Least-squares Monte Carlo for backward SDEs

Christian Bender[1] and Jessica Steiner[1]

December 21, 2010

Abstract

In this paper we first give a review of the least-squares Monte Carlo approach for approximating the solution of backward stochastic differential equations (BSDEs) first suggested by Gobet, Lemor, and Warin (Ann. Appl. Probab., 15, 2005, 2172–2202). We then propose the use of basis functions, which form a system of martingales, and explain how the least-squares Monte Carlo scheme can be simplified by exploiting the martingale property of the basis functions. We partially compare the convergence behavior of the original scheme and the scheme based on martingale basis functions, and provide several numerical examples related to option pricing problems under different interest rates for borrowing and investing.

Keywords: Backward SDE, numerical approximation, Monte Carlo, option pricing.

AMS classification: 65C30, 65C05, 91G20, 91G60.

1 Introduction

Many pricing and optimization problems in financial mathematics can be reformulated in terms of backward stochastic differential equations (BSDEs), see e.g. the classical survey paper by El Karoui et al. (1997). These equations are non-anticipating terminal value problems for stochastic differential equations of the form

$$dY_t = f(t, Y_t, Z_t)dt + Z_t dW_t, \quad Y_T = \xi.$$

Here, a D-dimensional Brownian motion W, the square-integrable terminal condition ξ (measurable with respect to the filtration generated up to time T by the Brownian motion) and the so-called driver f are given. The solution itself consists of a pair of square-integrable adapted processes (Y, Z), such that the corresponding integral equation is satisfied.

[1]Saarland University, Department of Mathematics, PO Box 151150, D-66041 Saarbrücken, Germany. bender@math.uni-sb.de, steiner@math.uni-sb.de.

Roughly speaking, in many pricing and hedging problems, Y_t corresponds to the option price and Z_t is related to the hedging portfolio. In many portfolio optimization problems, Y_t corresponds to the value process while an optimal control can often be derived from Z_t. Finally, BSDEs can also be applied in order to obtain Feynman-Kac-type representation formulas for nonlinear parabolic PDEs. Here Y_t and Z_t correspond to the solution and the gradient of the PDE, respectively. With these applications in mind, the numerical approximation of BSDEs becomes an important, but challenging problem.

One branch of numerical algorithms for BSDEs exploits the connection to PDEs and basically reduces the numerical approximation of the BSDE to solving the corresponding parabolic PDE numerically, see e.g. Douglas et al. (1996); Milstein and Tretyakov (2006); Ma et al. (2009). The practical applicability of these algorithms may be limited due to high-dimensionality or lack of smoothness of the coefficients. However, for low-dimensional problem with smooth coefficients the PDE methods are hard to beat. Another branch of algorithms, which is the one we discuss in the present paper, deals directly with the stochastic problem. These stochastic algorithms can typically be decomposed into a two-step procedure. The first step consists of a time discretization of the BSDE. The main difficulty here is that, on the one hand, the discretization quite naturally works backwards in time, because the terminal condition is given. On the other hand, the numerical solution should be adapted to the filtration (because the true solution is so). However, the information grows forwards in time. This problem can be solved by projecting the solution on the available information in each step while going backwards in time. While these ideas can be traced back to the papers by Bally (1997) and Chevance (1997), a detailed analysis of the corresponding time discretization scheme under quite general assumptions was first given by Zhang (2004) and Bouchard and Touzi (2004). However, 'projecting on the available information' means that in each time step a conditional expectation must be evaluated. Going backwards step by step one, hence, ends up with a high order nesting of conditional expectations. As the conditional expectation cannot be calculated in closed form, in a second step one has to apply an approximation procedure for the conditional expectations which can be nested without running into explosive computational costs.

In this paper we will focus on the least-squares Monte Carlo approach for estimating the conditional expectations which was made popular in financial mathematics by Longstaff and Schwartz (2001) in the context of Bermudan option pricing. It was first applied to BSDEs and analyzed in this setting by Gobet et al. (2005) and Lemor et al. (2006). The basic idea here is to replace the conditional expectations by projections on finite-dimensional subspaces which are spanned by pre-selected basis functions. The coefficients for the projection on the finite-dimensional subspaces are approximated by the solution of a linear least-squares problem making use of simulated sample paths.

After having discussed the time discretization step and the least-squares Monte Carlo approach, we propose the use of basis functions, which form a system of martingales. A similar idea can be found in Glasserman and Yu (2002) in the context of Bermudan option pricing. For the BSDE case the use of martingale basis functions is motivated by the following observation: Going backwards in time, one actually has to evaluate three conditional expectations per time step. If the approximation of Y at time t_{i+1}, say, is a linear combination of basis functions and these basis functions satisfy appropriate conditions related to the martingale property, then two of the conditional expectations can be calculated in closed form. Only one conditional expectation which involves the nonlinearity of the driver f must still be approximated by least-squares Monte Carlo. Based on this observation we suggest a simplified version of the least-squares Monte Carlo algorithm, when 'martingale basis functions' are at our disposal. An example shows how to construct such basis functions for a multi-dimensional Black-Scholes setting, and we point to possible extensions for more general models. We also analyze the projection error of the new scheme based on martingale basis functions.

Finally, we present a simulation study for the pricing problem of a call spread option under different interest rates for borrowing and investing. Here we compare the original least-squares Monte Carlo scheme with the new scheme, which exploits the use of martingale basis functions. The numerical experiments contain situations with a small and a larger Lipschitz constant of the nonlinearity of the driver and with options on a single stock or on the maximum of several stocks. Overall we find that the use of martingale basis functions improves on the quality of the numerical solutions in our test example and, at the same time, significantly reduces the simulation costs.

The paper is organized as follows: In Section 2 we give a review of the least-squares Monte Carlo scheme for BSDEs. In this section we also refer to various variants concerning the time discretization and the approximation of the conditional expectations which are available in the literature. Section 3 is devoted to the new scheme based on martingale basis functions, while the numerical experiments are discussed in Section 4.

2 Least-squares Monte Carlo for BSDEs

In this section we give a review of the least-squares Monte Carlo approach to BSDEs initiated by Gobet et al. (2005). As it is the case for most of the numerical algorithms for BSDEs, it consists of two steps: a time discretization and a procedure for the approximation of (nested) conditional expectations. We will discuss both steps separately, pointing to alternative ways for designing algorithms to solve BSDEs.

Before we explain the time discretization step we first introduce the

standing assumptions throughout the paper. The aim is to approximate a decoupled forward backward SDE of the form

$$\begin{aligned} dX_t &= b(t, X_t)dt + \sigma(t, X_t)dW_t, \quad X_0 = x_0, \\ dY_t &= f(t, X_t, Y_t, Z_t)dt + Z_t dW_t, \quad Y_T = g(X_T). \end{aligned}$$

Here $W_t = (W_{1,t}, \dots, W_{D,t})^*$, (the star denoting matrix transposition), is a D-dimensional Brownian motion on $[0, T]$ and $Z_t = (Z_{1,t}, \dots, Z_{D,t})$. The process X is $\mathbb{R}^M$-valued and the process Y is $\mathbb{R}$-valued. We assume Lipschitz continuity of the coefficient functions in the following sense:

Assumption 2.1. There is a constant κ such that

$$\begin{aligned} & |b(t,x) - b(t',x')| + |\sigma(t,x) - \sigma(t',x')| \\ & + |f(t,x,y,z) - f(t',x',y',z')| + |g(x) - g(x')| \\ \leq\ & \kappa(\sqrt{|t-t'|} + |x-x'| + |y-y'| + |z-z'|) \end{aligned}$$

for all $(t, x, y, z),\ (t', x', y', z') \in [0, T] \times \mathbb{R}^M \times \mathbb{R} \times \mathbb{R}^D$.

With this assumption we strive for notational simplicity rather than for generality. We emphasize that, for example, path dependent terminal conditions of the form $Y_T = \Phi(X)$, where the functional Φ satisfies some suitable Lipschitz conditions on the path space, can be easily incorporated, see Zhang (2004) or Lemor et al. (2006).

2.1 Time discretization

For the time discretization we consider a partition $\pi = \{t_0, \dots, t_N\}$ of the interval $[0, T]$, i.e. $0 = t_0 < t_1 < t_2 < \cdots < t_N = T$. We suppose that the forward SDE is already discretized in a suitable way by a process $X^\pi_{t_i}$, $t_i \in \pi$, such that

$$\max_{t_i \in \pi} E[|X_{t_i} - X^\pi_{t_i}|^2] \leq C|\pi| \tag{1}$$

for a constant $C \geq 0$, and $(X^\pi_{t_i}, \mathcal{F}_{t_i})_{t_i \in \pi}$ is Markovian. In the numerical examples in Section 4, X is a (multi-dimensional) geometric Brownian motion and can, hence, be sampled perfectly on the grid π. In general situations, one can e.g. apply an Euler scheme on X. We now motivate a natural time discretization for the pair (Y, Z), which works backwards in time. Denoting $\Delta_i = t_{i+1} - t_i$, $\Delta W_{d,i} = W_{d,t_{i+1}} - W_{d,t_i}$, and $\Delta W_i = (\Delta W_{1,i}, \dots, \Delta W_{D,i})^*$ for $t_i \in \pi$, we write

$$Y_{t_i} \approx Y_{t_{i+1}} - f(t_i, X_{t_i}, Y_{t_i}, Z_{t_i})\Delta_i - Z_{t_i}\Delta W_i. \tag{2}$$

Multiplying with a Brownian increment $\Delta W_{d,i}$ for some $d = 1, \dots, D$ and taking conditional expectation yields,

$$\begin{aligned} 0 &= E[\Delta W_{d,i}(Y_{t_i} + f(t_i, X_{t_i}, Y_{t_i}, Z_{t_i})\Delta_i)|\mathcal{F}_{t_i}] \\ &\approx E[\Delta W_{d,i} Y_{t_{i+1}}|\mathcal{F}_{t_i}] - \sum_{l=1}^{D} E[Z_{l,t_i}\Delta W_{l,i+1}\Delta W_{d,i}|\mathcal{F}_{t_i}] \\ &= E[\Delta W_{d,i} Y_{t_{i+1}}|\mathcal{F}_{t_i}] - Z_{d,t_i}\Delta_i. \end{aligned}$$

This suggests that, given $Y_{t_{i+1}}$, Z_{t_i} can be approximated as

$$Z_{t_i} \approx \frac{1}{\Delta_i} E[\Delta W_i^* \, Y_{t_{i+1}}|\mathcal{F}_{t_i}]. \tag{3}$$

In order to obtain an approximation of Y_{t_i}, given $Y_{t_{i+1}}$, we simply take conditional expectation in (2) and get

$$\begin{aligned} Y_{t_i} = E[Y_{t_i}|\mathcal{F}_{t_i}] &\approx E[Y_{t_{i+1}} - f(t_i, X_{t_i}, Y_{t_i}, Z_{t_i})\Delta_i|\mathcal{F}_{t_i}] \\ &\approx E[Y_{t_{i+1}} - f(t_i, X_{t_i}, Y_{t_{i+1}}, Z_{t_i})\Delta_i|\mathcal{F}_{t_i}]. \end{aligned} \tag{4}$$

The last approximation makes the approximation explicit in time. The heuristics in (2)–(4) lead to the time discretization (Y^π, Z^π) for (Y, Z) which was studied by Zhang (2004) and Bouchard and Touzi (2004):

$$\begin{aligned} Y_{t_N}^\pi &= g(X_{t_N}^\pi), \quad Z_{t_N}^\pi = 0, \\ Z_{t_i}^\pi &= \frac{1}{\Delta_i} E[\Delta W_i^* \, Y_{t_{i+1}}^\pi|\mathcal{F}_{t_i}], \quad i = N-1, \dots, 0 \\ Y_{t_i}^\pi &= E[Y_{t_{i+1}}^\pi - f(t_i, X_{t_i}^\pi, Y_{t_{i+1}}^\pi, Z_{t_i}^\pi)\Delta_i|\mathcal{F}_{t_i}], \quad i = N-1, \dots, 0. \end{aligned} \tag{5}$$

The results in Zhang (2004) and Bouchard and Touzi (2004) (see also Lemor et al., 2006) imply that, under Assumption 2.1, the time discretization error in the L^2-sense is of order 1/2, i.e. there is a constant C (independent of π) such that

$$\sup_{0 \le t \le T} E[|Y_t - Y_t^\pi|^2] + \int_0^T E[|Z_t - Z_t^\pi|^2] \le C|\pi|, \tag{6}$$

where (Y_t^π, Z_t^π) is the piecewise constant interpolation of (5). We note that Bally (1997) and Chevance (1997) were the first to study this type of time discretization with a (hardly implementable) random time partition respectively under strong regularity assumptions.

Although the time discretization scheme in (5) is explicit in time, each time step requires the evaluation of conditional expectations, which leads to a high order nesting of conditional expectations. The numerical approximation of nested conditional expectations is a highly demanding problem, in particular when the forward SDE takes values in a high-dimensional state

space. We will discuss some aspects related to this issue in the next subsection.

Before doing so, we give some remarks concerning related results on the the time discretization of BSDEs:

1. The first line of (4) suggests an implicit scheme for the Y-part replacing

$$E[Y^\pi_{t_{i+1}} - f(t_i, X^\pi_{t_i}, Y^\pi_{t_{i+1}}, Z^\pi_{t_i})\Delta_i|\mathcal{F}_{t_i}]$$

in (5) by

$$E[Y^\pi_{t_{i+1}}|\mathcal{F}_{t_i}] - f(t_i, X^\pi_{t_i}, Y^\pi_{t_i}, Z^\pi_{t_i})\Delta_i.$$

Concerning the time discretization error, the convergence of this implicit scheme is also of order 1/2, see Bouchard and Touzi (2004). It requires, however, some iteration procedure to become explicit in time. The iteration can be done in each time step (inner iteration) as in Gobet et al. (2005) or mimicking a Picard iteration (outer iteration) as in Bender and Denk (2007) and Gobet and Labart (2010). Bender and Denk (2007) argue that the outer iteration reduces the error propagation when the conditional expectations are approximated numerically. Gobet and Labart (2010) explain how to obtain efficient control variates for the estimation of the conditional expectations in a Monte Carlo setting via the outer iteration. As an alternative method for reducing the variance, Bender and Moseler (2010) adjust the importance sampling technique to a BSDE setting.

2. When the terminal condition g is less regular than Lipschitz continuous, a time discretization error of order 1/2 can still be achieved in many cases by choosing appropriate, possibly non-equidistant, partitions, see Gobet and Makhlouf (2010). Under stronger smoothness conditions on the coefficient functions b, σ, f, g the error at time 0 $|Y_0 - Y^\pi_0|$ converges to zero at a rate of 1, see Gobet and Labart (2007) who extend a related result by Chevance (1997). For a time discretization scheme of BSDEs with jumps under Lipschitz conditions we refer to Bouchard and Elie (2008). For coupled forward backward SDEs, Bender and Zhang (2008) provide sufficient conditions to obtain a time discretization error of order 1/2 and an iterative procedure for decoupling the equation. The case of second order BSDEs is discussed in Bouchard et al. (2009).

3. Some first results on the time discretization of BSDEs with quadratic growth of the driver f in the z-variable can be found in Imkeller et al. (2010) and Richou (2010). Imkeller et al. (2010) apply a truncation argument and, thus, use an approximation via Lipschitz drivers, while Richou (2010) makes use of (time-dependent) bounds on Z_t. So, from a practical point of view, in both cases the situation is, at best,

comparable with the Lipschitz case with a 'large' Lipschitz constant. However, the constant C in (6) depends exponentially on the Lipschitz constant of f. So, it is no suprise that our numerical results in Section 4 demonstrate that even in the Lipschitz case with a large Lipschitz constant, numerical algorithms may run into problems.

4. For reflected BSDEs a time discretization scheme related to (5) was studied by Ma and Zhang (2005) and Bouchard and Chassagneux (2008). Their results suggests that, in general, this scheme only converges at a rate of 1/4.

2.2 Approximation of conditional expectations

In order to transform the time discretization scheme in (5) into a viable numerical scheme, the conditional expectations must be replaced by an approximation procedure which can be nested several times without running into explosive costs. Different techniques have been suggested in the literature including:

- Approximation of the driving Brownian motion by trees for low-dimensional problems, see Briand et al. (2001) and Ma et al. (2002).
- Cubature methods, see Crisan and Manolarakis (2010), and sparse grids methods, see Gunzburger and Zhang (2010), which rely on some smoothness assumptions.
- Quantization methods, see Bally and Pagès (2003) for reflected BSDEs and Delarue and Menozzi (2006) for coupled FBSDEs.
- Nonparametric kernel estimators and Malliavin Monte Carlo, as discussed by Bouchard and Touzi (2004).
- Least-squares Monte Carlo, which we will now explain in more detail.

The least-squares Monte Carlo method for approximating conditional expectations was made popular in financial mathematics by the Longstaff and Schwartz (2001) algorithm for the pricing of American options. More generally, it can be applied to compute conditional expectations of the form $E[Y|X]$ for square integrable random variables X and Y numerically, provided a machinery for sampling independent copies of the pair (X, Y) is at hand. The method builds upon the elementary property that $E[Y|X] = u(X)$, where the function u solves

$$u = \arg\min_v E[|v(X) - Y|^2]$$

and v runs over all measurable functions with $E[|v(X)|^2] < \infty$. In order to simplify this infinite-dimensional minimization problem, one chooses a

row vector of so-called basis functions $\eta(x) = (\eta_1(x), \dots, \eta_K(x))$, for some $K \in \mathbb{N}$, and considers the K-dimensional minimization problem

$$\alpha^{(K)} = \arg \min_{\alpha \in \mathbb{R}^K} E[|\eta(X)\,\alpha - Y|^2].$$

In a final step the problem can be simplified to a linear least-squares problem. To this end one just replaces the expectation by a sample mean

$$\alpha^{(K,L)} = \arg \min_{\alpha \in \mathbb{R}^K} \frac{1}{L} \sum_{\lambda=1}^{L} |\eta({}_\lambda X)\,\alpha - {}_\lambda Y|^2,$$

where $({}_\lambda X,\ {}_\lambda Y)$, $\lambda = 1, \dots, L$, are independent copies of (X, Y). Given the matrix

$$\mathcal{A}^{(K,L)} = \frac{1}{\sqrt{L}} \left(\eta_k({}_\lambda X)\right)_{\lambda=1,\dots,L, k=1,\dots,K},$$

one has

$$\alpha^{(K,L)} = \left((\mathcal{A}^{(K,L)})^* \mathcal{A}^{(K,L)}\right)^{-1} (\mathcal{A}^{(K,L)})^* \begin{pmatrix} {}_1Y \\ \vdots \\ {}_LY \end{pmatrix}.$$

(Here, one can apply the pseudo-inverse of $\mathcal{A}^{(K,L)}$, if the inverse in the previous expression does not exist). The least-squares Monte Carlo estimator for the conditional expectation $u(x) := E[Y|X = x]$ is then given by

$$u^{(K,L)}(x) := \eta(x)\,\alpha^{(K,L)}.$$

Clearly, this estimation procedure has two error sources, a systematic error induced by the choice of basis functions and a simulation error.

Gobet et al. (2005) first suggested the use of least-squares Monte Carlo for BSDEs and analyzed the different error sources. We now describe the algorithm proposed by Lemor et al. (2006), which combines the explicit time discretization scheme (5) with least-squares Monte Carlo for estimating the conditional expectations. Notice first that, due to the Markovianity of $(X^\pi_{t_i}, \mathcal{F}_{t_i})_{t_i \in \pi}$, the time discretization in (5) can be rewritten as

$$\begin{aligned} Y^\pi_{t_N} &= g(X^\pi_{t_N}), \quad Z^\pi_{t_N} = 0, \\ Z^\pi_{t_i} &= \frac{1}{\Delta_i} E[\Delta W_i^*\, Y^\pi_{t_{i+1}} | X^\pi_{t_i}], \quad i = N-1, \dots, 0 \\ Y^\pi_{t_i} &= E[Y^\pi_{t_{i+1}} - f(t_i, X^\pi_{t_i}, Y^\pi_{t_{i+1}}, Z^\pi_{t_i})\Delta_i | X^\pi_{t_i}], \quad i = N-1, \dots, 0. \quad (7) \end{aligned}$$

Hence, there are functions $y^\pi_i(x)$ and $z^\pi_i(x)$ such that

$$Y^\pi_{t_i} = y^\pi_i(X^\pi_{t_i}), \quad Z^\pi_{t_i} = z^\pi_i(X^\pi_{t_i}).$$

These functions $(y^\pi_i(x), z^\pi_i(x))$ are estimated recursively by least-squares Monte Carlo. To this end one chooses basis functions

$$\eta_0(i, x) = (\eta_{0,1}(i, x), \dots, \eta_{0,K}(i, x))$$

for the estimation of $y_i^\pi(x)$, and

$$\eta_d(i,x) = (\eta_{d,1}(i,x), \ldots, \eta_{d,K}(i,x)), \quad d = 1, \ldots, D,$$

for the estimation of the dth component $z_{d,i}^\pi(x)$ of $z_i^\pi(x)$. In principle, the number of basis functions can be different for each time step and for the y- and z-part, which we suppress for simplicity. Then, given L independent copies $(\Delta_\lambda W_i,\ {}_\lambda X_{t_{i+1}}^\pi)_{i=0,\ldots,N-1}$, $\lambda = 1, \ldots, L$, of $(\Delta W_i, X_{t_{i+1}}^\pi)_{i=0,\ldots,N-1}$, we define

$$\begin{aligned}
\tilde{y}_N^{\pi,K,L}(x) &= g(x), \quad \tilde{z}_N^{\pi,K,L} = 0, \\
\alpha_{d,i}^{\pi,K,L} &= \arg\min_{\alpha\in\mathbb{R}^K} \frac{1}{L}\sum_{\lambda=1}^{L}\left(\eta_d(i,\ {}_\lambda X_{t_i}^\pi)\,\alpha - \frac{\Delta_\lambda W_{d,i}}{\Delta_i}\tilde{y}_{i+1}^{\pi,K,L}({}_\lambda X_{t_{i+1}}^\pi)\right)^2 \\
\tilde{z}_{d,i}^{\pi,K,L}(x) &= \eta_d(i,x)\,\alpha_{d,i}^{\pi,K,L}, \quad d = 1, \ldots, D; \quad i = N-1, \ldots, 0, \\
\alpha_{0,i}^{\pi,K,L} &= \arg\min_{\alpha\in\mathbb{R}^K} \frac{1}{L}\sum_{\lambda=1}^{L}\Big(\eta_0(i,\ {}_\lambda X_{t_i}^\pi)\,\alpha - \tilde{y}_{i+1}^{\pi,K,L}({}_\lambda X_{t_{i+1}}^\pi) \\
&\qquad + f(t_i,\ {}_\lambda X_{t_i}^\pi, \tilde{y}_{i+1}^{\pi,K,L}({}_\lambda X_{t_{i+1}}^\pi), \tilde{z}_i^{\pi,K,L}({}_\lambda X_{t_i}^\pi))\Delta_i\Big)^2 \\
\tilde{y}_i^{\pi,K,L}(x) &= \eta_0(i,x)\,\alpha_{0,i}^{\pi,K,L}, \quad i = N-1, \ldots, 0.
\end{aligned} \tag{8}$$

Once the basis functions are chosen and the sample paths are generated, the algorithm is straightforward to implement, as it only requires to solve some linear least-squares problems numerically.

The L^2-error between $(\tilde{y}_i^{\pi,K,L}(x), \tilde{z}_{d,i}^{\pi,K,L}(x))$ and $(y_i^\pi(x), z_i^\pi(x))$ with respect to the law of $X_{t_i}^\pi$ has been analyzed by Lemor et al. (2006), Theorem 2 and Remark 1, for a suitably truncated scheme. The complete error analysis is rather technical, particularly because the use of the *same* simulated paths for estimating *all* conditional expectations induces a somewhat complicated dependency structure. We now roughly explain the influence of the different error sources, but refer the interested reader to the original paper by Lemor et al. (2006) for the very details. In order to simplify the presentation, we assume that the partition π of $[0,T]$ is equidistant with $(N+1)$ time points:

1. The *time discretization error* decreases at a rate of $N^{-1/2}$, see (6).

2. The *projection error* is induced by choosing the basis functions. The squared projection error can be bounded by a constant times

$$\sum_{i=0}^{N-1}\inf_{\alpha\in\mathbb{R}^K} E[|Y_{t_i}^\pi - \eta_0(i, X_{t_i}^\pi)\,\alpha|^2] + \sum_{d=1}^{D}\inf_{\alpha\in\mathbb{R}^K} E[|\sqrt{\Delta_i}Z_{d,t_i}^\pi - \eta_d(i, X_{t_i}^\pi)\,\alpha|^2]. \tag{9}$$

   Notice, that this expression is the sum of the squared distance between the time discretized solution $(Y_{t_i}^\pi, \sqrt{\Delta_i}Z_{t_i}^\pi)$ and its best projection on

the basis functions. The time discretized solution and its best projection are both not available in closed form (but for trivial cases). So this error bound is still difficult to quantify except for some special classes of basis functions such as indicator functions of hypercubes which form a partition of the state space of X, see Gobet et al. (2005).

Recall that throughout the algorithm conditional expectations of the form $E[Y^\pi_{t_{i+1}}|X^\pi_{t_i}]$ are approximated recursively for $i = N-1,\dots,0$. The approximation errors in the different time steps may sum up in the worst case, which explains the sum over time of the projection errors.

3. We finally discuss the *simulation error*. The results by Lemor et al. (2006) imply that it can be bounded in terms of the number of time points N (up to logarithmic factors) by $N^{-\rho/2}$ for $\rho \in [0,1]$, if the number of basis functions K increases proportional to N^δ, $\delta \geq 0$, and the number of simulated paths L increases proportional to $N^{2+2\delta+\rho}$.

   Here the worst contribution stems from estimating the conditional expectation $E[\frac{\Delta W_i^*}{\Delta_i}\ Y^\pi_{t_{i+1}}|X^\pi_{t_i}]$ for the Z-part, because the variance blows up when the time partition becomes finer due to the factor $\frac{\Delta W_i^*}{\Delta_i}$.

To sum up, a finer time partition requires a better choice of the basis functions (typically a significant increase in the number of basis functions), which in turn leads to a larger number of simulated paths. We note that the number of simulated paths must grow polynomially in the number of basis functions, while even an exponential growth of sample paths is necessary for the Longstaff-Schwartz algorithm for pricing American options, see Glasserman and Yu (2004). Nonetheless our numerical study in Section 4 will exhibit some limitations of the algorithm, when a fine time grid is required.

3 Martingale basis functions

In this section we propose the use of basis functions, which form a system of martingales. This approach is in the spirit of Glasserman and Yu (2002) who applied martingale basis functions for computing dual upper bounds for American options. We first motivate the martingale basis approach.

Taking another look at the time discretization scheme (7), we notice that three conditional expectations must be approximated in each time step,

namely

$$E\left[\frac{\Delta W_i^*}{\Delta_i} Y_{t_{i+1}}^\pi | X_{t_i}^\pi\right], \tag{10}$$

$$E[Y_{t_{i+1}}^\pi | X_{t_i}^\pi], \tag{11}$$

$$E[f(t_i, X_{t_i}^\pi, Y_{t_{i+1}}^\pi, Z_{t_i}^\pi)\Delta_i | X_{t_i}^\pi]. \tag{12}$$

We have observed in the previous section that estimating the conditional expectation in (10), which is related to the Z-part of the solution, is the dominant term for choosing the number of simulated paths in order to deal with the increasing variance of $\frac{\Delta W_i^*}{\Delta_i}$. Moreover, we have seen that estimating the conditional expectation in (11) leads to an unfortunate propagation in time of the projection error. So, estimating the conditional expectation in (12) appears to be numerically the easiest of the three estimation problems, particularly as the multiplication with the time step Δ_i is expected to reduce the error. Hence, our aim is to choose the basis functions in such a way that the conditional expectations in (10) and (11) can be computed in closed form, when $Y_{t_{i+1}}^\pi$ is replaced by a linear combination of basis functions.

To fix the ideas, let us assume that, at time t_{i+1}, an approximation $\hat{y}_{i+1}^{\pi,K,L}(X_{t_{i+1}}^\pi)$ of $Y_{t_{i+1}}^\pi = y_{i+1}^\pi(X_{t_{i+1}}^\pi)$ is already constructed and $\hat{y}_{i+1}^{\pi,K,L}(x)$ is a linear combination of basis functions, i.e.

$$\hat{y}_{i+1}^{\pi,K,L}(x) = \sum_{k=1}^{K} \beta_k \, \eta_{0,k}(i+1, x)$$

for some $\beta_1, \ldots, \beta_K \in \mathbb{R}$. If the basis functions form martingales in the following sense

$$E[\eta_{0,k}(i+1, X_{t_{i+1}}^\pi) | X_{t_i}^\pi = x] = \eta_{0,k}(i, x),$$

we can compute the conditional expectation of type (11) in closed form:

$$E[\hat{y}_{i+1}^{\pi,K,L}(X_{t_{i+1}}^\pi) | X_{t_i}^\pi] = \sum_{k=1}^{K} \beta_k \, \eta_{0,k}(i, X_{t_i}^\pi).$$

Similar considerations for the conditional expectation of type (10) then lead to the following assumption on the basis choice.

Assumption 3.1. We choose, at time $t_N = T$, a row vector of K basis functions

$$\eta_0(N, x) = (\eta_{0,1}(N, x), \ldots, \eta_{0,K}(N, x)).$$

Then, we define the basis functions $\eta_d(i, x) = (\eta_{d,1}(i, x), \ldots, \eta_{d,K}(i, x))$, $d = 0, \ldots, D$, at the earlier time steps $i = 0, \ldots, N-1$ via the conditional expectations

$$\eta_{0,k}(i, x) = E[\eta_{0,k}(N, X_{t_N}^\pi) | X_{t_i}^\pi = x] \tag{13}$$

$$\eta_{d,k}(i, x) = E\left[\frac{\Delta W_{d,i}}{\Delta_i} \eta_{0,k}(N, X_{t_N}^\pi) \middle| X_{t_i}^\pi = x\right], \quad d = 1, \ldots, D, \tag{14}$$

which we assume to be computable in closed form.

The terminology *martingale basis functions* refers to the setting of Assumption 3.1. Note, that by the tower property of the conditional expectations, we have

$$\eta_{0,k}(i,x) = E[\eta_{0,k}(i+1, X^\pi_{t_{i+1}})|X^\pi_{t_i} = x], \tag{15}$$

$$\eta_{d,k}(i,x) = E\left[\frac{\Delta W_{d,i}}{\Delta_i}\,\eta_{0,k}(i+1, X^\pi_{t_{i+1}})\middle| X^\pi_{t_i} = x\right], \quad d = 1,\dots,D \tag{16}$$

Before we provide some examples for martingale basis functions, we first explain how the least-squares Monte Carlo algorithm for BSDEs can be simplified, when a set of martingale basis functions is available. The modified algorithm exploits properties (15)–(16).

If, for the terminal condition g, the conditional expectations

$$E[g(X^\pi_{t_N})|X^\pi_{t_i} = x], \quad E\left[\frac{\Delta W_{d,i}}{\Delta_i}\,g(X^\pi_{t_N})\middle| X^\pi_{t_i} = x\right]$$

are available in closed form, one, of course, adds g to the martingale basis. Otherwise an initialization step at time $t_N = T$ is required in order to approximate the terminal condition g by a linear combination of basis functions. Such approximation can e.g. be done by a least-squares Monte Carlo projection of g on the basis:

$$\beta^{\pi,K,L}_N = \arg\min_{\beta\in\mathbb{R}^K} \frac{1}{L}\sum_{\lambda=1}^{L}\Big(\eta_0(N,\,{}_\lambda X^\pi_{t_N})\,\beta - g({}_\lambda X^\pi_{t_N})\Big)^2,$$

where here and in the following the averaging is again over independent sample copies $(\Delta_\lambda W_i,\ {}_\lambda X_{t_{i+1}})_{i=0,\dots,N-1}$, $\lambda = 1,\dots,L$, of $(\Delta W_i, X_{t_{i+1}})_{i=0,\dots,N-1}$.

In any case, we suppose that a vector $\beta^{\pi,K,L}_N \in \mathbb{R}^K$ has been chosen and $\eta_0(N,x)\,\beta^{\pi,K,L}_N$ is interpreted as an approximation of $g(x)$. Given $\beta^{\pi,K,L}_N$ the modified algorithm computes, for $i = N-1,\dots,0$,

$$\begin{aligned}
\hat{y}^{\pi,K,L}_{i+1}(x) &= \eta_0(i+1,x)\,\beta^{\pi,K,L}_{i+1}\\
\hat{z}^{\pi,K,L}_{d,i}(x) &= \eta_d(i,x)\,\beta^{\pi,K,L}_{i+1}, \quad d = 1,\dots,D,\\
\bar{\beta}^{\pi,K,L}_i &= \arg\min_{\beta\in\mathbb{R}^K} \frac{1}{L}\sum_{\lambda=1}^{L}\Big(\eta_0(i,\,{}_\lambda X^\pi_{t_i})\,\beta\\
&\qquad + f(t_i,\,{}_\lambda X^\pi_{t_i}, \hat{y}^{\pi,K,L}_{i+1}({}_\lambda X^\pi_{t_{i+1}}), \hat{z}^{\pi,K,L}_i({}_\lambda X^\pi_{t_i}))\Delta_i\Big)^2\\
\beta^{\pi,K,L}_i &= \beta^{\pi,K,L}_{i+1} + \bar{\beta}^{\pi,K,L}_i.
\end{aligned} \tag{17}$$

The algorithm terminates at time $t = 0$ with

$$\hat{y}^{\pi,K,L}_0(x) = \eta_0(0,x)\,\beta^{\pi,K,L}_0.$$

The final approximation for $(Y_{t_i}^\pi, Z_{t_i}^\pi)$ is given by $(\hat{y}_i^{\pi,K,L}(X_{t_i}^\pi), \hat{z}_i^{\pi,K,L}(X_{t_i}^\pi))$.

We emphasize that in the modified algorithm, by employing properties (15)–(16) of the martingale basis functions, only the conditional expectation of type (12) is approximated by least-squares Monte Carlo.

We now give some examples for basis functions which can be included into martingale bases, when the forward SDE is a (multi-dimensional) geometric Brownian motion. This situation corresponds to the numerical examples in Section 4.

Example 3.2. Suppose we are given D Black-Scholes stocks, which are for simplicity assumed to be independent and identically distributed, i.e.

$$X_{d,t} = x_0 \exp\{(\mu - \sigma^2/2)t + \sigma W_{d,t}\}, \quad d = 1, \dots, D,$$

where $x_0, \sigma > 0$ and $\mu \in \mathbb{R}$. Here, X can be sampled perfectly, and we hence write X instead of X^π. The martingale basis functions which we apply for the numerical examples below are built from indicator functions of hypercubes, monomials, and the payoff function of a max-call option.

For the indicator functions of the form $\eta_{\mathbf{a},\mathbf{b}} := \mathbf{1}_{[\mathbf{a},\mathbf{b}]} = \mathbf{1}_{[a_1,b_1]\times\cdots\times[a_D,b_D]}$ one easily calculates,

$$E[\eta_{\mathbf{a},\mathbf{b}}(X_T)|X_{t_i} = x] = \prod_{d=1}^D E[\mathbf{1}_{[a_d,b_d]}(X_{d,T})|X_{d,t_i} = x_d] = \prod_{d=1}^D \mathcal{N}(\tilde{a}_d) - \mathcal{N}(\tilde{b}_d),$$

where $\mathcal{N}$ is the cumulative distribution function of a standard normal and for $y = a, b$

$$\tilde{y}_d = \frac{\log(y_d/x_d) - (\mu - 0.5\sigma^2)(T - t_i)}{\sigma\sqrt{T - t_i}}.$$

For monomials $\eta_{\mathbf{p}}(x) := x_1^{p_1} \cdots x_D^{p_D}$ one has

$$E[\eta_{\mathbf{p}}(X_T)|X_{t_i} = x] = \prod_{d=1}^D x_d^{p_d} \exp\{(p_d\mu + 0.5p_d(p_d - 1)\sigma^2)(T - t_i)\}.$$

For the payoff functions of a max-call option $\eta_K(x) = (\max_{d=1,\dots,D} x_d - K)_+$, it can be derived from the results by Johnson (1987) that

$$\begin{aligned} E[\eta_K(X_T)|X_{t_i} = x] = &\sum_{d=1}^D e^{\mu(T-t_i)} x_d \mathcal{N}_{0,\Sigma}(a_{d,+}) \\ &- K(1 - \prod_{d=1}^D \mathcal{N}(\frac{\log(K/x_d) - (\mu - 0.5\sigma^2)(T - t_i)}{\sigma\sqrt{T - t_i}})), \end{aligned}$$

where $\mathcal{N}_{0,\Sigma}$ is the distribution function of a D-variate normal with mean vector 0 and covariance matrix Σ. Moreover,

$$a_{d,+} = \frac{1}{\sigma\sqrt{T-t_i}} \begin{pmatrix} \log(x_d/K) + (\mu + 0.5\sigma^2)(T-t_i) \\ \frac{1}{\sqrt{2}}(\log(x_d/x_{\tilde{d}}) + \sigma^2(T-t_i)) \\ \vdots \\ \frac{1}{\sqrt{2}}(\log(x_d/x_D) + \sigma^2(T-t_i)) \end{pmatrix}, \tilde{d} = 1, \ldots, D, \tilde{d} \neq d,$$

and

$$\Sigma = \begin{pmatrix} 1 & 1/\sqrt{2} & 1/\sqrt{2} & \cdots & 1/\sqrt{2} \\ 1/\sqrt{2} & 1 & 1/2 & \cdots & 1/2 \\ 1/\sqrt{2} & 1/2 & 1 & & 1/2 \\ \vdots & \vdots & & \ddots & \vdots \\ 1/\sqrt{2} & 1/2 & \cdots & 1/2 & 1 \end{pmatrix}$$

Hence, for such functions the conditional expectations required in (13) are available.

Concerning the conditional expectations of the form (14), we assume that $\eta(x)$ is a function such that $\eta_0(i,x) := E[\eta(X_t)|X_{t_i} = x]$ can be computed. Under appropriate growth conditions (which allow to introduce the derivatives below under the integral sign), we get for $d = 1, \ldots, D$ and $i < N$,

$$\eta_d(i,x) := E\left[\frac{\Delta W_{d,i}}{\Delta_i}\,\eta(X^\pi_{t_N})\middle| X^\pi_{t_i} = x\right] = \sigma x_d \frac{\partial}{\partial x_d}\eta_0(i,x). \tag{18}$$

Indeed, for the one-dimensional case ($D = 1$) one easily computes

$$\begin{aligned}
&\sigma x \frac{d}{dx}\eta_0(i,x) = \sigma x \frac{d}{dx} E\left[\eta_0(i+1, X_{t_{i+1}})\,|X_{t_i} = x\right] \\
&= \sigma x \frac{1}{\sqrt{2\pi\Delta_i}} \int_{-\infty}^{\infty} e^{-\frac{u^2}{2\Delta_i}} \frac{d}{dx}\eta_0(i+1, xe^{\sigma u + (\mu - 0.5\sigma^2)\Delta_i})du \\
&= \frac{1}{\sqrt{2\pi\Delta_i}} \int_{-\infty}^{\infty} e^{-\frac{u^2}{2\Delta_i}} \frac{d}{du}\eta_0(i+1, xe^{\sigma u + (\mu - 0.5\sigma^2)\Delta_i})du \\
&= \frac{1}{\sqrt{2\pi\Delta_i}} \int_{-\infty}^{\infty} \eta_0(i+1, xe^{\sigma u + (\mu - 0.5\sigma^2)\Delta_i}) \frac{d}{du}\left(-e^{-\frac{u^2}{2\Delta_i}}\right) du \\
&= \frac{1}{\sqrt{2\pi\Delta_i}} \int_{-\infty}^{\infty} \eta_0(i+1, xe^{\sigma u + (\mu - 0.5\sigma^2)\Delta_i}) \frac{u}{\Delta_i} e^{-\frac{u^2}{2\Delta_i}} du \\
&= E\left[\frac{\Delta W_i}{\Delta_i}\eta_0(i+1, X_{t_{i+1}})\,|X_{t_i} = x\right] = E\left[\frac{\Delta W_i}{\Delta_i}\,\eta(X^\pi_{t_N})\middle| X^\pi_{t_i} = x\right].
\end{aligned}$$

The multi-dimensional case can be treated analogously. Using formula (18) we can then calculate the conditional expectations (14) for e.g. the indicator functions, monomials, and the call payoff.

Remark 3.3. The above example is, admittedly, somewhat simplistic. We note, however, that for more sophisticated models, good closed-form approximations for many European option prices and their deltas are often available. These can be applied to built basis functions in the spirit of the previous example, which at least approximately fit into the martingale basis setting.

We now study the projection error, i.e. the error induced by choosing the basis functions, in the setting of martingale basis functions. In order to separate this error from the simulation error, we now assume that the orthogonal projections on the basis can be computed in closed form. Hence, we define

$$\beta_N^{\pi,K} = \arg\min_{\beta\in\mathbb{R}^K} E\Big[\eta_0(N, X_{t_N}^\pi)\,\beta - g(X_{t_N}^\pi)\Big]^2,$$

and for $i = N-1, \dots, 0$,

$$\begin{aligned}
\hat{y}_{i+1}^{\pi,K}(x) &= \eta_0(i+1,x)\,\beta_{i+1}^{\pi,K} \\
\hat{z}_{d,i}^{\pi,K}(x) &= \eta_d(i,x)\,\beta_{i+1}^{\pi,K}, \quad d = 1,\dots,D, \\
\bar{\beta}_i^{\pi,K} &= \arg\min_{\beta\in\mathbb{R}^K} E\Big[\eta_0(i, X_{t_i}^\pi)\,\beta \\
&\qquad + f(t_i, X_{t_i}^\pi, \hat{y}_{i+1}^{\pi,K}(X_{t_{i+1}}^\pi), \hat{z}_i^{\pi,K}(X_{t_i}^\pi))\Delta_i\Big]^2 \\
\beta_i^{\pi,K} &= \beta_{i+1}^{\pi,K} + \bar{\beta}_i^{\pi,K}. \qquad (19)
\end{aligned}$$

At time $t = 0$ we set

$$\hat{y}_0^{\pi,K}(x) = \eta_0(0,x)\,\beta_0^{\pi,K}.$$

Theorem 3.4. *Under Assumptions 2.1 and 3.1, there is a constant C such that*

$$\begin{aligned}
&\max_{0\le i\le N} E[|Y_{t_i}^\pi - \hat{y}_i^{\pi,K}(X_{t_i}^\pi)|^2] + \sum_{i=0}^{N-1} E[|Z_{t_i}^\pi - \hat{z}_i^{\pi,K}(X_{t_i}^\pi)|^2]\Delta_i \\
\le\ & C\Bigg(\inf_{\beta\in\mathbb{R}^K} E[|\eta_0(N, X_{t_N}^\pi)\,\beta - g(X_{t_N}^\pi)|^2] \\
&+ \sum_{i=0}^{N-1} \Delta_i \inf_{\beta\in\mathbb{R}^K} E[|\eta_0(i, X_{t_i}^\pi)\,\beta - E[f(t_i, X_{t_i}^\pi, Y_{t_{i+1}}^\pi, Z_{t_i}^\pi)|X_{t_i}^\pi]|^2]\Bigg) \qquad (20)
\end{aligned}$$

The proof will be postponed to the Appendix.

Remark 3.5. Recall that the first term on the right hand side of (20) vanishes, when the terminal condition g can be added to the martingale basis. The remaining error term averages over time the squared projection errors between $E[f(t_i, X_{t_i}^\pi, Y_{t_{i+1}}^\pi, Z_{t_i}^\pi)|X_{t_i}^\pi]$ and its best projection on the basis. So here we do not observe the unfavorable error propagation over time, which we found in the upper bound for the projection error of the orignial scheme in (9).

Remark 3.6. We notice that, by a straightforward application of the law of large numbers, the simulation error in the martingale basis setting converges to zero, as the number of simulated paths L tends to infinity. A preliminary error analysis for a suitably truncated scheme suggests, that the simulation error converges at $N^{-\rho/2}$ for $\rho \in [0,1]$ (N the number of time steps in an equidistant partition), if the number of basis functions K increases proportional to N^{δ}, $\delta \geq 0$, and the number of simulated paths L increases proportional to $N^{2+\delta+\rho}$ (compared to $N^{2+2\delta+\rho}$ in the original scheme). A detailed analysis is, however, beyond the scope of this paper.

4 Numerical experiments

4.1 The test example

We now introduce the test example for our numerical experiment, which is the pricing problem of a call spread option under different interest rates. Actually, this example is taken from Lemor et al. (2006) and hence allows for a comparison with their results. We shall also consider some variations of this example in order to study the influence of larger Lipschitz constants and multi-dimensional situations.

Suppose we are given a market with D risky assets X_t, which are modeled by Black-Scholes stocks. For simplicity we assume that the D stocks are independent and identically distributed, i.e.

$$X_{d,t} = x_0 \exp\{(\mu - \sigma^2/2)t + \sigma W_{d,t}\}, \quad d = 1, \dots, D,$$

where $W_t = (W_{1,t}, \dots, W_{D,t})$ is a D-dimensional Brownian motion and $x_0, \sigma, \mu > 0$. The trader can also invest into a riskless bond with rate $r \geq 0$ for investing and rate $R \geq r$ for borrowing from the bond. Our aim is to price a call spread option on the maximum of the stocks, which here is assumed to be of the form

$$\xi = \left(\max_{d=1,\dots,D} X_{d,T} - K_1 \right)_+ - 2 \left(\max_{d=1,\dots,D} X_{d,T} - K_2 \right)_+$$

for constants $K_1, K_2 > 0$. Following Lemor et al. (2006) we choose the constants

$$x_0 = 100,\ \mu = 0.05,\ \sigma = 0.2,\ T = 0.25,\ r = 0.01,\ K_1 = 95,\ K_2 = 105.$$

As interest rate for borrowing we choose $R = 0.06$ for the economically sensible case with a small Lipschitz constant. In order to test the algorithms in a situation with larger Lipschitz constant we shall also consider the case $R = 3.01$. We run this problem for the one-dimensional case ($D = 1$), where the option reduces to a call spread option on a single stock, and for the three-dimensional problem ($D = 3$).

It follows from results by Bergman (1995) that this option pricing problem under different interest rates can be formulated in terms of a BSDE by

$$\begin{aligned} Y_t &= \xi - \int_t^T \Big(rY_s + \frac{(\mu - r)}{\sigma} \sum_{d=1}^{D} Z_{d,s} - (R - r) \big(Y_s - \sigma^{-1} \sum_{d=1}^{D} Z_{d,s} \big)_- \Big) ds \\ &\quad - \sum_{d=1}^{D} \int_t^T Z_{d,s} dW_{d,s}. \end{aligned}$$

Note that in the case of a vanilla call option, the investor is bound to perpetually borrow money in order to hedge the option. Hence the closed-form solution for such option is given by the hedging problem in a standard Black-Scholes setting with a bank account given by e^{Rt}. Contrarily, for the call spread option case the problem is truly nonlinear and the solution (Y, Z) of the BSDE is not available in closed form. Therefore we require a tool to measure the performance of the numerical algorithm. We here stick to an error criterion suggested and studied in Bender and Steiner (2010). We now explain the idea in the general setting of the present paper.

Let us suppose that some approximation $(\hat{y}_i^\pi(x), \hat{z}_i^\pi(x))$ of $(y_i^\pi(x), z_i^\pi(x))$ for every $t_i \in \pi$ was computed by some numerical scheme. In the examples we consider the approximations obtained by the least-squares Monte Carlo scheme $(\tilde{y}_i^{\pi,K,L}(x), \tilde{z}_i^{\pi,K,L}(x))$ in (8) and by the martingale basis scheme $(\hat{y}_i^{\pi,K,L}(x), \hat{y}_i^{\pi,K,L}(x))$ in (17). Given a generic approximation $(\hat{y}_i^\pi(x), \hat{z}_i^\pi(x))$, we set

$$(\hat{Y}_{t_i}^\pi, \hat{Z}_{t_i}^\pi) = (\hat{y}_i^\pi(X_{t_i}^\pi), \hat{z}_i^\pi(X_{t_i}^\pi))$$

and define $(\hat{Y}_{t_i}^\pi, \hat{Z}_{t_i}^\pi)$, $t \in [0, T]$, by piecewise constant interpolation.

Then we consider as an error criterion

$$\begin{aligned} &\mathcal{E}_\pi(\hat{y}^\pi, \hat{z}^\pi) := E[|g(X_{t_N}^\pi) - \hat{Y}_{t_N}^\pi|^2] \\ + \quad &\max_{0 \le i \le N} E[|\hat{Y}_{t_i}^\pi - \hat{Y}_{t_0}^\pi - \sum_{j=0}^{i-1} f(t_j, X_{t_j}^\pi, \hat{Y}_{t_j}^\pi, \hat{Z}_{t_j}^\pi)\Delta_j - \sum_{j=0}^{i-1} \hat{Z}_{t_j}^\pi \Delta W_j|^2]. \end{aligned}$$

We emphasize that this criterion does only depend on the numerical solution $(\hat{y}_{t_i}^\pi(x), \hat{z}_{t_i}^\pi(x))$ and, thus, can be consistently estimated by a plain Monte Carlo approach. The second term on the right hand side measures, whether the approximative solution is 'close' to solving the SDE (run as a forward SDE). The first term on the right hand side measures how well it fits to the terminal condition. So, in a sense, we check how close the approximative solution is to solving the BSDE, while we are actually interested in how close it is to the true solution of the BSDE.

On the one hand, the error criterion is of some interest quantitatively due to its simple and meaningful interpretation. Moreover, it is intuitively

clear that being close to solving the BSDE is necessary for being close to the solution of the BSDE. On the other hand, the criterion is also of interest qualitatively, because there are constants $c_1, c_2, C \geq 0$ such that for sufficiently fine partitions π

$$\begin{aligned} & c_1 \mathcal{E}_\pi(\hat{y}^\pi, \hat{z}^\pi) - c_2|\pi| \qquad (21) \\ \leq & \sup_{t\in[0,T]} E[|Y_t - \hat{Y}_t^\pi|^2] + \int_0^T E[|Z_t - \hat{Z}_t^\pi|^2]dt \\ \leq & C\left(\mathcal{E}_\pi(\hat{y}^\pi, \hat{z}^\pi) + |\pi|\right), \qquad (22) \end{aligned}$$

see Bender and Steiner (2010). This means that the square root of the error criterion is – up to terms of order 1/2 in the mesh size of the partition – equivalent to the L^2-error between approximation and true solution. We also emphasize that the constant c_2 can be taken as 0, when the driver $f(t, x, y, z)$ does not depend on (t, x) which is the case in our option pricing example. Thus, in such situation, we arrive at the improved lower bound

$$c_1 \mathcal{E}_\pi(\hat{y}^\pi, \hat{z}^\pi) \leq \sup_{t\in[0,T]} E[|Y_t - \hat{Y}_t^\pi|^2] + \int_0^T E[|Z_t - \hat{Z}_t^\pi|^2]dt. \qquad (23)$$

Remark 4.1. Note that we cannot expect that the squared L^2-error

$$\sup_{t\in[0,T]} E[|Y_t - \hat{Y}_t^\pi|^2] + \int_0^T E[|Z_t - \hat{Z}_t^\pi|^2]dt$$

converges to zero faster than at the order $|\pi|$, because this error typically corresponds to the L^2-regularity in t of the solution Y_t and so persists, even if $\hat{Y}_{t_i}^\pi$ coincides with Y_{t_i} on the grid π. So, by looking at the error criterion, we are mainly aiming to judge whether the way, in which the estimator for the conditional expectation is designed in dependence of the mesh of the partition, retains the convergence rate of order 1/2 in the mesh or not. The error criterion decreases more slowly than $|\pi|$ in the latter case.

4.2 Numerical results

Case 1: 'Small' Lipschitz constant

We first consider the one-dimensional case ($D = 1$) and set $R = 0.06$. In this case, the nonlinearity has a rather small Lipschitz constant of $(R - r)/\sigma = 0.25$. Concerning the time discretization we apply an equidistant partition with N time steps. For the original least-squares Monte Carlo scheme we choose as basis functions the payoff function of the call spread and, following Lemor et al. (2006), indicator functions of K equidistant intervals which form a partition of the domain $[40, 180]$. For the scheme based on martingale basis functions, we also use the payoff function and the same number of indicator functions at terminal time $t_N = T$, and then the

basis functions at the other time steps are computed by formulas (13) and (14). However, the intervals for the indicator function are not chosen in an equidistant way, but such that X_T hits each interval with equal probability. For different values $\beta, \gamma > 0$, we choose in dependence of $\nu \in \mathbb{N}$

$$N = \left[2\sqrt{2}^{(\nu-1)}\right], \quad K = \left[\frac{14}{5}\sqrt{2}^{(\beta+1)(\nu-1)/2}\right] + 2, \quad L = \left[2\sqrt{2}^{\gamma(\nu-1)}\right].$$

Table 1 shows the numerical approximations for the price Y_0 of the call spread option under different interest rates for borrowing and investing. Here, LSM stands for the original least-squares Monte Carlo scheme by Lemor et al. (2006) and MBF stands for the use of martingale basis functions.

			Y_0													
		N	2	3	4	6	8	11	16	23	32	45	64	91	128	181
β	γ	type														
1	3	LSM	1.85	1.25	4.17	2.86	2.53	2.81	2.82	2.98	2.93	2.95	2.93	2.95	2.95	2.95
	4	LSM	7.63	3.95	2.66	2.99	2.71	3.14	2.86	2.95	2.93	2.94	2.95			
	5	LSM	3.59	3.52	2.56	2.63	2.82	2.88	2.94	2.93	2.94	2.94				
	3	MBF	2.79	4.51	2.90	2.92	2.93	2.94	2.95	2.95	2.95	2.95	2,96	2,96	2,96	2,96
0.5	3	LSM	6.20	4.56	3.01	3.33	3.63	3.05	2.90	2.82	2.94	2.90	2.95	2.93	2.94	2.94
	4	LSM	1.08	3.17	2.58	3.18	3.10	2.80	2.77	2.89	2.90	2.93	2.94	2.94		
	2	MBF	2.81	2.65	4.01	3.06	2.95	2.95	2.95	2.95	2.95	2.95	2.96	2.96	2.96	2.96
	3	MBF	2.84	2.85	2.97	2.93	2.93	2.94	2.95	2.95	2.95	2.95	2.96	2.96	2.96	2.96

Table 1: Numerical price Y_0 of the call spread option

For all variations of the two algorithms the numerical prices converge to values around 2.96. Overall, the convergence of the MBF-algorithm appears to be faster than for the LSM-algorithm. Moreover, in this example in the MBF-algorithm a faster increase of the number of basis functions ($\beta = 1$ vs. $\beta = 0.5$) and a faster increase of the number of sample paths ($\gamma = 3$ vs. $\gamma = 2$) does not significantly change the numerical results. Contrarily, for the LSM-algorithm, the values for Y_0 are improved by increasing β and γ. We emphasize that the choice of the parameters β and γ may drastically change the computational effort. For instance, for $N = 45$ and $\gamma = 5$, about 12 million paths must be simulated, while for $N = 45$ and $\gamma = 2$ only 1024 paths are required.

In order to derive information about the quality of the whole numerical solution (Y-part and Z-part at all time points) and not only about the Y_0-value, we plot the error criterion, which we motivated in the previous subsection. Figure 1 illustrates the error criterion (on a logarithmic scale) for $\beta = 1$, which is estimated using a new sample of L independent paths. In this case, the projection error in the LSM-scheme theoretically converges at order $1/2$ in the number of time steps N. In order to get the same theoretical convergence rate (up to logarithmic factors) for the simulation error, $\gamma = 5$ is required. $\gamma = 4$ is the theoretical threshold for convergence, while for $\gamma = 3$ convergence of the simulation error is not supported by the theoretical analysis in Lemor et al. (2006). The error criterion is smaller for a larger number of sample paths (i.e. larger values of γ), which indicates that

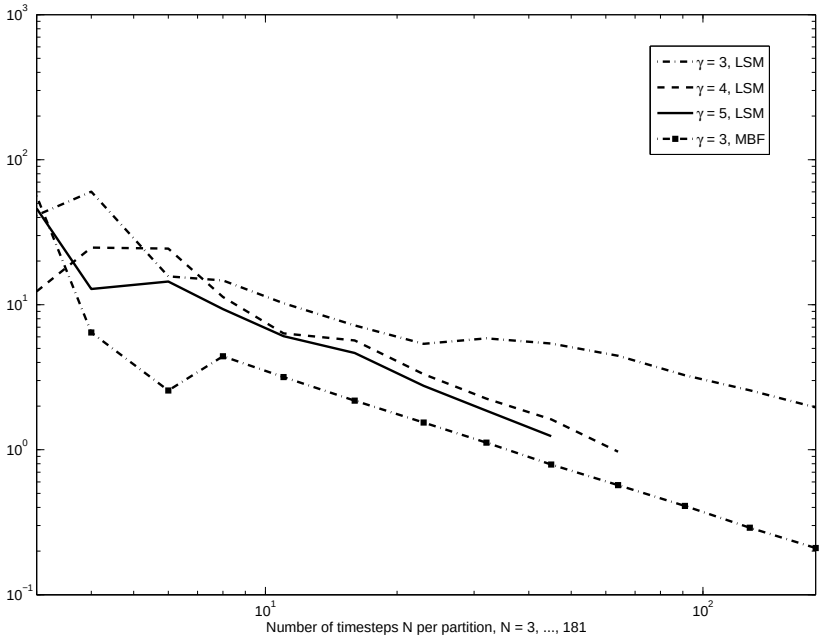

Figure 1: Error criterion for $\beta = 1$

the larger computational effort improves on how close the numerical solution is to solving the BSDE. Somewhat surprisingly, the difference between the cases $\gamma = 5$ and $\gamma = 4$ is rather small and for both values of γ a convergence of the LSM-scheme at order $1/2$ in the number of time steps is indicated by the error criterion. For $\gamma = 3$ the error criterion is significantly larger. Here it is less obvious, whether the LSM-scheme with $\beta = 3$ converges at all, but definitely it does not seem to converge at the same speed as $\gamma = 4, 5$. For the MBF-algorithm we observe, that the error criterion is significantly lower with $\gamma = 3$ than it is for the LSM-scheme with $\gamma = 5$. The slope of the line of about -0.95 suggests that the MBF-algorithm with $\gamma = 3$ converges almost at rate of $1/2$. We recall that it is hardly possible to run the LSM-algorithm with $\gamma \geq 4$ for larger values than $N = 64$ (and hence to further decrease the error) in an acceptable time due to the tremendous simulation costs.

Figure 2 shows the error criterion for the case $\beta = 0.5$. Here, for the LSM-algorithm, the projection error theoretically decreases as $N^{-1/4}$, and so does the simulation error (up to logarithmic factors) for $\gamma = 4$. The theoretical convergence threshold for the simulation error is $\gamma = 3$. A look at the error criterion indicates that the LSM-algorithm for $\gamma = 2$ does not seem to converge in accordance with the theoretical error bounds. For $\gamma = 3$ and $\gamma = 4$, the error criterion only slightly differs. The slope of the lines is about -0.9 in both cases, which corresponds to a rate of about 0.45. This suggests that, in practice, the worst case error propagation backwards in time, which is reflected in theoretical rate $1/4$, is not present. Again, for the

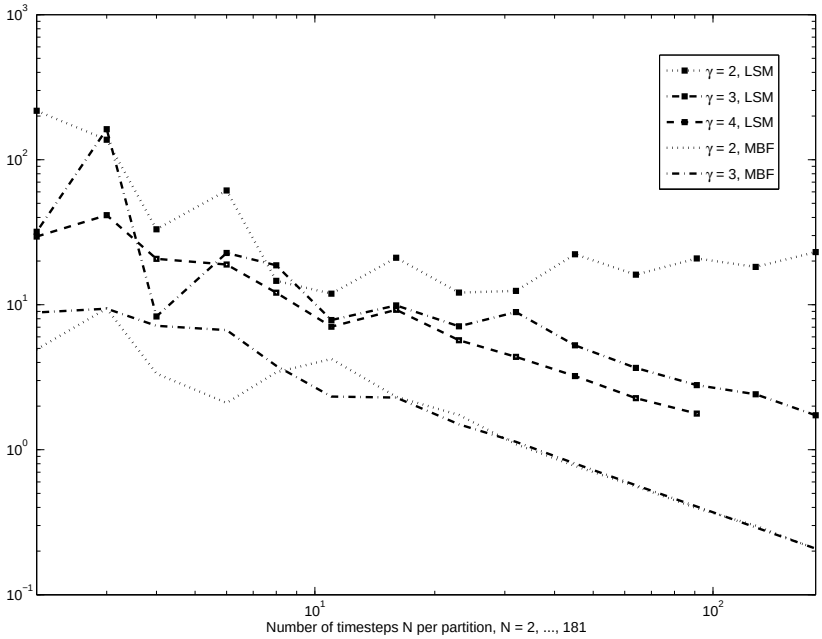

Figure 2: Error criterion for $\beta = 0.5$

MBF-scheme the error criterion is overall smaller and the scheme converges with lower simulation costs at $\gamma = 2$. Indeed, the additional simulation effort for $\gamma = 3$ does not improve the convergence behavior of the MBF-scheme. The slope is at -0.95 identical to the case $\beta = 1$.

In summary, in this example we find that using martingale basis functions leads to significant improvements of the numerical approximations of the whole solution of the BSDE. Moreover, the improved numerical solutions are computed with drastically less simulation effort.

Case 2: 'Large' Lipschitz constant

We now test the algorithms in a situation with a larger Lipschitz constant, but still in the one-dimensional case. As the Lipschitz constant of f enters exponentially in some of the error estimates, we expect that the numerical algorithms may run into difficulties. We set $R = 3.01$. Hence, the nonlinearity in f has as Lipschitz constant $(R - r)/\sigma = 15$. Of course, from the point of view of the financial application an interest rate of 301% is not relevant, but we believe that it is important to test the algorithms in some extreme situations as well. Moreover, as R tends to infinity, the price of the call spread option under different interest rates converges to the superhedging price under the no-borrowing constraint, see e.g. Bender and Kohlmann (2008). So the case of a large rate R for borrowing may still be of some interest from a financial point of view. We note that the superhedging price under the no-borrowing constraint can be computed analytically for

the call spread option by applying the techniques developed by Broadie et al. (1998). It is 7.18 and serves as an upper bound for our test BSDE, in which we use the same specification for the number of time steps, the basis functions, and the number of sample paths as in the previous example.

			Y_0													
β	γ	N type	2	3	4	6	8	11	16	23	32	45	64	91	128	181
1	4	LSM	29.52	5.14	13.52	11.78	6.78	8.67	7.41	7.36	113.35	157.49	4994.81			
	5	LSM	32.82	20.44	11.99	6.96	5.66	7.28	38.87	105.70	6.84	91.01				
	2	MBF	18.58	15.96	-22.23	-11.80	6.23	14128.47	8.84	7.99	7.33	6.90	6.66	6.53	6.47	6.44
	3	MBF	12.27	474.02	113.35	129.68	10.52	9.78	8.89	7.97	7.35	6.91	6.66	6.53	6.47	6.44

Table 2: Numerical value of Y_0 for the case with higher Lipschitz constant

Table 2 displays the numerical approximations for Y_0 calculated with the LSM-algorithm and the MBF-algorithm. On the one hand, for the LSM-algorithm no convergence pattern can be observed for $\gamma = 4$ and N up to 64 and $\gamma = 5$ and N up to 45. As in the latter case ($\gamma = 5$) the algorithm theoretically converges at a rate of $1/2$ in the number of time steps N, we conclude that larger values of N are required. As the number of sample paths also increases as N^γ, large values of N become, however, numerically untractable. Recall that $N = 45$ and $\gamma = 5$ already leads to 12 million sample paths. Nonetheless, the somewhat wild fluctuations in the estimated Y_0-values suggest that even larger number of sample paths cannot be avoided in the LSM-algorithm for this example. On the other hand, for the MBF-algorithm the pattern of the estimated Y_0-values apparently converges for $\gamma = 2$ and $\gamma = 3$. Convergence is not yet achieved for $N = 181$, but it seems plausible that Y_0 is about 6.40.

A look at the error criterion, which is plotted in Figure 3, confirms these observations. The LSM-algorithm is seen not to be in the range of convergence for the given values of N. For the MBF-algorithm we first note that the observed convergence behavior does not really differ for the cases $\gamma = 2$ and $\gamma = 3$. So, again, the use of more sample paths than for $\gamma = 2$ does not appear to be necessary for this scheme. It is interesting that the error criterion for the MBF-algorithm is comparable in absolute values to the case of the small Lipschitz constant for $N \geq 16$.

This example demonstrates that calculating some of the conditional expectations in closed form by using martingale basis functions stabilizes the algorithm. Hence the new algorithm based on martingale basis functions can compute reasonable approximations for the solution of the BSDE in situations, where the original algorithm already breaks down due to the large Lipschitz constant of the nonlinearity.

Case 3: Three-dimensional case

We finally return to the case of the small Lipschitz constant, i.e. the rate for borrowing R is again set to 6%, but we now price a call spread option on the maximum of three stocks ($D = 3$). In the previous examples

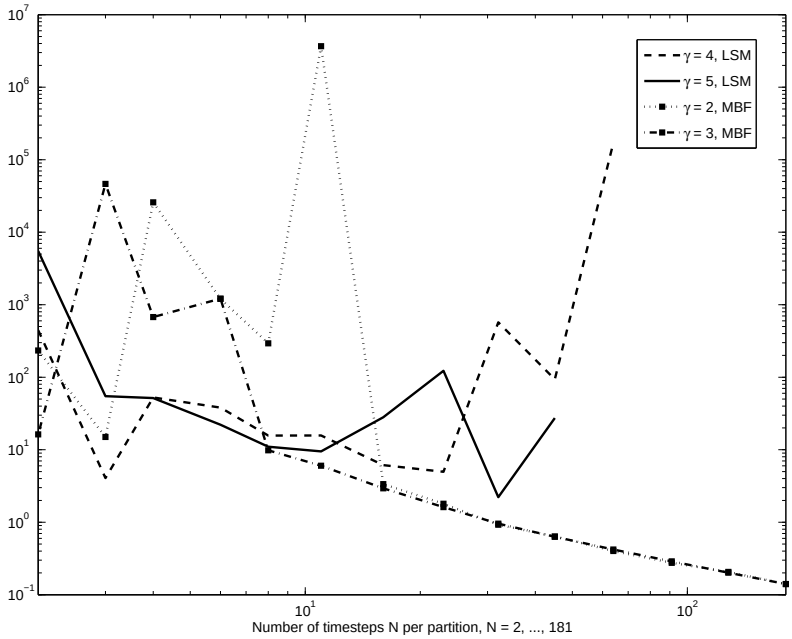

Figure 3: Error criterion for $\beta = 1$ and the case of a larger Lipschitz constant

the number of basis functions was increased with the number of time steps N, in order to make the projection error converge as N tends to infinity. In this example we test the use of a small number of basis functions. Here we take as basis functions the constant 1, the three first-order monomials, and the payoff function of the max-call-spread for the original least-squares approach. For the MBF-algorithm, the basis functions are only specified this way at terminal time and are the computed by formulas (13) and (14) at the other time points. Fixing a finite number of basis functions automatically introduces a bias to the numerical scheme which cannot be removed, but this procedure corresponds to what is usually done in Bermudan option pricing by the Longstaff-Schwartz algorithm. For the number of time steps and the number of sample paths we use the same specifications as before.

			Y_0									
		N	2	3	4	6	8	11	16	23	32	45
β	γ	type										
1	4	LSM	-0.96	6.30	2.61	3.03	2.90	3.09	3.05	3.08	3.12	3.09
	5	LSM	8.50	3.57	3.10	3.08	2.80	3.08	3.14	3.09	3.09	3.09
	2	MBF	3.05	4.64	3.03	3.10	3.10	3.11	3.12	3.12	3.12	3.12
	3	MBF	2.91	3.19	3.10	3.10	3.13	3.12	3.12	3.12	3.12	3.12

Table 3: Price Y_0 of the 3-dimensional max-call-spread

The numerical prices for the max-call-spread on three stocks under different interest rates are shown in Table 3. Here the values of the LSM-algorithm and the MBF-algorithm converge to similar but slightly different values. In both cases the number of simulated paths ($\gamma = 4$ vs. $\gamma = 5$ for the LSM-

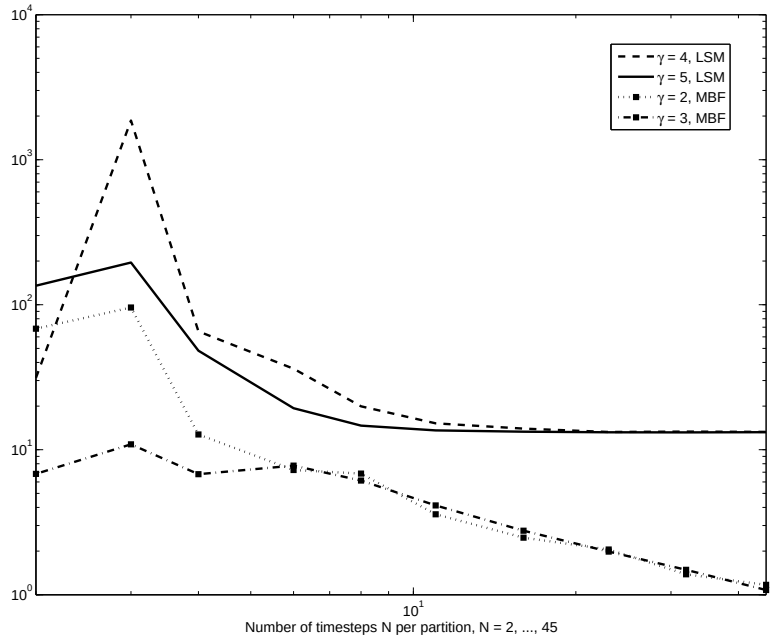

Figure 4: Error criterion for the 3-dimensional max-call-spread

algorithm, $\gamma = 2$ vs. $\gamma = 3$ for the MBF-algorithm), does not significantly change the convergence pattern.

We now look at the error criterion for this example (Figure 4). It shows that the simple basis consisting of the payoff function and some monomials is clearly inappropriate to recover the whole solution of the BSDE numerically. Indeed, for the LSM-scheme the error criterion stays roughly constant for $N \geq 11$ at a level larger than 10. This clearly indicates that the error arising from the choice of the small basis dominates the time discretization error and the simulation error, which both converge like $N^{-1/2}$. In the MBF-scheme the basis functions computed from the payoff function correspond to the price of the European option (without different interest rates) and to the deltas and are therefore automatically constructed in a more problem-specific way. We observe that for the MBF-algorithm and $N \leq 45$ the error criterion corresponds to a decrease of the error at order $1/2$. This indicates that the projection error is still dominated by the time discretization error and the simulation error for this range of N. We did not try larger values for N, but of course the projection error will be dominant for sufficiently large N. The key observation, which we make here, is that also for multi-dimensional problems a reasonable approximation of the whole solution of the BSDE may still be possible with only a few relevant basis functions, in particular when one can additionally exploit the fact that some of the conditional expectations can be computed in closed form by using martingale basis functions.

To conclude, in our numerical examples we find that the use of martingale basis functions yields significantly better numerical approximations at a much lower computational cost compared to the original least-squares Monte Carlo scheme. However, the new algorithm is less generic, because the construction of martingale basis functions depends on the law of X and restricts the choice of basis functions. So, we finally recommend to exploit the advantages of martingale basis functions when a good set of such functions is available.

A Proof of Theorem 3.4

Throughout the proof, C denotes a generic constant, which may vary from line to line. In order to simplify the notation, and without any real loss of generality, we restrict ourselves to the case $D = 1$. We also make use of the following abbreviations:

$$f_i^\pi := f(t_i, X_{t_i}^\pi, Y_{t_{i+1}}^\pi, Z_{t_i}^\pi), \quad f_i^{\pi,K} := f(t_i, X_{t_i}^\pi, \hat{y}_{t_{i+1}}^{\pi,K}(X_{t_i}^\pi), \hat{z}_{t_i}^{\pi,K}(X_{t_i}^\pi)).$$

Furthermore, P_i^K, $i = 0, \dots N$, denotes the orthogonal projection on the linear span of $\{\eta_{0,1}(i, X_{t_i}^\pi), \dots, \eta_{0,K}(i, X_{t_i}^\pi)\}$ as a subspace of $L^2(P)$. Then we obtain by the definitions in (5) and (19) and Young's inequality fo every $\gamma > 0$

$$\begin{aligned} E[|Y_{t_i}^\pi - \hat{y}_i^{\pi,K}(X_{t_i}^\pi)|^2] &\le (1+\gamma\Delta_i) E[|E[Y_{t_{i+1}}^\pi - \hat{y}_{i+1}^{\pi,K}(X_{t_{i+1}}^\pi)|X_{t_i}^\pi]|^2] \\ &\quad + (1+\gamma\Delta_i)\frac{\Delta_i}{\gamma} E[|P_i^K(f_i^{\pi,K}) - E[f_i^\pi|X_{t_i}^\pi]|^2] \\ &= (I) + (II). \end{aligned}$$

The orthogonality and the contraction property of P_i^K as well as the Lipschitz condition of f and the definition of P_i^K yield

$$\begin{aligned} (II) &= (1+\gamma\Delta_i)\frac{\Delta_i}{\gamma} E[|P_i^K(f_i^{\pi,K}) - P_i^K(f_i^\pi)|^2 + |P_i^K(f_i^\pi) - E[f_i^\pi|X_{t_i}^\pi]|^2] \\ &\le (1+\gamma\Delta_i)\frac{2\Delta_i\kappa^2}{\gamma} E[|Y_{t_{i+1}}^\pi - \hat{y}_{i+1}^{\pi,K}(X_{t_{i+1}}^\pi)|^2 + |Z_{t_i}^\pi - \hat{z}_i^{\pi,K}(X_{t_i}^\pi)|^2] \\ &\quad + (1+\gamma\Delta_i)\frac{\Delta_i}{\gamma} \inf_{\beta\in\mathbb{R}^K} E[|\eta_0(i, X_{t_i}^\pi)\beta - E[f_i^\pi|X_{t_i}^\pi]|^2]. \end{aligned} \tag{24}$$

Next, we obtain by the definitions in (5) and (19) and Hölder's inequality

$$
\begin{aligned}
&E[|Z^\pi_{t_i} - \hat{z}^{\pi,K}_i(X^\pi_{t_i})|^2] \\
&= E[|E[\frac{\Delta W_i}{\Delta_i}\{Y^\pi_{t_{i+1}} - \hat{y}^{\pi,K}_{i+1}(X^\pi_{t_{i+1}})\}|X^\pi_{t_i}]|^2] \\
&= E[|E[\frac{\Delta W_i}{\Delta_i}\{Y^\pi_{t_{i+1}} - \hat{y}^{\pi,K}_{i+1}(X^\pi_{t_{i+1}}) - E[Y^\pi_{t_{i+1}} - \hat{y}^{\pi,K}_{i+1}(X^\pi_{t_{i+1}})|X^\pi_{t_i}]\}|X^\pi_{t_i}]|^2] \\
&\leq \frac{1}{\Delta_i}E[|Y^\pi_{t_{i+1}} - \hat{y}^{\pi,K}_{i+1}(X^\pi_{t_{i+1}})|^2] - \frac{1}{\Delta_i}E[|E[Y^\pi_{t_{i+1}} - \hat{y}^{\pi,K}_{i+1}(X^\pi_{t_{i+1}})|X^\pi_{t_i}]|^2]. \quad (25)
\end{aligned}
$$

Applying this result on (24) for $\gamma = 2\kappa^2$ we get

$$
\begin{aligned}
E[|Y^\pi_{t_i} - \hat{y}^{\pi,K}_i(X^\pi_{t_i})|^2] \leq (1 + 2\kappa^2\Delta_i)(1 + \Delta_i)E[|Y^\pi_{t_{i+1}} - \hat{y}^{\pi,K}_{i+1}(X^\pi_{t_{i+1}})|^2] \\
+ C\Delta_i \inf_{\beta\in\mathbb{R}^K} E[|\eta_0(i, X^\pi_{t_i})\beta - E[f^\pi_i|X^\pi_{t_i}]|^2].
\end{aligned}
$$

Thanks to the discrete Gronwall inequality and the definiton of $y^{\pi,K}_N(X^\pi_{t_N})$, we get

$$
\begin{aligned}
E[|Y^\pi_{t_i} - \hat{y}^{\pi,K}_i(X^\pi_{t_i})|^2] \leq C \inf_{\beta\in\mathbb{R}^K} E[|\eta_0(N, X^\pi_{t_N})\beta - g(X^\pi_{t_N})|^2] \\
+ C\sum_{j=i}^{N-1} \inf_{\beta\in\mathbb{R}^K} E[|\eta_0(j, X^\pi_{t_j})\beta - E[f^\pi_j|X^\pi_{t_j}]|^2]. \quad (26)
\end{aligned}
$$

So we have proved the required approximation (20) for the Y-part. It remains to prove the upper bound for the Z-part. By (25) and the definitions

in (5) and (19) we have for every $\Gamma > 0$

$$
\begin{aligned}
& \sum_{i=0}^{N-1} \Delta_i E[|Z_{t_i}^\pi - \hat{z}_i^{\pi,K}(X_{t_i}^\pi)|^2] \\
\leq \; & \sum_{i=0}^{N-1} E[|Y_{t_{i+1}}^\pi - \hat{y}_{i+1}^{\pi,K}(X_{t_{i+1}}^\pi)|^2] - \sum_{i=0}^{N-1} E[|Y_{t_i}^\pi - \hat{y}_i^{\pi,K}(X_{t_i}^\pi)|^2] \\
& -2 \sum_{i=0}^{N-1} \Delta_i E[\{Y_{t_i}^\pi - \hat{y}_i^{\pi,K}(X_{t_i}^\pi)\}\{E[f_i^\pi | X_{t_i}^\pi] - P_i^K(f_i^{\pi,K})\}] \\
\leq \; & E[|Y_{t_N}^\pi - \hat{y}_N^{\pi,K}(X_{t_N}^\pi)|^2] + \sum_{i=0}^{N-1} \Delta_i \Gamma E[|Y_{t_i}^\pi - \hat{y}_i^{\pi,K}(X_{t_i}^\pi)|^2] \\
& + \sum_{i=0}^{N-1} \frac{\Delta_i}{\Gamma} E[|P_i^K(f_i^\pi - f_i^{\pi,K})|^2] + \sum_{i=0}^{N-1} \frac{\Delta_i}{\Gamma} E[|E[f_i^\pi | X_{t_i}^\pi] - P_i^K(f_i^\pi)|^2] \\
\leq \; & E[|Y_{t_N}^\pi - \hat{y}_N^{\pi,K}(X_{t_N}^\pi)|^2] + T(\Gamma + \frac{2\kappa^2}{\Gamma}) \max_{0 \leq i \leq N-1} E[|Y_{t_i}^\pi - \hat{y}_i^{\pi,K}(X_{t_i}^\pi)|^2] \\
& + \frac{2\kappa^2}{\Gamma} \sum_{i=0}^{N-1} \Delta_i E[|Z_{t_i}^\pi - \hat{z}_i^{\pi,K}(X_{t_i}^\pi)|^2] \\
& + \frac{1}{\Gamma} \sum_{i=0}^{N-1} \Delta_i \inf_{\beta \in \mathbb{R}^K} E[|\eta_0(i, X_{t_i}^\pi)\beta - E[f_i^\pi | X_{t_i}^\pi]|^2].
\end{aligned}
$$

Setting $\Gamma = 4\kappa^2$ and taking (26) into account immediately gives the estimate for the Z-part.

Acknowledgements

The authors gratefully acknowledge financial support by the Deutsche Forschungsgemeinschaft under grant BE3933/3-1.

References

V. Bally Approximation scheme for solutions of BSDE. In: El Karoui, N., Mazliak, L. (eds.) *Backward Stochastic Differential Equations*, Addison Wesley Longman (1997), 177-191.

V. Bally, G. Pagès, A quantization algorithm for solving multi-dimensional discrete-time optimal stopping problems, *Bernoulli* **9** (2003) 1003 - 1049.

C. Bender, R. Denk, A forward scheme for backward SDEs, *Stochastic Process. Appl.* **117** (2007) 1793 - 1812.

C. Bender, M. Kohlmann, Optimal superhedging under nonconvex constraints: a BSDE approach. *Int. J. Theor. Appl. Finance* **11**, 363-380.

C. Bender, T. Moseler, Importance sampling for backward SDEs. *Stoch. Analysis Appl.* **28** (2010) 226-253.

C. Bender, J. Steiner, Error criteria for numerical solutions of backward SDEs, Preprint (2010).

C. Bender, J. Zhang, Time discretization and Markovian iteration for coupled FBSDEs, *Ann. Appl. Probab.* **18** (2008) 143 - 177.

Y. Z. Bergman, Option pricing with differential interest rates. *Rev. of Financial Studies* **8** (1995) 475-500.

B. Bouchard, J.-F. Chassagneux, Discrete-time approximation for continuously and discretely reflected BSDEs, *Stochastic Process. Appl.* **118** (2008) 2269 - 2293.

B. Bouchard, R. Elie, Discrete-time approximation of decoupled forward-backward SDE with jumps, *Stochastic Process. Appl.* **118** (2008) 53 - 75.

B. Bouchard, R. Elie, N. Touzi, Discrete-time approximation of BSDEs and probabilistic schemes for fully nonlinear PDEs, Advanced financial modelling, *Radon Ser. Comput. Appl. Math.*, **8**, Walter de Gruyter, Berlin, (2009) 91 - 124.

B. Bouchard, N. Touzi, Discrete-time approximation and Monte Carlo simulation of backward stochastic differential equations, *Stochastic Process. Appl.* **111** (2004) 175 - 206.

P. Briand, B. Delyon, J. Mèmin, Donsker-type theorem for BSDEs, *Electron. Comm. Probab.* **6** (2001) 1 - 14.

M. Broadie, J. Cvitanic, M. Soner, Optimal replication of contingent claims under portfolio constraints, *Review of Financial Studies*, **11**, 59-79.

D. Chevance, Numerical methods for backward stochastic differential equations, *Numerical methods in finance*, Publ. Newton Inst., *Cambridge Univ. Press, Cambridge* (1997) 232 - 244.

D. Crisan, K. Manolarakis, Solving Backward Stochastic Differential Equations using the Cubature Method, Preprint (2010).

F. Delarue, S. Menozzi, A forward-backward stochastic algorithm for quasi-linear PDEs, *Ann. Appl. Probab.* **16** (2006) 140 - 184.

J. Douglas, J. Ma, P. Protter, Numerical methods for forward-backward stochastic differential equations, *Ann. Appl. Probab.* **6** (1996) 940 - 968.

N. El Karoui, S. Peng, M. C. Quenez, Backward stochastic differential equations in finance, *Math. Finance* **7** (1997) 1 - 71.

P.Glasserman, B. Yu, Pricing American options by simulation: Regression now or regression later? In: H. Niederreiter (ed.), *Monte Carlo and Quasi-Monte Carlo Methods*, Springer, Berlin, 2002.

P. Glasserman, B. Yu, Number of paths versus number of basis functions in American option pricing, *Ann. Appl. Probab.* **14** (2004) 2090–2119.

E. Gobet, C. Labart, Error expansion for the discretization of backward stochastic differential equations, *Stochastic Process. Appl.* **117** (2007) 803 - 829.

E. Gobet, C. Labart, Solving BSDE with adaptive control variate, *SIAM J. Numer. Anal.* **48** (2010) 257 - 277.

E. Gobet, J.-P. Lemor, X. Warin, A regression-based Monte Carlo method to solve backward stochastic differential equations, *Ann. Appl. Probab.* **15** (2005) 2172 - 2202.

E. Gobet, A. Makhlouf, L^2-time regularity of BSDEs with irregular terminal functions, *Stochastic Process. Appl.* **120** (2010) 1105-1132.

M. Gunzburger, G. Zhang, Efficient Numerical Methods for High-Dimensional Backward Stochastic Differential Equations, Preprint (2010).

P. Imkeller, G. Dos Reis, J. Zhang, Results on numerics for FBSDE with drivers of quadratic growth, *Contemporary Quantitative Finance (Essays in Honour of Eckhard Platen)*, (2010).

H. Johnson, Options on the maximum or the minimum of several assets, *J. Fin. Quant. Analysis* **22** (1987) 277 - 283.

J.-P. Lemor, E. Gobet, X. Warin, Rate of convergence of an empirical regression method for solving generalized backward stochastic differential equations, *Bernoulli* **12** (2006) 889 - 916.

F. A. Longstaff, R. S. Schwartz Valuing American Options by Simulation: A Simple Least-Squares Approach. *Review of Financial Studies* **14** (2001) 113-147.

J. Ma, P. Protter, J. San Martín, S. Torres, Numerical method for backward stochastic differential equations, *Ann. Appl. Probab.* **12** (2002) 302 - 316.

J. Ma, J. Shen, Y. Zhao, On numerical approximations of forward-backward stochastic differential equations, *SIAM J. Numer. Anal.* **46** (2009) 2636 - 2661.

J. Ma, J. Zhang, Representations and regularities for solutions to BSDEs with reflections, *Stochastic Process. Appl.* **115** (2005) 539 - 569.

G. N. Milstein, M. V. Tretyakov, Numerical algorithms for forward-backward stochastic differential equations, *SIAM J. Sci. Comput.* **28** (2006) 561 - 582.

A. Richou, Numerical simulation of BSDEs with drivers of quadratic growth, *Ann. Appl. Probab.*, forthcoming (2010).

J. Zhang, A numerical scheme for BSDEs, *Ann. Appl. Probab.* **14** (2004) 459 - 488.