the proofs which we will need for our local approximation analysis, which is performed in Section 5. In Section 6 we describe the effect of these results on our numerical scheme and in Section 7 we show the performance of our scheme by numerical examples. Finally, Section 8 summarizes our findings and concludes the paper.

2 Problem formulation and discretization

In the paper we are concerned with the numerical approximation of viscosity solutions of Hamilton–Jacobi–Bellman equations of the type

$$\min_{u\in U}\{-DV(x)f(x,u)-g(x,u)\}=0. \tag{1}$$

The solution V of this Hamilton–Jacobi–Bellman equation is the optimal value function of the discounted optimal control problem

$$V(x)=\max_{u\in\mathcal{U}}\int_0^\infty e^{-\delta t}g(\varphi(t,x,u),u(t))dt \tag{2}$$

where $\varphi(t,x,u)$ solves

$$\frac{d}{dt}\varphi(t,x,u)=f(\varphi(t,x,u),u(t)),\quad \varphi(0,x,u)=x\in\mathbb{R}^n \tag{3}$$

for measurable control functions from $u\in\mathcal{U}=L^1(\mathbb{R},U)$ with $U\subset\mathbb{R}^m$ compact. We assume that f and g are uniformly Lipschitz in x with Lipschitz constant independent of $u\in U$.

We apply a semi–Lagrangian discretization technique which consists of two steps. In the first step, we discretize (1) in time, which for a time step $h>0$ leads to the discrete Hamilton–Jacobi–Bellman equation

$$V_h(x)=\max_{u\in U}\{hg(x,u)+\beta V_h(\Phi_h(x,u))\}. \tag{4}$$

This idea goes back to Capuzzo Dolcetta [8], where a first order approximation is studied using $\beta=1-\delta h$ and choosing Φ_h to be the Euler approximation of φ, i.e., $\Phi_h(x,u)=x+hf(x,u)$ for constant control value $u\in U$. Higher order approximations are also possible and were studied in [13]. Note that a similar discretization is possible for second order Hamilton–Jacobi–Bellman equations related to stochastic optimal control problems, see [7] for details.

The important feature of this discretization is the relation of (4) to a discrete time optimal control problem, because V_h is the optimal value function of the problem

$$V(x)=\max_{u\in U^{\mathbb{N}}}h\sum_{k=0}^{\infty}\beta^k g(\varphi_h(hk,x,u),u_k) \tag{5}$$

where

$$\varphi_h((k+1)h,x,u)=\Phi_h(\varphi_h(kh,x,u),u_k),\quad \varphi_h(0,x,u)=x\in\mathbb{R}^n. \tag{6}$$

Due to this fact the following second discretization step can also be applied to general discrete time problems of the type (5)–(6), i.e., to discounted optimal control problems based on genuine discrete time models.

Defining the dynamic programming operator T_h by

$$T_h(V_h)(x) = \max_{u \in U} \{hg(x,u) + \beta V_h(\Phi(x,u))\} \tag{7}$$

V_h can be characterized as the unique solution of the fixed point equation

$$V_h(x) = T_h(V_h)(x) \text{ for all } x \in \mathbb{R}^n. \tag{8}$$

The second step of the algorithm, going back to Falcone [11] (see also [12]), now approximates the solution on a suitable finite dimensional function space $\mathcal{W}_\Gamma$ related to a grid Γ with nodes $x_0, \ldots, x_P$, covering a compact subset Ω of the state space. Note that optimal control problems are often defined on a non–compact state space and in this case the reduction to a compact domain Ω is a nontrivial task. Typically, we pick a reasonable set Ω and consider only those trajectories which remain in Ω for all future times. Often it is possible to identify sets Ω which are optimally invariant, i.e., which have the property that any optimal trajectory of the unconstrained problem which starts in Ω remains in Ω for all future times. In this case, the restriction to Ω does not change the optimal value functions, otherwise one has to interpret more carefully the final results. In order to keep the presentation simple, here we assume that for any point $x \in \Omega$ there exists at least one control value u such that $\Phi_h(x,u) \in \Omega$ holds and we assume that V_h is at least continuous.

If we fix a projection $\pi : C(\Omega, \mathbb{R}) \to \mathcal{W}_\Gamma$, then we can now look for an approximation $V_\Gamma \in \mathcal{W}_\Gamma$ of V_h by solving the fixed point equation

$$V_\Gamma(x) = \pi \circ T_h(V_\Gamma)(x). \tag{9}$$

For an overview of algorithms for the solution of this fixed point problem see [18, Section 6.1].

A typical choice of $\mathcal{W}_h$ is the space of continuous and piecewise linear functions if Γ is a simplicid grid and the space of continuous piecewise multilinear functions if Γ is a cuboid grid, where π in both cases is the projection obtained by interpolating the given function at the grid nodes x_i. For a rigorous convergence analysis of this discretization scheme we refer to Bardi and Capuzzo Dolcetta [2] and Falcone and Giorgi [15]. Here we only remark that the scheme converges at least with order γ, where $\gamma > 0$ depends on the regularity properties of V and V_h. In particular, if V and V_h are Lipschitz then we have convergence with order $\gamma = 1/2$ in time and $\gamma = 1$ in space, see [11] or [12], while if V_h is twice continuously differentiable then we have convergence of order $\gamma = 2$ in space, see [28, Theorem 3.5].

In order to make this spatial approximation more efficient, two main approaches have been proposed in the literature. On the one hand, high order approximations have been used, see [10, 14, 16, 26, 30], in which the spaces of (multi–)linear functions are replaced by higher order interpolants. On the other hand, adaptive gridding strategies have been proposed [9, 17, 18, 25, 30], which aim at adaptively constructing a grid Γ in which the grid points are distributed efficiently.

In this paper we will investigate a combination of these ideas using an adaptive grid algorithm together with a cubic spline approximation. In the next sections we will develop the necessary background.

3 Error estimation and adaptive grids

The idea of an adaptive grid generation lies in distributing the nodes of the grid efficiently. Here, in order to identify those regions where a larger number of grid points is needed, we make use of a posteriori error estimation. In order to obtain such error estimates, observe that we want to solve the equation $V_h = T_h(V_h)$ over the space of continuous functions but we do only solve the equation $V_\Gamma = \pi \circ T_h(V_\Gamma)$ on the finite dimensional space $\mathcal{W}_\Gamma$. Therefore, for the estimation of the approximation error we estimate the residual of the dynamic programming operator, i.e., the difference between $V_\Gamma(x)$ and $T_h(V_\Gamma)(x)$ for all $x \in \Omega$. Thus, for each element C_l of the grid Γ we compute local error estimates

$$\eta_l := \max_{x \in C_l} |T_h(V_\Gamma)(x) - V_\Gamma(x)| .$$

It was shown in [17, Theorem 2.2] (see also [18, Theorem 3.2] for the analogous result in a stochastic setting) that the error estimators η_l give upper and lower bounds for the real error. More precisely, for $h < 1/\rho$, i.e., $\beta \in (0,1)$, one can prove the estimates

$$\frac{1}{2} \max_l \eta_l \leq \max_{x \in \Omega} |V_h(x) - V_\Gamma(x)| \leq \frac{1}{1-\beta} \max_l \eta_l. \tag{10}$$

Note that the specific choice of the approximating function space $\mathcal{W}_\Gamma$ does not play a role in this result because it relies on properties of the operator T_h only.

Thus these local error estimates fit into the error estimation theory for PDEs and can be used as indicators for the following local refinement strategy of the grid Γ:

(0) Pick an initial grid Γ_0 and set $i = 0$. Fix a refinement parameter $\theta \in (0,1)$ and a tolerance $rtol > 0$.

(1) Compute the solution V_{Γ_i} on Γ_i.
(2) Evaluate the error estimates η_l. If $\eta_l < rtol$ for all l then stop.
(3) Refine all cells C_j with $\eta_j \geq \theta \max_l \eta_l$, set $i = i + 1$ and go to (1).

As an alternative to specifying *rtol* one could also stop the refinement iteration when a prescribed maximum number of elements is reached, which is what we did in the numerical experiments in this paper. In all these experiments we used $\theta = 0.1$.

It should be noted that both the upper and the lower bound in (10) are necessary for an efficient and reliable adaptation routine: the upper bound ensures that small local errors indeed imply small global errors, which means that refining elements with large local errors — thus reducing the local error — will be the right strategy to reduce the global error (the error estimate is *reliable*). On the other hand, the lower bound ensures that large local errors do indeed indicate that there is something wrong with the numerical solution, which means that we do only refine if there really are large errors (the estimate is *efficient* in the sense that we do not make useless refinements). Note that the gap between the upper and lower bound becomes large if $\beta \approx 1$; in this case numerical experience shows that the upper bound is likely to become too conservative in the sense that the estimated errors typically overestimate the real error. However, since the lower bound is independent of β, even in the case $\beta \approx 1$ — which causes problems in many numerical schemes — the efficiency of the adaptive gridding method is maintained, i.e., we do not make unnecessary refinements.

In practice, the error estimates η_l cannot be evaluated exactly, because it involves maximization over infinitely many points in the element C_l. Instead, we use a number of test points $\tilde{x}_1, \ldots, \tilde{x}_m$ in each element and evaluate $\tilde{\eta}_l = \max \eta(\tilde{x}_i)$. Note that this procedure may underestimate the real error on large elements, thus it is advisable not to choose a too coarse initial grid Γ_0.

This basic adaptation routine can be improved in various ways, e.g., by allowing anisotropic refinement (i.e., refinement in one coordinate direction only) or coarsening (i.e., removing fine elements if they turn out to be unnecessary). We will not elaborate on these aspects here and refer to [18, 20, 21] for details.

4 A short introduction to spline interpolation

The idea of the approach we want to investigate is to use the space of cubic splines as the approximating function space $\mathcal{W}_\Gamma$ in the approximation of the discrete time optimal value function V_h. Before we will address this problem and in particular before we investigate the effects on local grid refinements on spline interpolation we will recall some basic definitions and properties of splines in this section. We start with the basic definition.

Definition 1 *For a given one dimensional grid* $\Gamma = \{x_0, \ldots, x_P\}$ *with* $x_0 < x_1 < \ldots < x_P$ *a function* $s : [x_0, x_P] \to \mathbb{R}$ *is called* spline *of degree* $k \in \mathbb{N}$ *if the following conditions are met:*

- $s \in C^{k-1}[x_0, x_P]$
- *for* $x \in I_i = [x_{i-1}, x_i]$, $i = 1, 2, \ldots, P$, *s equals a polynomial of degree k, i.e., for* $x \in I_i$ *we can write*

$$s(x) = a_{i0} + a_{i1}(x - x_{i-1}) + \ldots + a_{ik}(x - x_{i-1})^k = p_i(x). \tag{11}$$

In our approach we will use splines of degree 3, so called *cubic splines*, hence $\mathcal{W}_\Gamma$ is the space of cubic splines on the interval $\Omega = [x_0, x_P]$. The projection $\pi : C(\Omega, \mathbb{R}) \to \mathcal{W}_\Gamma$ is the interpolation of a continuous function $F : \Omega \to \mathcal{W}_\Gamma$ by the spline s, i.e., $\pi(F) = s$ with s being a cubic spline satisfying

$$s(x_i) = F(x_i),\ i = 0, \ldots, P.$$

This cubic interpolation spline with given values $F(x_i)$ at the nodes x_i for $i = 0, \ldots, P$ is not unique. Therefore we need boundary conditions.

Lemma 1 *Let s be a cubic interpolation spline to given values* $(x_i, F(x_i))$, $i = 0, \ldots, P$. *Then s is unique if it fulfills one of the following boundary conditions (BCs):*

(i) $s''(x_0) = s''(x_P) = 0$ (*natural* BCs)

(ii) $s'(x_0) = s'(x_P)$ *and* $s''(x_0) = s''(x_P)$ (*periodic* BCs)

(iii) $s'(x_0) = F'(x_0)$ *and* $s'(x_P) = F'(x_P)$ (*Hermite* BCs)

Note that the theoretical study of splines usually demands Hermite boundary conditions, which we will assume in all the following results. In practice, however, the information on the derivatives in the interval endpoints will not be available which is why we used natural boundary conditions in our numerical experiments which coincides with the spline routines offered by MATLAB which we used for our implementation.

In the next section we want to investigate the effect of adaptive grid refinement on the interpolation error using cubic interpolation splines. As a prerequisite, in this section we recall a classical theorem due to Hall [22] and revisit the necessary parts of its proof.

Theorem 1 *Let s be a cubic interpolation spline with Hermite BCs interpolating a function* $F \in C^4(\Omega, \mathbb{R})$ *on* $\Gamma = \{x_0, \ldots, x_P\}$ *with* $x_0 < \ldots < x_P$ *and*

$\Omega = [x_0, x_P]$. *Then*

$$\|s - F\|_\infty \leq C\|F^{(4)}\|_\infty \Delta_{max}^4$$

with $C = \frac{5}{384}$, $\Delta_{max} = \max_{i=1,\dots,P}\{x_i - x_{i-1}\}$ *and* $\|G\|_\infty := \sup_{x\in[x_0,x_P]} |G(x)|$ *for* $G \in C(\Omega, \mathbb{R})$.

The proof of Theorem 1 requires three preparatory lemmas. Since the respective proofs are quite lengthy we will omit most of them, except for the proof of Lemma 2 which we will need later on. The complete proofs can be found in [22]. In order to simplify the notation we write

$$s_i := s(x_i),\ s_i' := s'(x_i),\ F_i := F(x_i),\ F_i' := F'(x_i),\ \Delta_i := x_i - x_{i-1}.$$

First, we estimate the difference of the first derivative of F and the one of s at the grid nodes.

Lemma 2 *Let* F*,* s *and* Γ *be from Theorem 1. Then*

$$|s_i' - F_i'| \leq \frac{1}{24}\|F^{(4)}\|_\infty \Delta_{max}^3, \quad i = 0, \dots, P.$$

Proof. The cubic spline s can be written as

$$\begin{aligned} s(x) &= s_{i-1} + s_{i-1}'(x - x_{i-1}) \\ &\quad + \left[3\frac{s_i - s_{i-1}}{\Delta_i^2} - \frac{s_i' + 2s_{i-1}'}{\Delta_i}\right](x - x_{i-1})^2 \\ &\quad + \left[-2\frac{s_i - s_{i-1}}{\Delta_i^3} + \frac{s_i' + s_{i-1}'}{\Delta_i^2}\right](x - x_{i-1})^3. \end{aligned}$$

for $x \in [x_{i-1}, x_i],\ i = 1, \dots, P$.

Because of the continuity of s and its derivatives, we have the following system of linear equations for $i = 1, \dots, P-1$

$$\Delta_{i+1}s_{i-1}' + 2(\Delta_{i+1} + \Delta_i)s_i' + \Delta_i s_{i+1}' = 3\left[\Delta_{i+1}\frac{s_i - s_{i-1}}{\Delta_i} + \Delta_i\frac{s_{i+1} - s_i}{\Delta_{i+1}}\right] \quad (12)$$

Using the Taylor formula one can show that

$$\begin{aligned} \Delta_{i+1}F_{i-1}' + 2(\Delta_{i+1} + \Delta_i)F_i' + \Delta_i F_{i+1}' &= 3\left[\Delta_{i+1}\frac{F_i - F_{i-1}}{\Delta_i} + \Delta_i\frac{F_{i+1} - F_i}{\Delta_{i+1}}\right] \quad (13) \\ &\quad + \frac{1}{24}F^{(4)}(\xi_i)[\Delta_{i+1}(\Delta_i)^3 + \Delta_i(\Delta_{i+1})^3] \end{aligned}$$

with $i = 1, \dots, P-1$ and $x_{i-1} \leq \xi_i \leq x_{i+1}$.

We have $F_i = s_i$ for $i = 0, \dots, P$. Hence from (12) and (13) we obtain

$$\begin{aligned} &\Delta_{i+1}(s'_{i-1} - F'_{i-1}) + 2(\Delta_{i+1} + \Delta_i)(s'_i - F'_i) + \Delta_i(s'_{i+1} - F'_{i+1}) \\ &= \; -\frac{1}{24} F^{(4)}(\xi_i)[\Delta_{i+1}(\Delta_i)^3 + \Delta_i(\Delta_{i+1})^3] \end{aligned}$$

for $i = 1, \dots, P-1$.

Using the Hermite boundary conditions we can write

$$ME = Z \tag{14}$$

where $[E]_i = s_i - F_i$, $[Z]_i = \frac{-1}{24} F^{(4)}(\xi_i)[\Delta_{i+1}(\Delta_i^3) + \Delta_i(\Delta_{i+1})^3]$ for $i = 1, \dots, P-1$ and

$$M = \begin{pmatrix} 2(\Delta_2 + \Delta_1) & \Delta_1 & 0 & \cdots & 0 & 0 \\ \Delta_3 & 2(\Delta_3 + \Delta_2) & \Delta_2 & \ddots & & 0 \\ 0 & \Delta_4 & \ddots & \ddots & & \vdots \\ \vdots & & & \ddots & \ddots & 0 \\ 0 & & & \ddots & \ddots & \Delta_{P-2} \\ 0 & 0 & \dots & 0 & \Delta_P & 2(\Delta_P + \Delta_{P-1}) \end{pmatrix} \in \mathbb{R}^{P-1} \times \mathbb{R}^{P-1}.$$

Now, multiplying both sides with the diagonal matrix $D \in \mathbb{R}^{P-1} \times \mathbb{R}^{P-1}$ with diagonal elements

$$[D]_{ii} = \frac{1}{2(\Delta_{i+1} + \Delta_i)}$$

we get $DME = DZ$. Hence, the elements of the vector E are bounded by

$$\|E\|_\infty = \|(DM)^{-1} DZ\|_\infty \leq \|(DM)^{-1}\|_\infty \cdot \|DZ\|_\infty. \tag{15}$$

Estimating the right side of the equation, we get

$$\|E\|_\infty \leq \frac{1}{24} \|F^{(4)}\|_\infty \Delta_{max}^3,$$

i.e., for $i = 1, \dots, P-1$ we obtain

$$\|s'_i - F'_i\|_\infty \leq \frac{1}{24} \|F^{(4)}\|_\infty \Delta_{max}^3.$$

For $i = 0$ and $i = P$ the Hermite boundary conditions imply $s'_i = F'_i$ which finishes the proof. □

The next lemma gives an upper bound for the difference between F and the unique piecewise cubic polynomial $q \in C^1[x_0, x_P]$ with

$$q(x_i) = F(x_i) \quad \text{and} \quad q'(x_i) = F'(x_i) \quad for\ i = 0, \dots, P. \tag{16}$$

Lemma 3 *For $F \in C^4(\Omega, \mathbb{R})$ and q from (16) we have*

$$\|q - F\|_\infty \leq \frac{1}{384} \|F^{(4)}\|_\infty \Delta_{max}^4.$$

On each interval $I_i = [x_{i-1}, x_i]$ we obtain

$$\max_{x \in I_i} |q(x) - F(x)| \leq \frac{1}{384} \|F^{(4)}\|_\infty \Delta_i^4.$$

The proof can be found in [4].

The third preparatory lemma is needed to evaluate an upper bound for the difference between q and the spline s interpolating F.

Lemma 4 *Let s be the cubic interpolation spline interpolating F on Γ and let q be defined by (16). Then*

$$\|s - q\|_\infty \leq \frac{1}{96} \|F^{(4)}\|_\infty \Delta_{max}^4$$

and in $I_i := [x_{i-1}, x_i]$ we obtain

$$\max_{x \in I_i} |s(x) - q(x)| \leq \frac{1}{96} \|F^{(4)}\|_\infty \Delta_{max}^3 \Delta_i.$$

Again, the proof can be found in [22]. Here the bound $\frac{1}{96}\|F^{(4)}\|_\infty \Delta_{max}^3 \Delta_i$ is obtained from the bound $\frac{1}{24}\|F^{(4)}\|_\infty \Delta_{max}^3$ from Lemma 2 multiplied by the additional factor $\frac{\Delta_i}{4}$.

With these results one can easily prove Theorem 1.

Proof of Theorem 1: The triangle inequality and the estimates from Lemma 3 and 4 yield

$$\begin{aligned} \|s - F\|_\infty = \|q - F - (q - s)\|_\infty &\leq \|q - F\|_\infty + \|q - s\|_\infty \\ \leq \frac{1}{384}\|F^{(4)}\|_\infty \Delta_{max}^4 + \frac{1}{96}\|F^{(4)}\|_\infty \Delta_{max}^4 &= \frac{5}{384}\|F^{(4)}\|_\infty \Delta_{max}^4, \end{aligned}$$

i.e., the assertion of the theorem. □

Remark 1 *In [23] it is proved that the constant $\frac{5}{384}$ in Theorem 1 is optimal.*

5 Spline interpolation on adaptive grids

In this section we investigate the effect of a local grid refinement on the interpolation error. In general, when performing a local refinement, the global bound $\frac{5}{384}\|F^{(4)}\|_\infty \Delta_{max}^4$ will not change, because the maximum length Δ_{max} of the intervals will not change either. Since the spline approximation procedure is not a local interpolation technique (due to the coupling of the local polynomials at the interval boundaries), we have to investigate the reduction of the local error bound directly in the refined areas. The following theorem gives an estimate for the local interpolation error.

Theorem 2 *Let s be the cubic interpolation spline interpolating a function F on the grid Γ with maximum length of the intervals Δ_{max}. Then, on each interval $I_i = [x_{i-1}, x_i]$ of length $\Delta_i = x_i - x_{i-1}$ the interpolation error between F and s satisfies*

$$\max_{x \in I_i} \|s(x) - F(x)\| \leq \frac{1}{384}\|F^{(4)}\|_\infty \Delta_i (\Delta_i^3 + 4\Delta_{max}^3).$$

Proof. The proof follows from revisiting the lemmas used in the proof of Theorem 1:

In Lemma 3 a bound is given for the difference of F and the polynomial q, which approximates the function values and the first derivatives of F exactly. In $I_i = [x_{i-1}, x_i]$, this difference satisfies

$$\|F - q\|_\infty \leq \frac{1}{384}\|F^{(4)}\|_\infty \Delta_i^4.$$

Lemma 4 gives a bound for the difference between q and s in $I_i = [x_{i-1}, x_i]$ of the form

$$\|s - q\|_\infty \leq \frac{1}{96}\|F^{(4)}\|_\infty \Delta_{max}^3 \Delta_i.$$

Combining these two estimates using the triangle inequality as in the proof of Theorem 1 yields the assertion. □

It is illuminating to investigate explicitly the consequence of Theorem 2 to successively refined intervals. To this end we consider a uniform initial grid with $\Delta_i \equiv \Delta_{max}$ and pick one subinterval which is subdivided several times by inserting a new node in its midpoint, i.e., such that its length $\Delta_{min} = \Delta_i^k$ after the k–th subdivision satisfies $\Delta_i^k = \Delta_i^{k-1}/2$. Then in each refinement step the bound in Theorem 2 on this interval will reduce by the factor

$$\alpha_k := \frac{1}{16}\,\frac{\Delta_{min}^3 + 32\Delta_{max}^3}{\Delta_{min}^3 + 4\Delta_{max}^3}. \tag{17}$$

Using $\Delta_{max} = 0.4$, we obtain the following bounds for the reduction factor α_k and the order of convergence p measured by

$$\left(\frac{\Delta_i^{k+1}}{\Delta_i^k}\right)^p = \alpha_k. \tag{18}$$

k	Δ_{min}	reduction factor α_k	order p
1	0.2	0.4832	1.0494
2	0.1	0.4978	1.0063
3	0.05	0.4997	1.0008
4	0.025	0.5000	1.0001

Table 1
Reduction of the theoretical error bound for local refinement

It is easily seen, that for small Δ_{min} the order of convergence of the bound from Theorem 2 is only linear. This linear order of convergence for locally refined grids obtained in Theorem 2 is due to the factor Δ_{max}^3 which remains constant during the local refinement. An inspection of the proof shows that this factor origins from the estimate

$$|s_i' - F_i'| \leq \frac{1}{24}\|F^{(4)}\|_\infty \Delta_{max}^3$$

obtained from Lemma 2. It turns out that this can be a conservative bound and that a local refinement may also have a decreasing effect on this bound, as the following example shows.

In the proof of Lemma 2 the elements $E_i = s'(x_i) - F'(x_i)$ of the error vector $E \in \mathbb{R}^{P-1}$ are obtained by solving the system of linear equations

$$\begin{pmatrix} 2(\Delta_2+\Delta_1) & \Delta_1 & 0 & \cdots & 0 & 0 \\ \Delta_3 & 2(\Delta_3+\Delta_2) & \Delta_2 & \ddots & & 0 \\ 0 & \Delta_4 & \ddots & \ddots & & \vdots \\ \vdots & & & \ddots & \ddots & 0 \\ 0 & & & \ddots & \ddots & \Delta_{P-2} \\ 0 & 0 & \dots & 0 & \Delta_P & 2(\Delta_P+\Delta_{P-1}) \end{pmatrix} E = Z$$

with $[Z]_i = \frac{-1}{24}F^{(4)}(\xi_i)[\Delta_{i+1}(\Delta_i^3) + \Delta_i(\Delta_{i+1})^3]$ for $i = 1, \ldots, P-1$ and $\xi_i \in [x_{i-1}, x_{i+1}]$.

To simplify the computations we consider a very small equidistant grid consisting of seven equidistant nodes, i.e., with constant interval lengths $\Delta_i = h,\ i = 1, \ldots, 6$. For these data we obtain

$$\begin{pmatrix} 4h & h & 0 & 0 & 0 \\ h & 4h & h & 0 & 0 \\ 0 & h & 4h & h & 0 \\ 0 & 0 & h & 4h & h \\ 0 & 0 & 0 & h & 4h \end{pmatrix} E = Z \tag{19}$$

with $[Z]_i = \frac{-1}{24}F^{(4)}(\xi_i)2h^4\ ,\ i = 1, \ldots, 5$.

Solving this system of equations (e.g. using MAPLE) yields

$$E_1 = \frac{-11}{624}\tilde{F}h^3,\ E_2 = \frac{-1}{78}\tilde{F}h^3,\ E_3 = \frac{-3}{208}\tilde{F}h^3,\ E_4 = \frac{-1}{78}\tilde{F}h^3,\ E_5 = \frac{-11}{624}\tilde{F}h^3,$$

where $\tilde{F} = \|F^{(4)}\|_\infty$. Hence, in particular, on I_4 we obtain

$$|E_4| = 0.01282\tilde{F}h^3.$$

Now we proceed as in the computations for Table 1, i.e., we refine the interval I_4 by inserting an additional node at its midpoint, thus obtaining two new intervals $\hat{I}_4$ and $\hat{I}_5$ of length $\hat{h}_4 = \hat{h}_5 = \frac{h}{2}$. Then the system of linear equations above changes to

$$\begin{pmatrix} 4h & h & 0 & 0 & 0 & 0 \\ h & 4h & h & 0 & 0 & 0 \\ 0 & \frac{1}{2}h & 3h & h & 0 & 0 \\ 0 & 0 & \frac{1}{2}h & 2h & \frac{1}{2}h & 0 \\ 0 & 0 & 0 & h & 3h & \frac{1}{2}h \\ 0 & 0 & 0 & 0 & h & 4h \end{pmatrix} \hat{E} = \hat{Z}$$

with

$$\left[\hat{Z}\right]_i = \frac{-1}{24}F^{(4)}(\xi_i)2h^4$$

for $i = 1, 2, 6$ and

$$
\begin{aligned}
\left[\hat{Z}\right]_3 &= \frac{-1}{24}F^{(4)}(\xi_i)\left[\frac{h}{2}h^3 + h\left(\frac{h}{2}\right)^3\right] &&= \frac{-1}{24}F^{(4)}(\xi_i)\frac{5}{8}h^4,\\
\left[\hat{Z}\right]_4 &= \frac{-1}{24}F^{(4)}(\xi_i)2\left(\frac{h}{2}\right)^4 &&= \frac{-1}{24}F^{(4)}(\xi_i)\frac{1}{8}h^4,\\
\left[\hat{Z}\right]_5 &= \frac{-1}{24}F^{(4)}(\xi_i)\left[h\left(\frac{h}{2}\right)^3 + \frac{h}{2}h^3\right] &&= \frac{-1}{24}F^{(4)}(\xi_i)\frac{5}{8}h^4,
\end{aligned}
$$

always with $\xi_i \in [x_{i-1}, x_{i+1}]$. This system has the solutions

$$
\begin{aligned}
\hat{E}_1 &= \frac{-5359}{313632}\tilde{F}h^3, \quad \hat{E}_2 = \frac{-1175}{78408}\tilde{F}h^3, \quad \hat{E}_3 = \frac{-659}{104544}\tilde{F}h^3,\\
\hat{E}_4 &= \frac{-227}{627264}\tilde{F}h^3, \quad \hat{E}_5 = \frac{-109}{19602}\tilde{F}h^3, \quad \hat{E}_6 = \frac{-3049}{156816}\tilde{F}h^3.
\end{aligned}
$$

One sees that the difference between s' and F' is considerably reduced in the refined area, i.e., on $\hat{I}_4 \cup \hat{I}_5 = I_4$ we obtain

$$\max\{|\hat{E}_4|, |\hat{E}_5|\} = 0.00556\tilde{F}h^3 \approx 0.43\,|E_4|.$$

Unfortunately, since this computation and in particular a similar study of the error terms E_i on a grid with seven non–equidistant nodes with varying interval lengths $\Delta_1, \dots, \Delta_6$ as performed in [3, Appendix A.1] yield rather complicated expressions, the derivation of a general estimate for the E_i terms has not been met with success. Still, these case studies indicate that the factor $\frac{1}{24}\|F^{(4)}\|_\infty \Delta_{max}^3$ in Theorem 2 is too pessimistic on locally refined grids and that one may expect a higher convergence order for the local approximation error than the order 1 predicted by Theorem 2. In fact, the local approximation order may be $p \approx 2$ as the following example shows, where we interpolate the function

$$F(x) = \frac{1}{1+x^2}\,, \quad x \in [-4, 4],$$

known as *Runge function.*

We start with an equidistant grid consisting of the nodes $\{x_0, x_1, \dots, x_{20}\} = \{-4, -3.6, \dots, 4\}$. Taking 800 test points we computed the maximal interpolation error 0.0017 on $[-4, 4]$ which is attained on the interval $[-0.8, -0.4]$. Because of the symmetry of F, the same error appears in $[0.4, 0.8]$. However, without loss of generality we will focus on the region $[-4, 0]$. We refine the interval which includes the point with the maximum interpolation error obtaining a new node at $x = -0.6$. Then the maximum interpolation error on $[-0.8, -0.4]$ reduces to $1.994 \cdot 10^{-4}$ attained in the interval $[-0.6, -0.4]$. Refining this one, the new node $x = -0.5$ arises. By analyzing the new intervals the maximum error decreases to $3.988 \cdot 10^{-5}$ and is attained in the interval

$[-0.5, -0.4]$. Continuing this procedure we obtain the following results:

k	Δ_{min}	error e_k	reduction factor α_k	order
0	0.4	$1.70 \cdot 10^{-3}$		
1	0.2	$1.99 \cdot 10^{-4}$	0.117	3.0912
2	0.1	$3.99 \cdot 10^{-5}$	0.201	2.3147
3	0.05	$1.00 \cdot 10^{-5}$	0.251	1.9942
4	0.025	$2.59 \cdot 10^{-6}$	0.259	1.9490

Table 2
Reduction of the experimental error bound for local refinement

Here Δ_{min} is the length of smallest locally refined interval and the indicated error e_k is the local approximation error $|s(x) - F(x)|$ for x from this interval. The reduction factor α_k is given by $\alpha_k = e_k/e_{k-1}$ and the order is computed according to (18). It is readily seen that here an approximation order $p \approx 2$ is achieved.

Of course these results depend on the given problem, the investigated interval and the way of adapting the grids. In our case, only one of the minimum intervals is refined successively. Refining additional surrounding intervals will yield other results.

6 Adaptive spline interpolation for HJB equations

In our numerical approach we combine the adaptive gridding technique described in Section 3 with the spline interpolation from Section 4. To this end we denote by $\mathcal{W}_\Gamma$ the space of cubic spline functions on the grid $\Gamma = \{x_0, \dots, x_P\}$ and choose $\pi : C(\Omega, \mathbb{R}) \to \mathcal{W}_\Gamma$ to be the function that assigns to each function $F : \Omega \to \mathbb{R}$ the interpolating natural cubic spline s.

From (10) we know that the local error estimates η_i will give an estimate for the real error between V_h and its approximation $V_\Gamma \in \mathcal{W}_\Gamma$. The following theorem shows that under an appropriate smoothness condition one can expect that the local error estimate η_l is small if the size of the corresponding grid element is small.

Theorem 3 *Let V_Γ be a solution of the fixed point equation $\pi \circ T_h(V_\Gamma) = V_\Gamma$ on the space $\mathcal{W}_\Gamma$ of cubic splines and assume that $T_h(V_\Gamma) \in C^4(\Omega, \mathbb{R})$ with $\|(T_h(V_\Gamma))^{(4)}\|_\infty \leq C$ for some constant $C > 0$. Then the inequality*

$$\eta_i \leq \frac{C}{384} \Delta_i (\Delta_i^3 + 4\Delta_{max}^3)$$

holds.

Proof. Let $F \in C^4(\Omega, \mathbb{R})$ be an arbitrary function satisfying $\|F^{(4)}\|_\infty \leq C$. Then from Theorem 2 we know that there exists a constant $\widetilde{C}$ such that for each $x \in I_i = [x_{i-1}, x_i]$ the inequality

$$\|\pi(F)(x) - F(x)\| \leq \frac{C}{384}\Delta_i(\Delta_i^3 + 4\Delta_{max}^3)$$

holds. For $F = T(V_\Gamma)$ this implies

$$\begin{aligned} |V_\Gamma(x) - T_h(V_\Gamma)(x)| &= |\pi \circ T_h(V_\Gamma)(x) - T_h(V_\Gamma)(x)| \\ &\leq \frac{C}{384}\Delta_i(\Delta_i^3 + 4\Delta_{max}^3), \end{aligned}$$

i.e., the assertion. □

Remark 2 *(i) As outlined in the discussion after Theorem 2, this bound is essentially linear in Δ_i which in practical examples turns out to be too conservative, i.e., in practice one may expect a quadratic estimate of the type $\widetilde{C}\Delta_i^2$.*

(ii) It is known that even for merely continuous functions $F \in C(\Omega, \mathbb{R})$ the interpolating cubic spline s will converge to F if the size of the grid elements $\Delta_{\max}$ will tend to 0, see, e.g., [1, pp 22f.]. We conjecture that also in this case one may be able to derive a statement similar to Theorem 3, however, as we will see in the numerical examples in the next section, non–differentiability poses other difficulties which is why we did not pursue this topic.

(iii) Note that in Theorem 3 we have only assumed that V_Γ is a solution of the fixed point problem $\pi \circ T_h(V_\Gamma) = V_\Gamma$, we do not impose any uniqueness or stability properties. In the numerical practice, however, in particular the stability of V_Γ is important, because it ensures that the fixed point equation can be solved iteratively. The numerical examples in the next section will show that this property (which can be rigorously shown in the case of linear and multilinear interpolation) may fail to hold in case of nonsmooth functions V_h.

7 Numerical examples

In this section we provide two numerical examples which illustrate the performance of the adaptive spline interpolation method. In both examples the solution to the fixed point equation $V_\Gamma = \pi \circ T_h(V_\Gamma)$ using spline interpolation was computed by the iteration

$$V_\Gamma^{(i+1)} = \pi \circ T_h(V_\Gamma^{(i)}) \tag{20}$$

which was stopped once a prescribed accuracy $\|V_\Gamma^{(i+1)} - V_\Gamma^{(i)}\|_\infty \le \varepsilon$ was reached. In fact, there are more efficient approaches for solving this fixed point problem (see [18, Section 6.1] for an overview for linear interpolation), but since the aim of our study was to investigate the interpolation behavior, this simplest of all possible choices was sufficient for our purpose. Furthermore, using (20) in our numerical examples allows to derive experimental information about the numerical stability of the adaptive spline approach, as in particular our second example will show.

Our first example is a basic economic growth model in discrete time going back to [5] and studied numerically in [28, 29, 21]. It is given by

$$\Phi(x,u) = Ax^\alpha - u(t) \quad \text{and} \quad g(x,u) = \ln u \tag{21}$$

and possesses the analytically computable solution $V(x) = B + C\ln x$ with

$$C = \frac{\alpha}{1-\alpha\beta} \qquad \text{and} \qquad B = \frac{\ln((1-\alpha\beta)A) + \frac{\beta\alpha}{1-\beta\alpha}\ln(\alpha\beta A)}{1-\beta},$$

cf. Figure 1.

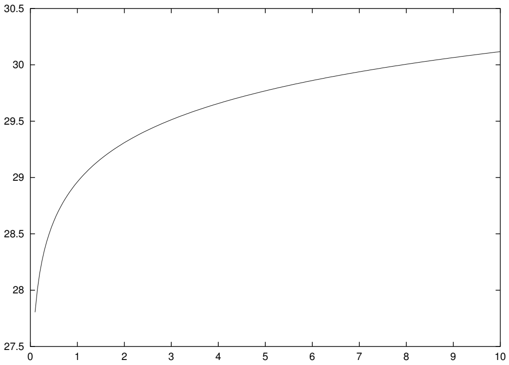

Fig. 1. Solution V_h for Example (21)

For our numerical test we specify $A = 5$, $\alpha = 0.34$ and the discount factor to $\beta = 0.95$ and solve the problem on $\Omega = [0, 1, 10]$. Note that here we have $h = 1$ in (4) because this is a genuine discrete time problem. The iteration (20) was performed until the accuracy $\varepsilon = 10^{-5}$ was reached, then the adaptive refinement procedure with $\theta = 0.1$ was started. The maximization in the operator T_h was performed by discretizing the control value set $U = [0.1, 10]$ with 501 equidistant values and comparing the resulting values of the right hand side of T_h.

Table 3 shows the resulting numerical error for adaptive linear and adaptive spline interpolation and Figure 2 shows the corresponding adaptive grid nodes with the corresponding interval lengths h_i on the y–axis.

#nodes	error $\|V_\Gamma - V_h\|_\infty$	error est. $\max_i \eta_i$
99	$3.3 \cdot 10^{-2}$	$3.0 \cdot 10^{-2}$
103	$1.3 \cdot 10^{-2}$	$1.1 \cdot 10^{-2}$
111	$5.5 \cdot 10^{-3}$	$2.8 \cdot 10^{-3}$
136	$2.9 \cdot 10^{-3}$	$9.2 \cdot 10^{-4}$
185	$6.2 \cdot 10^{-4}$	$2.4 \cdot 10^{-4}$
300	$2.4 \cdot 10^{-4}$	$8.8 \cdot 10^{-5}$
495	$1.9 \cdot 10^{-4}$	$3.8 \cdot 10^{-5}$

#nodes	error $\|V_\Gamma - V_h\|_\infty$	error est. $\max_i \eta_i$
99	$6.3 \cdot 10^{-3}$	$5.3 \cdot 10^{-3}$
101	$1.1 \cdot 10^{-3}$	$9.7 \cdot 10^{-4}$
103	$1.6 \cdot 10^{-4}$	$1.2 \cdot 10^{-4}$
109	$3.8 \cdot 10^{-5}$	$3.0 \cdot 10^{-5}$

Table 3
Accuracy for adaptive linear (left) and adaptive spline interpolation (right)

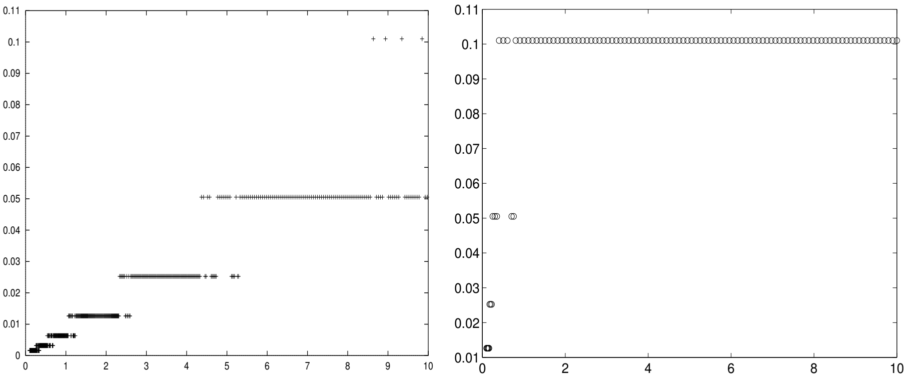

Fig. 2. Grid nodes and width for linear (left) and spline interpolation (right)

One sees that the adaptive spline interpolation provides a more accurate solution with considerably fewer points. In particular, looking at Figure 2 one observes that for the linear interpolation method a considerably large region of the state space is refined while for the spline interpolation method only a small part of the state space is refined in which the solution has large gradient and large curvature. Hence, for this example with smooth solution we can conclude that adaptive spline approximation is considerably more efficient.

Our second example is an ecologic lake management problem introduced in [6] and studied numerically in [19]. It is a continuous time problem given by

$$\dot{x}(t) = u(t) - \mu x(t) + \frac{m x(t)^\rho}{n^\rho + x(t)^\rho} \quad \text{and} \quad g(x,u) = a u^\sigma - k\beta x^2.$$

We specify the parameters $a = 2$, $\sigma = \beta = \frac{1}{2}$, $m = n = 1$, $\rho = 2$, $\mu = 0.55$ and discount rate $\delta = 0.1$. For this parameters the solutions V and V_h of (1) and (4), respectively, have a point of nondifferentiability which corresponds to a

Skiba point for the optimal control problem (see [19] for a detailed discussion and interpretation of this phenomenon).

The numerical parameters were chosen as time step $h = 1/20$ and 50 equidistant control values discretizing $U = [0, 0.4]$, the initial grid had 150 equidistant grid nodes. Using the desired accuracy $\varepsilon = 10^{-5}$ one observes that the iteration (20) does not converge. Figure 3 shows four different approximations $V_\Gamma^{(i)}$ (dashed) starting from $V_\Gamma^{(0)} \equiv 0$ and an accurate solution computed with adaptive linear interpolation and an adaptive grid with 225 nodes (solid).

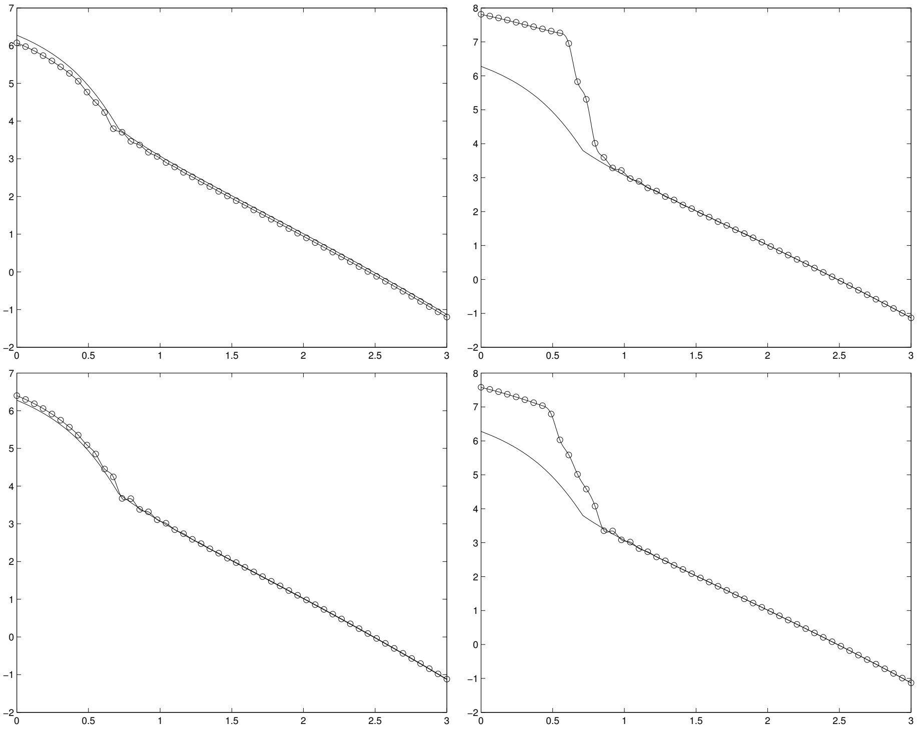

Fig. 3. $V_\Gamma^{(i)}$ from (20) for $i = 500$ (top left), 1000 (top right), 2000 (bottom left), 3000 (bottom right), $\varepsilon = 10^{-5}$

Plotting the value $\|V_\Gamma^{(i+1)} - V_\Gamma^{(i)}\|_\infty$ against i one observes that instead of converging to a fixed point the iteration (20) approaches a periodic orbit, cf. Figure 4 (left).

In this example it turns out that relaxing the termination value from $\varepsilon = 10^{-5}$ to $\varepsilon = 10^{-4}$ — thus starting the adaptation routine at $i = 1090$, i.e., before the periodic orbit is reached — stabilizes the iteration in the sense that after 5 adaptation steps we obtain an accurate numerical solution which is well adapted to the problem. Figure 4 (right) shows the corresponding values of $\|V_{\Gamma_k}^{(i+1)} - V_{\Gamma_k}^{(i)}\|_\infty$, here each grid adaptation step is indicated by a vertical line. Note that after every adaptation the value $\|V_\Gamma^{(i+1)} - V_\Gamma^{(i)}\|_\infty$ is large because the grid changes from i to $i + 1$, in fact, this value represents the local error estimate $\max_i \eta_i$. Figure 5 shows the respective solution and the corresponding

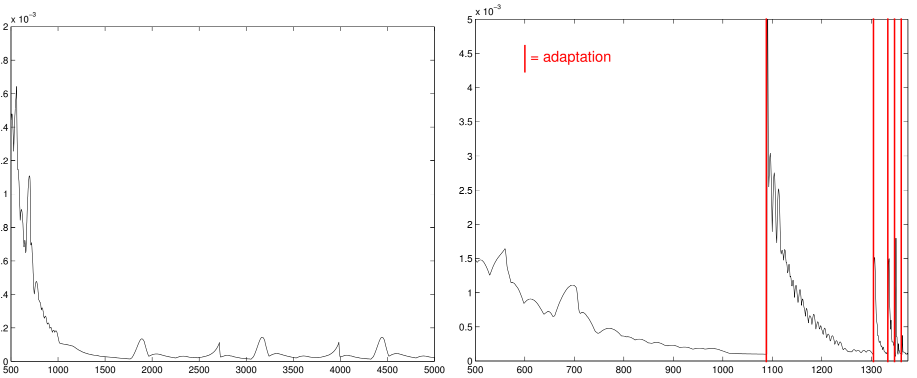

Fig. 4. The value $\|V_\Gamma^{(i+1)} - V_\Gamma^{(i)}\|_\infty$ for $\varepsilon = 10^{-5}$ (left) and $\varepsilon = 10^{-4}$ (right)

adaptive grid.

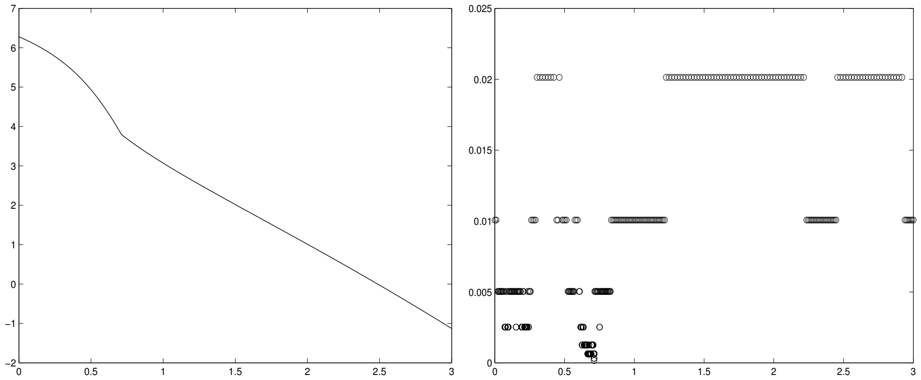

Fig. 5. Adaptive spline solution and grid, $\varepsilon = 10^{-4}$

It should be noted that this stabilizing effect of the adaptive refinement strongly depends on the chosen parameters and also on the fineness of the initial grid and on the initial function $V_\Gamma^{(0)}$. In particular, there exist examples where we could not achieve this stabilizing effect, see [3, Chapter 6].

8 Conclusion

We have investigated adaptive spline approximations in the solution of Hamilton–Jacobi–Bellman equations. The approach turns out to be efficient in case of smooth solutions which was investigated analytically and illustrated by a numerical example. For non–smooth viscosity solutions, numerical instabilities may occur which can destroy the convergence of the scheme. In some cases, a judicious choice of the adaptation parameters can lead to a stabilization of the scheme.

References

[1] J. H. Ahlberg, E. N. Nilson, and J. L. Walsh. *The theory of splines and their applications.* Academic Press, New York, 1967.

[2] M. Bardi and I. Capuzzo Dolcetta. *Optimal Control and Viscosity Solutions of Hamilton-Jacobi-Bellman equations.* Birkhäuser, Boston, 1997.

[3] F. Bauer. Adaptive Spline–Approximation in der optimalen Steuerung (in German). Diploma Thesis, Universität Bayreuth, 2004.

[4] G. Birkhoff and A. Priver. Hermite interpolation error for derivatives. *J. Math. Phys.*, 46:440–447, 1967.

[5] W. A. Brock and L. Mirman. Optimal economic growth and uncertainty: the discounted case. *J. Econ. Theory*, 4:479–513, 1972.

[6] W. A. Brock and D. Starrett. Nonconvexities in ecological management problems. SSRI Working Paper 2026, Department of Economics, University of Wisconsin – Madison, 2000.

[7] F. Camilli and M. Falcone. An approximation scheme for the optimal control of diffusion processes. *RAIRO, Modélisation Math. Anal. Numér.*, 29:97–122, 1995.

[8] I. Capuzzo Dolcetta. On a discrete approximation of the Hamilton-Jacobi equation of dynamic programming. *Appl. Math. Optim.*, 10:367–377, 1983.

[9] B. Cockburn and B. Yenikaya. An adaptive method with rigorous error control for the Hamilton–Jacobi equations. Part I: The one–dimensional steady state case. *Appl. Numer. Math.*, 2005. to appear.

[10] J. W. Daniel. Splines and efficiency in dynamic programming. *J. Math. Anal. Appl.*, 54:402–407, 1976.

[11] M. Falcone. A numerical approach to the infinite horizon problem of deterministic control theory. *Appl. Math. Optim.*, 15:1–13, 1987. *Corrigenda*, ibid., 23 (1991), 213–214.

[12] M. Falcone. Numerical solution of dynamic programming equations. Appendix A in Bardi, M. and Capuzzo Dolcetta, I., Optimal control and viscosity solutions of Hamilton-Jacobi-Bellman equations, Birkhäuser, Boston, 1997.

[13] M. Falcone and R. Ferretti. Discrete time high-order schemes for viscosity solutions of Hamilton-Jacobi-Bellman equations. *Numer. Math.*, 67:315–344, 1994.

[14] M. Falcone, R. Ferretti, and T. Manfroni. Optimal discretization steps in semi-Lagrangian approximation of first-order PDEs. In M. Falcone and Ch. Makridakis, editors, *Numerical methods for viscosity solutions and applications*, volume 59 of *Ser. Adv. Math. Appl. Sci.*, pages 95–117. World Scientific, Singapore, 2001.

[15] M. Falcone and T. Giorgi. An approximation scheme for evolutive Hamilton-Jacobi equations. In W. M. McEneaney, G.G. Yin, and Q. Zhang, editors, *Stochastic analysis, control, optimization and applications*, pages 288–303. Birkhäuser, Boston, 1999.

[16] R. Ferretti. Convergence of semi-Lagrangian approximations to convex Hamilton-Jacobi equations under (very) large Courant numbers. *SIAM J. Numer. Anal.*, 40(6):2240–2253, 2003.

[17] L. Grüne. An adaptive grid scheme for the discrete Hamilton-Jacobi-Bellman equation. *Numer. Math.*, 75(3):319–337, 1997.

[18] L. Grüne. Error estimation and adaptive discretization for the discrete stochastic Hamilton–Jacobi–Bellman equation. *Numer. Math.*, 99:85–112, 2004.

[19] L. Grüne, M. Kato, and W. Semmler. Solving ecological management problems using dynamic programming. *J. Econ. Behavior and Organization*, 2005. to appear.

[20] L. Grüne, M. Metscher, and M. Ohlberger. On numerical algorithm and interactive visualization for optimal control problems. *Comput. Vis. Sci.*, 1(4):221–229, 1999.

[21] L. Grüne and W. Semmler. Using dynamic programming with adaptive grid scheme for optimal control problems in economics. *J. Econ. Dyn. Control*, 28:2427–2456, 2004.

[22] C. A. Hall. On error bounds for spline interpolation. *J. Approx. Theory*, 1:209–218, 1968.

[23] C. A. Hall and W. W. Meyer. Optimal error bounds for cubic spline interpolation. *J. Approx. Theory*, 16:105–122, 1976.

[24] S. A. Johnson, J. R. Stedinger, C. A. Shoemaker, Y. Li, and J. A. Tejada-Guibert. Numerical solution of continuous–state dynamic programs using linear and spline interpolation. *Oper. Res.*, 41(3):484–500, 1993.

[25] R. Munos and A. Moore. Variable resolution discretization in optimal control. *Mach. Learn.*, 2–3:291–323, 2002.

[26] M. Reiter. Solving higher–dimensional continuous–time stochastic control problems by value function regression. *J. Econ. Dyn. Control*, 23:1329–1353, 1999.

[27] J. Rust. Numerical dynamic programming in economics. In H. M. Amman, D. A. Kendrick, and J. Rust, editors, *Handbook of Computational Economics.* Elsevier, Amsterdam, 1996.

[28] M. S. Santos and J. Vigo-Aguiar. Accuracy estimates for a numerical approach to stochastic growth models. Discussion Paper 107, Institute for Empirical Macroeconomics, Federal Reserve Bank of Minneapolis, December 1995.

[29] M. S. Santos and J. Vigo-Aguiar. Analysis of a numerical dynamic programming algorithm applied to economic models. *Econometrica*, 66:409–426, 1998.

[30] M. A. Trick and S. E. Zin. Spline approximations to value functions: a linear programming approach. *Macroecon. Dyn.*, 1:255–277, 1997.