Moving Window Asian Options: Sparse Grids and Least-Squares Monte Carlo

Stefan Dirnstorfer **Andreas J. Grau** **Rudi Zagst**
dirnstor@web.de grau@ma.tum.de zagst@ma.tum.de

HVB-Institute for Mathematical Finance
Technical University of Munich

January 2006

Abstract

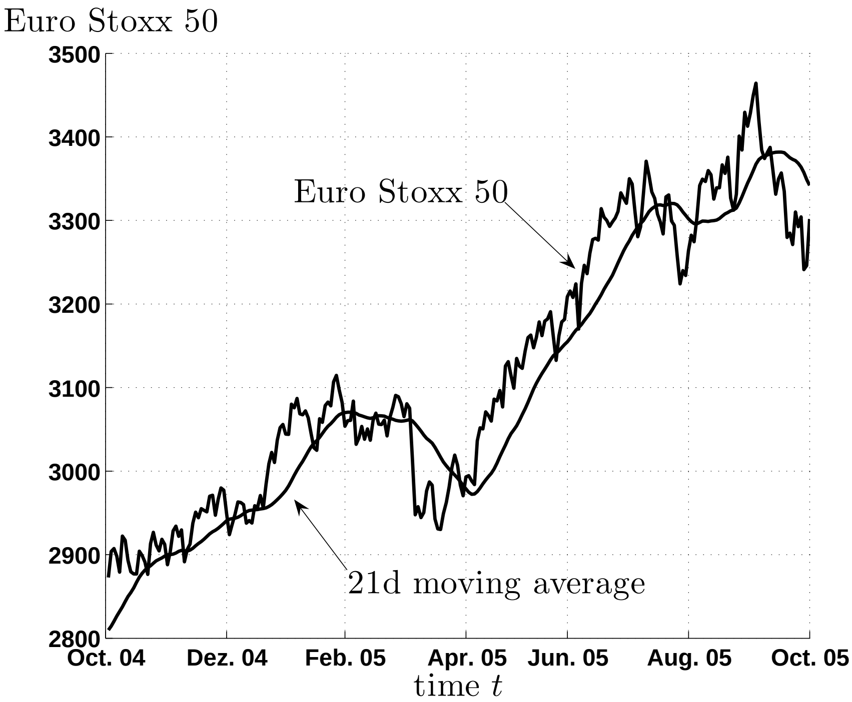
The pricing of moving window Asian options with an early exercise feature is considered as one of the most complex problems in numerical finance. The computational challenge is created by the unknown optimal exercise strategy and the high dimensionality that is required for its approximation. We use the Least Squares Monte Carlo approach together with Sparse Grid type basis functions to combine two simple and well established methods. The resulting algorithm provides a convergent and practical method for pricing the moving window Asian option as well as other high-dimensional, exercisable securities, which to our knowledge have not yet been solved with reasonable accuracy.

1 Introduction

Methods for pricing a large variety of exotic options have been developed in the past decades. Still, the pricing of high dimensional American style options remains challenging. The price of this kind of option depends on the complete price path not only on the stock price at the final exercise date. In this paper, we consider the price of a moving window Asian option (MWAO) with discrete and continuous observations for the computation of the exercise value. The exercise value of the MWAO depends on the average value of the underlying stock over a moving period of time, which means that a continuous observation leads to an infinite dimensional problem.

The idea of computing a moving average value comes from the technical analysis of stock price evolution: Chart analysts use the moving average as an indicator for future stock price movements and they often present charts similar to Figure 1. The figure shows a stock-price index and the corresponding moving average. The analysts claim that there is information about the future in such charts. However, we will not discuss wether this is true or not, we will use the moving average in a different setting, as a strike of stock options. This idea is simple and leads to a product which is easy to understand for investors. But, only a few options which have a moving average

Figure 1 *Although being a popular tool for chart analysts, pricing options on a moving average is challenging.*



as a strike or as an underlying are actively traded [15]. More common is the moving average computation in issuer-call features of some fixed income securities [16]. Our algorithm can easily be adapted to these securities, so that we will only present the simple case of MWAOs.

The foundation of almost any option pricing method is layed by the no-arbitrage framework introduced by Black and Scholes [6]. Several numerical methods have been developed within this framework, as it allows to represent the price of an option by it's expected payoff under the martingale measure. The first simulation technique for option pricing dates back to a publication by Boyle [8] but it took several years until a practical method was developed for an effective simulation-based American option valuation. The method is known as Least Squares Monte Carlo (LS Monte Carlo) since it involves least squares regressions to determine the optimal exercise price. This method was first introduced by Carriere [11] and a similar, but simpler method was presented by Longstaff and Schwartz [18]. Longstaff and Schwartz also presented an example of a moving window Asian option with early exercise. However, the option priced by their mathematical formula solves a much easier problem than indicated by their prosa. Another application of the LS Monte Carlo to the MWAO option is presented by Bilger [5]. His method is very limited and computationally extremely expensive. Accurate values can hardly be expected. However, Bilger's approach is closely related to our method, which only uses a different choice of basis functions for the conditional expected option value.

There are virtually no analytic pricing formulas known for American type options so that one has to rely on numerical methods, of which Monte Carlo simulation is among the most common. Alternative approaches are based on the Cox Ross Rubinstein (CRR) [12] binomial tree model, which can easily be adapted to American Asian options by using non-recombining trees. The size of non-recombining trees grows exponentially with the number of time steps, such that accurate results are hardly obtained. Window options in a recombining CRR model have been presented by Lau and Kwok [17] using forward shooting grids but they price Parisian or delayed-barrier options and not averaging options. Zvan, Forsyth and Vetzal present PDE methods for continuously [25] as well as for discretely sampled Asian options [26]. The averaging period in their model is limited to a start at a fixed point in time and cannot be easily adapted to a moving averaging period. Other authors like Wilmott [23] present the MWAO with early exercise as a challenging ("not easy") problem in a PDE framework.

In fact, pricing methods for MWAOs have been described by very few authors besides Longstaff and Schwartz [18] or Bilger [5]. Kao and Lyuu [15] present results for moving average-type options which are traded in the Taiwan market. Their method is based on the CRR model and can handle short averaging periods: the examples include up to 5 discrete observation in the averaging period.

Related to the MWAOs is the problem of multi-asset Asian options. An interesting approach using Markov transition matrices on low distortion grids has been presented by Berridge and Schumacher [4]. Their method seems to be promising for problems with medium dimensionality (4 to 10) and should be applicable to moving window Asian options. An implementation of their method is much more complex and less flexible than ours. Work on European Asian option contracts has been conducted by several authors, e.g. Kemna and Vorst [16] and Shao and Roe [20].

As the main extension to LS Monte Carlo we propose the utilization of sparse grids type basis functions in the regression, which allows for an accurate option valuation of up to 20 discrete observations on prevailing hardware. The idea of this technique was originally discovered by Smolyak [21] and was rediscovered by Zenger [24] for PDE solutions in 1990. It has been applied to many different topics since then, such as integration [7] or Fast Fourier Transformation [14]. Recently, sparse grids have been used for finite element PDE solutions by Bungartz [9], interpolation by Bathelmann et al [3], and clustering by Garcke et al [13]. They also have been applied to PDE option pricing by Reisinger [19]. An extensive overview of sparse grid methods is provided by Bungartz and Griebel [10].

This paper is structured as follows: First we formulate the problem of moving window Asian option pricing and explain why it is computationally challenging. It follows a brief description of the Least-Squares Monte Carlo and the introduction of sparse grids to the framework. We show some numerical examples that demonstrate the method's effectiveness. Finally we apply an extrapolation technique to further reduce the error originating from the discrete observations and other limiting parameters.

2 Moving Window Asian Option

In this section, we work out the details of a moving window Asian option and present some similar derivatives. The MWAO is a simple option that makes use of the moving average as it is plotted in many stock price charts. Similar to an American option which pays the difference between the current underlying price and a fixed strike, the MWAO pays the difference between the current stock price and the floating moving average. Since the computation of moving averages is well established in chart analysis, this option could be accepted by the market, despite its computational difficulties. Having derived a precise mathematical formulation for the price of an MWAO, we will be able to understand its computational challenge. Other securities which seem to be equally challenging at first sight are already very common and actively traded. We will show, how the valuation of the related securities avoid the computational difficulties of MWAOs. However, MWAOs might be more interesting for investors than the related securities because they have a more intuitive averaging mechanism.

2.1 Continuous version

Before we go into the details of the financial product we set up the process for the underlying variable. As usual in option pricing, we use a standard diffusion process that models the uncertainty in the stock price, according to the formula of Black and Scholes. We denote the stock price at time t with S_t and the option price in dependence of S_t with either V_t or $V_t(S_t)$. From the no-arbitrage arguments we know that the option value satisfies the partial differential equation,

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S_t^2 \frac{\partial^2 V}{\partial S_t^2} + rS_t \frac{\partial V}{\partial S_t} - rV = 0 \tag{1}$$

with risk free interest rate r.

Now, the peculiarity of the MWAO is expressed by a boundary condition to the option value V, known as an American constraint. The following condition states the minimum value for the function V and has to be satisfied at each time $t > t_0 + t_w$,

$$V(t,S) \geq P(A_t, S_t), \tag{2}$$

$$A_t = \frac{1}{\int_0^{t_w} \alpha(\tau) d\tau} \int_{t-t_w}^{t} \alpha(t-\tau) S_\tau d\tau, \tag{3}$$

where P is the option's payoff function that depends on the current stock price S_t and a weighted average A_t of the historic stock prices using the weight function α. The moving average is taken over a window ranging from $t - t_w$ to t. In the following, we will consider the payoff function

$$P(A, S_t) = \max\{A_t - S_t, 0\}. \tag{4}$$

Hence, the exercise value is greater zero if the stock price falls below its moving average. Effectively this is the case if the stock price drops either quickly or steadily.

The standard value for the weight α is

$$\alpha \equiv 1$$

which results in an arithmetic average.

We will discuss other values in section 3. The difficulty in this pricing equation is the boundary condition in equation (2) which depends on the whole history of stock prices S_τ within the averaging period $t - t_w \leq \tau \leq t$. In fact, it is almost impossible to represent this integral numerically, unless we discretize the path of S.

2.2 Discretization

For the computational implementation of this problem we introduce a number of additional variables that contain samples of historic values of S_{t_i} at different times $t_i \in \{t_0 = 0, t_1, \ldots, t_n = T\}$. The accuracy of this approximation depends on the time resolution of the samples. Thus the boundary condition (2) becomes a constraint in terms of historic samples. We assume the last m samples to form the historic window. The condition

$$V_{t_i}(S_{t_{i-m}}, \ldots, S_{t_i}) \geq P\left(\frac{1}{m} \sum_{j=i-m}^{i} \alpha(i-j) S_{t_j}, S_{t_i}\right) \tag{5}$$

holds for $i \geq m$, after an initial incubation. For weight α, we will consider two possible configurations. Since the sample points are used to approximate the integral over the stock price path, we can use the trapezoid method for integration as the preferred method for non-smooth integrands:

$$\alpha_1(t) = \begin{cases} \frac{1}{2} & \text{for } t = 0 \vee t = m \\ 1 & \text{otherwise} \end{cases} . \tag{6}$$

A simpler method is sometimes closer to reality. With a constant α we do not optimally approximate the continuous integral, but might do better at modeling the practical implementation of such an option. In a realistic setting, this option has predefined dates at which the stock price is fixed and considered in an equally weighted arithmetic average. That means we require a weight function α with

$$\alpha_2(t) = \begin{cases} 1 & \text{for } t < m \\ 0 & \text{for } t = m \end{cases} . \tag{7}$$

Our method for the valuation of the option uses the presented discretization and a quadrature of either α_1 or α_2, depending on the setting. The valuation proceeds backwards in time, starting at maturity T, where condition (5) holds with equality. Then, we solve for the option value at current time and current stock price $V = V_{t_0}(S_{t_0})$.

For low values of m this procedure can be rephrased in a PDE setting and solved numerically by standard methods. Without going into details, we recommend a method that is based on a finite volume discretization of the Black-Scholes PDE according to the model of Zvan et al [26]. However, due to the “curse of dimensions” it is traditionally thought that a function with more than three or four dimensions is extremely hard to discretize.

3 Related problems

As we have seen, the moving window Asian option is a derivative with the moving average as one of its underlyings. In order to determine its price correctly, the full history of previous prices has to be considered, which leads to an arbitrary number of relevant dimensions. Despite its intuitive definition the moving average presents a serious computational challenge. This section distinguishes the MWAO from other similar derivatives for which straight-forward implementations or even analytical formulas were derived. Since all the difficulties originate from the averaging mechanism A_t, we will focus on some alternative averaging styles.

3.1 Asian American option

The Asian American option (AAO) is very similar to the moving window Asian option. It differs in the time horizon over which the average is evaluated. While the MWAO has a moving window with constant length, the AAO has a window that increases in time. The averaging window always starts at t_0 and ends at the current time t. This slight difference considerably simplifies the computational procedure. In the following we will briefly show that this pricing problem can be solved in two dimensions.

Consider an asset price process S with an asset price at time t of S_t. The moving average A_t^{AAO} is given by

$$A_t^{AAO} = \frac{1}{t}\int_0^t S_\tau d\tau. \tag{8}$$

Differentiating this expression with resprect to time t, we obtain

$$dA_t^{AAO} = \frac{1}{t}S_t dt - \frac{1}{t}A_t^{AAO} dt \tag{9}$$

which does not depend on any historic stock price. Only the current stock price and the previous average is required.

3.2 Exponential weight

There exists another version of the moving window Asian option for which a good Markovian approximation of the update formula can be constructed. It uses the variable a as a decay factor which determines how much less old stock prices are weighted compared to newer values. Consider an exponentially weighted average for the payoff

$$V(t,S) \geq P(A_t, S_t), \tag{10}$$

$$A_t^{Exp} = \frac{1}{\int_0^t \alpha(\tau) d\tau}\int_0^t \alpha(t-\tau) S_\tau d\tau, \tag{11}$$

with

$$\alpha(t) = a\exp(-at). \tag{12}$$

The average theoretically depends on all previous prices, which makes it difficult to implement in practice. However, a simple update formula is available by differentiation of the expression A_t^{Exp}

with respect to time,

$$dA_t^{Exp} = \left(\frac{a}{1-e^{-at}} (S_t - A_t^{Exp}) \right) dt. \tag{13}$$

Since this special case assigns virtually no weight to very old asset prices, the method can be seen as a rough approximation to the MWAO in equation (2) with $\alpha(t) = a\exp(-at)$. This kind of approximation is presented by Longstaff and Schwartz [18].

3.3 Moving Window Asian Option

The previous paragraphs presented simple update formulas for averages of Asian options. A similar update formula can not be constructed for the MWAO. The complete set of historic asset prices in the window is relevant to the exercise decision of MWAOs.

To see that the problem of the MWAO is different from the presented Asian options, we reconsider the averaging function in equation (3) with a weight function $\alpha = 1$:

$$A_t = \frac{1}{t_w} \int_{t-t_w}^{t} S_\tau d\tau.$$

Differentiating this expression with respect to time t leads to

$$dA_t = \frac{1}{t_w} (S_t - S_{t-t_w}) \, dt,$$

which depends on the asset price at two different times. An optimal exercise strategy has to consider the two values S_t, S_{t-t_w} and all asset prices in between. The reason for this is that all the values $S_{t_i}, t > t_i > t - t_w$ will be used in the computation of future moving averages, which are required in the computation of the expected value of continuation. Since there are infinite many asset prices S_{t_i}, the computation of the optimal exercise strategy is hard.

4 Numerical procedure

The algorithm that is proposed by this paper is effectively combining three individual techniques which are well established in their respective fields. We combine Monte Carlo simulation, least squares regression and sparse grids to a practical method for American option valuation. Especially in quantitative finance the technique called sparse grids does not yet fully live to its potential. One of the purposes of this article is to demonstrate the flexibility and the simplicity of sparse grids. Since all the individual components of our algorithm have been elaborated in full detail by our cited sources, we will just summarize each of the components' main aspects.

4.1 Simulation

A standard method which is used when dimensionality causes numerical difficulties is Monte Carlo simulation. As we will see, this approach does not resolve our issue but will provide the framework for our algorithm. In a Monte Carlo simulation, we simulate different asset paths. Each of these paths follows a geometric Brownian motion. This is also the process underlying equation (1),

$$\mathrm{d}S_t = rS_t\mathrm{d}t + \sigma S_t dW_t$$

with a riskless interest rate r, volatility σ and the increment of a Wiener process $\mathrm{d}W$. This process is sampled at discrete times $t_i \in \{t_0, t_1, \ldots, t_n = T\}$ so that each of the s realization S^j, $j \in \{1, \ldots, s\}$ follows

$$S^j_{t_{i+1}} = S^j_{t_i} \exp\left((r - \frac{1}{2}\sigma^2)(t_{i+1} - t_i) + \sigma\sqrt{(t_{i+1} - t_i)}\phi \right)$$

with ϕ drawn from a standard Normal distribution. The price of the MWAO is the discounted expected value of the payoff at the optimal stopping time. The optimal stopping time provides a strategy maximizing the option value without information about the future of the asset path.

4.2 Least squares

At each exercise time t_i, the holder decides to exercise the option and get the payoff $P(S,t_i)$ or to continue. The variable S denotes a complete asset path but $P(S,t_i)$ may only depend on the values of S_t known at time t_i, i.e. $t \leq t_i$. In order to maximize the option value V, the holder exercises if

$$P(S,t_i) \geq \mathbb{E}_Q[V|S,t_i]$$

with $\mathbb{E}_Q[V|S,t_i]$ denoting the expected option value under the risk-neutral measure Q if the option is not exercised at time t_i. In the LS Monte Carlo approach, the value of $\mathbb{E}_Q[V|S,t_i]$ is approximated by

$$P^e(S,t_i) \approx \mathbb{E}_Q[V|S,t_i].$$

As in section 2.2, we assume that the historic window is formed by the last m samples. The value $P^e(S,t_i)$ is computed using a least square regression on many path-realizations S^j. The regressions start at the time step t_{n-1}, i.e. one step before maturity time T. The approximated values are

$$P^e(S,t_i) = \sum_k a^i_k B_k(S_{t_{i-m}}, \ldots, S_{t_i})$$

with some basis functions B_k and unknown coefficients a_k^i minimizing

$$\left\| \sum_k a_k^i B_k(S_{t_{i-m}}^j, \ldots, S_{t_i}^j) - e^{-r(t_{i+1}-t_i)} V_{t_{i+1}}^j \right\|_2 \tag{14}$$

where $V_{t_{i+1}}^j$ is the estimate of the option value for time t_{i+1} using the Monte Carlo path realization S^j. The value of $V_{t_i}^j$ is given as the maximum between the estimated value of the unexercised option P^e and the intrinsic value P,

$$V_{t_i}^j = \begin{cases} e^{-r(t_{i+1}-t_i)} V_{t_{i+1}}^j & \text{if } P^e(S^j, t_i) > P(S^j, t_i) \\ P(S^j, t_i) & \text{else} \end{cases} .$$

Given that the option value at maturity time equals the payoff $V_T^j = P(S^j, T)$, a dynamic program solves for all values $V_{t_i}^j$, starting at time T and iterating backwards to t_0. Based on the value $V_{t_0}^j$, we can compute an approximation to the option value, which is known as the in-sample price,

$$V^{in} = \frac{1}{s_1} \cdot \sum_{j=1}^{s_1} V_{t_0}^j,$$

where the asset paths are S^j, $j \in \{1, \ldots, s_1\}$. This approach has an obvious disadvantage. Each of the estimated option values $V_{t_0}^j$ contains information about its future asset path S_j. In order to avoid this property, we compute the out-of-sample option price: We generate additional simulation paths S^l, $l \in \{s_1+1, \ldots, s_1+s_2\}$ but use the coefficients a_k^i which were fitted to the old set of simulation paths S^j, $j \in \{1, \ldots, s_1\}$. Consequently, the out-of-sample value can not depend on the knowledge of the future paths. It is given by

$$V^{out} = \frac{1}{s_2} \cdot \sum_{l=s_1+1}^{s_1+s_2} V_{t_0}^l \tag{15}$$

with

$$V_{t_i}^l = \begin{cases} e^{-r(t_{i+1}-t_i)} V_{t_{i+1}}^l & \text{if } P^e(S^l, t_i) > P(S^l, t_i) \\ P(S^l, t_i) & \text{else} \end{cases} , l \in \{s_1+1, \ldots, s_1+s_2\}.$$

Under optimal conditions, the in-sample and the out-of-sample price converge to the correct arbitrage-free price. However, in our computations, we will only compute the out-of-sample value because it is the value for which we can state the optimal exercise policy without information about the future. Furthermore, the expected value of the out-of-sample price V^{out} is always a lower bound for the option value: The crucial point for the convergence of least squares Monte Carlo simulation is the estimate P^e. We are confined to finite many samples and to finite degrees of freedom in the regressions and will not be able to perfectly represent the real shape of P. A less than optimal exercise strategy is performed and provides a reduced option value.

4.3 Choice of basis functions

A tricky part of our numerical solution and in fact the crucial challenge is the careful choice of the basis functions B_k in equation (14). As described in the previous section we will use a linear combination of these basis functions to express an estimate for the current option value in dependence of all relevant input parameters.

The choice of the class of basis functions seems to have little effect on the values computed by least squares Monte Carlo [18]. Our experiments confirm this observation so that we choose the simplest set of basis functions, polynomials. In a straight forward approach we could use the full set B_ℓ^{full} of all m-dimensional polynomials up to a certain polynomial degree $\ell = (\ell_1, \dots, \ell_m)$,

$$B_\ell^{\text{full}}(x_1, \dots, x_m) := \left\{ \prod_{i=1}^{m} x_i^{g_i} \,\middle|\, g_i \in \mathbb{N}_0 \wedge g_i \leq \ell_i \right\}. \tag{16}$$

But, it is easy to see, that this construction by its own will quickly exhaust any available computational resources. A setting with 10 dimensions and a maximal polynomial degree of one in each direction already yields as many as $2^{10} = 1024$ basis functions, over which least squares regression has to be performed. A maximal quadratic polynomial degree in each direction leads to $3^{10} = 59049$ basis functions.

4.3.1 Piecewise Linear Sparse Grids

As already stated, the exponential growth of the number of basis functions of full grids quickly overextends any computer. Fortunately, a much more efficient selection of basis functions can be constructed, known as sparse grids or combination method [21]. This kind of function basis has been successfully applied in the field of high dimensional function approximation [13] and many others.

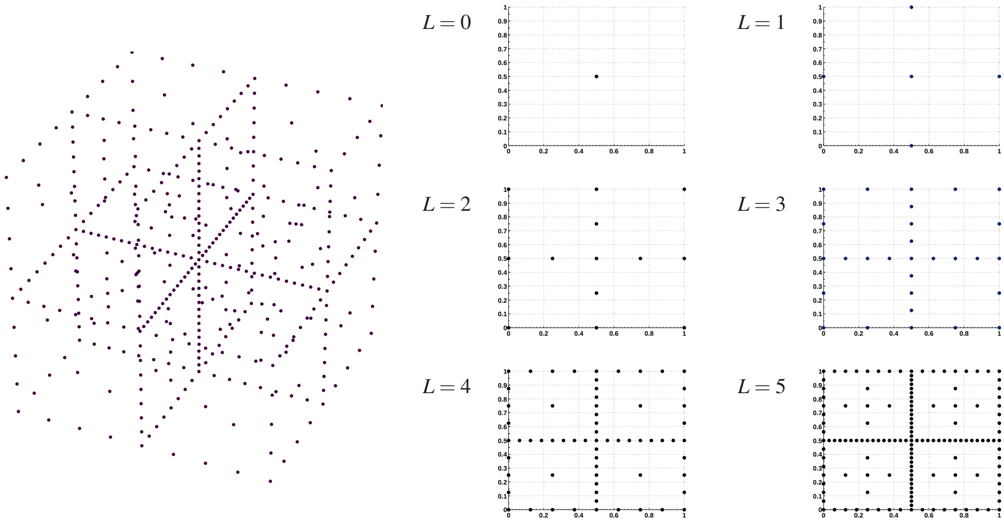
The original idea of sparse grids is based on piecewise linear basis functions which we will call grids. Similar to the full set of m-dimensional polynomials, we define the full grid Ω_ℓ, $\ell = (\ell_1, \dots, \ell_m)$ which has a different equidistant spacing $(2^{-\ell_1}, \dots, 2^{-\ell_m})$ for each dimension of $x = (x_1, \dots, x_m)$ and has grid points in $[0,1]^m$. Usually, for all i with $l_i = 0, i \in \{1, \dots, m\}$, the corresponding component of the grid points is changed to 0.5. The idea of sparse grids is summarized as follows. Instead of using a full grid Ω_ℓ, we combine multiple grids according to a sparse and error optimal scheme Ω_L^{sparse},

$$\Omega_L^{\text{sparse}}(x_1, \dots, x_m) := \bigcup_{\sum \ell_i = L} \Omega_\ell. \tag{17}$$

Instead of defining a multidimensional level ℓ we use the single sparse level L, that limits the sum of all components $\ell = (\ell_1, \dots, \ell_m)$. Figure 2 presents two and three dimensional sparse grids. This kind of combining regular grids has been shown to produce a reasonable sparseness for a wide class of smooth functions [9].

If we compare full and sparse grids, we can see that the computational effort decreases radically while the error rises only slightly: Comparing grids with minimal mesh size $h_L = 2^{-L}$, a full grid has $O(h_L^{-m})$ grid points and a sparse grid only employs $O(h_L^{-1} |\log h_L|^{m-1})$ points. At the same time, the L_2-interpolation error for smooth functions rises from $O(h_L^2)$ to $O(h_L^2 \cdot |\log h_L|^{m-1})$ [10]. In many applications, $L \in \{2, \dots, 4\}$ is already sufficient.

Figure 2 *A three-dimensional sparse grid with $L = 5$ is presented on the left and two-dimensional sparse grids from $L = 0, \ldots, 5$ on the right hand side.*



4.3.2 Sparse Polynomial Basis

The presented sparse grid approach uses piecewise linear basis functions supporting the grid nodes. In our example of the moving window Asian option, the function we want to estimate has the property of being very smooth. As we found that differentiable basis functions deliver better results, we present an idea which creates a sparse polynomial basis. A detailed analysis for this kind of sparse basis can be found in [3] so that we can focus on the main issues.

We combine the idea of a polynomial basis with the idea of sparse grids: instead of using a plain polynomial basis B_ℓ^{full} we combine the multiple polynomial orders according to the same sparse and error optimal scheme as for the sparse grids. The sparse basis B_L^{sparse},

$$B_L^{\text{sparse}}(x_1, \ldots, x_m) := \bigcup_{\sum \ell_i = L} B_{\beta(\ell)}^{\text{full}} \tag{18}$$

has many of the properties of the sparse grid with piecewise linear basis functions but it is smooth everywhere.

The polynomial basis sparse level L again limits the sum of all components $\ell = (\ell_1, \ldots, \ell_m)$. Furthermore, the degree of the combined full polynomial bases is transformed by a mapping function β that turns each level into a maximum polynomial degree

$$\beta(\ell) = 2 \cdot (2^{\ell_1} - 1, \ldots, \ldots, 2^{\ell_m} - 1). \tag{19}$$

This transformation cannot be applied to grids because a grid with $\ell_i = 0$ nodes in the ith dimension makes no sense. But, for polynomials, this reduces the number of basis functions.

Example: A sparse polynomial basis with $m = 3, L = 2$

First, we have to compute the full basis polynomials of equation (16). The sparse level $L = 2$ requires the computation of

$$
\begin{aligned}
\ell &= (2,0,0) \rightarrow B^{\text{full}}_{6,0,0} = && \{1, x_1, x_1^2, x_1^3, x_1^4, x_1^5, x_1^6\} \\
\ell &= (1,1,0) \rightarrow B^{\text{full}}_{2,2,0} = && \{1, x_1, x_2, x_1x_2, x_1^2, x_2^2, x_1^2x_2, x_1x_2^2, x_1^2x_2^2\} \\
\ell &= (1,0,1) \rightarrow B^{\text{full}}_{2,0,2} = && \{1, x_1, x_3, x_1x_3, x_1^2, x_3^2, x_1^2x_3, x_1x_3^2, x_1^2x_3^2\} \\
\ell &= (0,2,0) \rightarrow B^{\text{full}}_{0,6,0} = && \{1, x_2, x_2^2, x_2^3, x_2^4, x_2^5, x_2^6\} \\
\ell &= (0,1,1) \rightarrow B^{\text{full}}_{0,2,2} = && \{1, x_2, x_3, x_2x_3, x_2^2, x_3^2, x_2^2x_3, x_2x_3^2, x_2^2x_3^2\} \\
\ell &= (0,0,2) \rightarrow B^{\text{full}}_{0,0,6} = && \{1, x_3, x_3^2, x_3^3, x_3^4, x_3^5, x_3^6\}
\end{aligned}
$$

for the sparse grid basis and leads to 31 basis functions,

$$
\begin{aligned}
B^{\text{sparse}}_2(x_1, x_2, x_3) &= \bigcup_{\sum \ell_i = 2} B^{\text{full}}_{\beta(\ell)} \\
&= \{1, x_1, x_2, x_3, x_1x_2, x_1x_3, x_2x_3, x_1^2, x_2^2, x_3^2, x_1^2x_2, \\
&\quad x_1x_2^2, x_1^2x_2^2, x_1^2x_3, x_1x_3^2, x_1^2x_3^2, x_2^2x_3, x_2x_3^2, x_2^2x_3^2, \\
&\quad x_1^3, x_2^3, x_3^3, x_1^4, x_2^4, x_3^4, x_1^5, x_2^5, x_3^5, x_1^6, x_2^6, x_3^6\}.
\end{aligned}
$$

4.3.3 Implementation

In our implementation, we perform the regressions required by equation (14) on sparse polynomial basis functions as presented in the previous paragraphs. We use sparse levels L from 0 to 3 which are sufficient for our purposes. But, we do not perform the regressions on S directly. Instead, we use scaled values of S such that for each path j, we compute $x^j = (\gamma_1 S^j_{t_i}, \ldots, \gamma_m S^j_{t_{i-m}})$ which lie in a unit cube, $x^j \in [0,1]^{m+1}$. The regression itself is performed solving the linear least squares problem of equation (14) implicitly via QR-decompositions [2]. Furthermore, the regression is only performed on the paths with a positive exercise value $S^i : P(S^i, t) > 0$. This decreases the computational effort.

5 Numerical Examples

In order to demonstrate the efficiency of our approach, a numerical case study is provided in this final section. We will focus on a discretely sampled MWAO with properties sketched in Table 1. The option is sampled with a regular frequency, e.g. every trading day at a specified time. We will distinguish between two different sample techniques. The first one has a discretely sampled averaging window spanning ten observations and is consequently integrated with α_2 from equation (7). The second one is aimed at an approximation of the continuous-time version of the MWAO and is integrated with α_1 from (6).

Table 1 *Specifications of a moving window Asian option with a floating strike in discrete time.*

Option type	moving window Asian option
Maturity T	0.4 years
Risk free rate r	5%
Volatility σ	0.40
Dividend rate d	0
Daily observations $\Delta_{obs}t$	1/250 years
Length of observation period m	10 days
Exercise value	$P(S,t_i) = \max\left(\frac{1}{m}\left(\sum_{j=i-m+1}^{i} S_{t_j}\right) - S_{t_i}, 0\right)$

5.1 Convergence

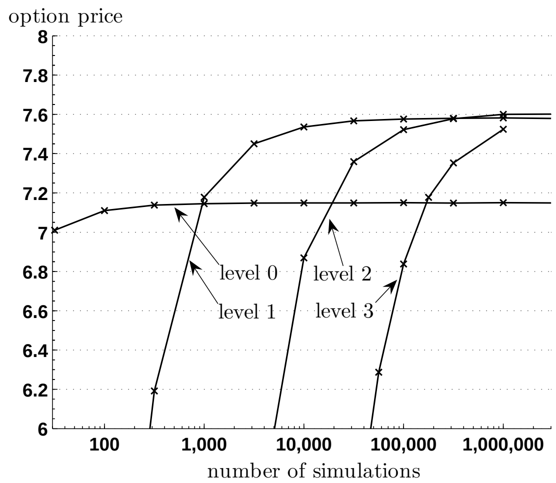
To analyze the convergence of the presented pricing algorithm for MWAOs, we will denote the computational result of V^{out} according to equation (15) by $\widetilde{V_a}^i(n,L,m)$. Thus, each Monte Carlo value $\widetilde{V_a}^i$ depends on the number of samples n, the level of the sparse grid function basis L, the number of observations in the window m and the quadrature scheme α_a. We compute different $\widetilde{V_a}^i(n,L,m)$ with n, L, a and m fixed in order to get an estimate for the mean

$$\overline{V_a}(n,L,m) = \frac{1}{I}\sum_{i=1}^{I} \widetilde{V_a}^i(n,L,m) \tag{20}$$

of I different Monte Carlo prices. The values for I range from 10 to 1000 depending on an estimate of the Monte Carlo error. The number of samples n per Monte Carlo price results from the in-sample paths $S^1,\dots,S^{s_1}$ and the out-of-sample paths $S^{s_1+1},\dots,S^{s_2}$, i.e. $n = s_1 + s_2$. We use 30% of the sample paths for regressions (s_1) and 70% for valuation out-of-sample (s_2).

Figure 3 presents the mean $\overline{V_2}(n,L,10)$ for different numbers of samples n and different levels L. The values at level $L = 1$ converge quickly to a value of about $\overline{V} = 7.15$ which does not change after 3000 simulations. Using $m = 10$, the level 0 consists of just one basis function and

Figure 3 *The option value of an MWAO option with data in Table 1 estimated by LSMC.*



the resulting exercising decision is almost trivial. Level 1 consists of 21 basis functions. This allows for a more sophisticated strategy with a better utilization of the option. After about 100.000 simulations, the option value saturates at 7.58. The level 2 with 241 basis functions results in an even higher value of $\overline{V} = 7.60$ after 1.000.000 simulations. A third level with 2001 basis functions already exceeds our available computational resources, such that the saturation level could not be computed. One thing worth mentioning is the initial inferiority of higher levels due to an over-fitted exercise strategy.

Table 2 *The option value of an MWAO option with data in Table 1 estimated by LS Monte Carlo. The mean of a series of evaluations with level L and a fixed # of samples is denoted by* $\overline{V_2}(n,L,10)$ *where as the standard deviation* $\sigma(\widetilde{V_2}^i(n,L,10))$ *of this series is denoted by* $\widehat{\sigma}$.

\ **level** L	$L=0$		$L=1$		$L=2$		$L=3$	
# samples n	$\overline{V_2}(n,1,10)$	$\widehat{\sigma}$	$\overline{V_2}(n,2,10)$	$\widehat{\sigma}$	$\overline{V_2}(n,3,10)$	$\widehat{\sigma}$	$\overline{V_2}(n,4,10)$	$\widehat{\sigma}$
3×10^1	7,010	0,499						
1×10^2	7,110	0,234	4,053	0,470				
3×10^2	7,138	0,134	6,192	0,264				
1×10^3	7,145	0,073	7,178	0,114	3,813	0,148		
3×10^3	7,148	0,043	7,450	0,061	5,399	0,083		
1×10^4	7,149	0,022	7,536	0,034	6,869	0,041	3,166	0,069
3×10^4	7,149	0,014	7,567	0,018	7,359	0,018	5,357	0,024
1×10^5	7,150	0,007	7,576	0,010	7,522	0,009	6,841	0,010
3×10^5	7,148	0,005	7,580	0,005	7,578	0,006	7,358	
1×10^6	7,150	0,004	7,582	0,003	7,600	0,002	7,524	
3×10^6	7,149	0,001	7,579	0,001	7,601	0,001		

The corresponding values to Figure 3 are presented in Table 2. The mean values of a series of valuations is denoted by $\overline{V_2}(n,L,10)$, the standard deviation of the series is denoted by $\widehat{\sigma}$. For a single evaluation with the LS Monte Carlo, $\widehat{\sigma}$ can be seen as a measure how close the value is to the mean of many valuations. Contrarily, $\widehat{\sigma}$ does not provide a measure for the error compared with the real value. The mean estimate will be biased towards lower than the real values due to the insufficient estimate of the optimal exercise strategy P^e. An approximation of the MWAO with 1.000.000 sample paths and level two regressions delivers cent accurate estimates. The value of an option with properties in Table 1 is 7.60.

5.2 Heuristic Extrapolation

After we have successfully handled the ten-dimensional case we will aim for the infinitely dimensional problem. While the previous option's exercise value depends on the average of ten discretely sampled stock prices we will now consider the continuous integral. The option has a payoff as defined in equation (4). In order to present the optimal approximation we will increase the number of samples in the averaging window and then extrapolate the value based on the obtained convergence properties. The derivative's specification can be found in Table 3.

Table 3 *Specifications of a moving window Asian option with a floating strike in continuous time.*

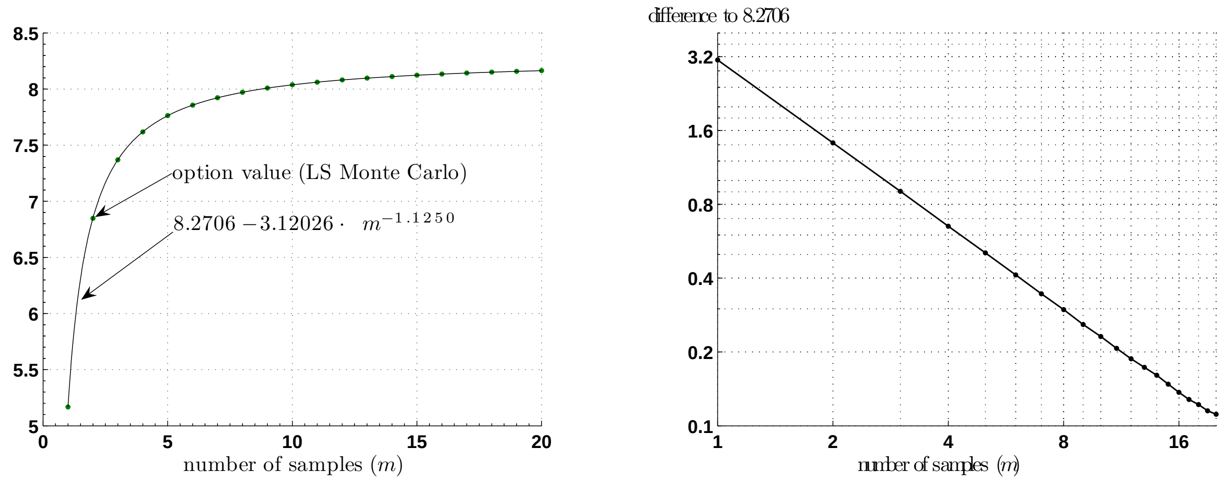
Option type	moving window Asian option
Maturity T	0.4 years
Risk free rate r	5%
Volatility σ	0.40
Dividend rate d	0
Averaging window length t_w	10 days = $\frac{10}{250}$ years
Exercise value	$P(S,t) = \max\left(\frac{1}{t_w}\left(\int_{t-t_w}^{t} S_\tau d\tau\right) - S_t, 0\right)$

For the continuous integral we rely on the trapezoidal quadrature rule α_1 as stated in (6). Hence, we compute $\overline{V_1}(n,L,m)$ and analyze the effect of increasing m arbitrarily. Figure 4 demonstrates the convergence on level $L = 2$ with 10^6 sample path's. We can clearly recognize the convergence with the number of observation samples within the averaging window m. Despite the converging shape there is still some slope in the curve's final point $\overline{V_1}(10^6,2,20) = 8.16$. Extrapolation will lead us to a final result that is about 2% higher than our best finite approximation.

In order to approximate the infinite-dimensional result as accurate as possible we use an extrapolation technique for the Least-Squares Monte Carlo, similar to the Richardson extrapolation [1]. For an extrapolation, we require a convergent, strictly increasing series of option values. Assuming that we knew the order of convergence of the error, we could extrapolate to infinity and solve for the value of the continuously averaging MWAO. Before going into mathematical details we can think of this method as a way of guessing the limit value based on the information known.

The price of our continuous average option has three main sources of error: the number of simulation paths n, the level of the function basis L and the number of integration samples m. The best possible approximation would have to limit each of these parameters towards infinity

Figure 4 *The mean option value* $\overline{V_1}(10^6,1,m)$ *of an MWAO option with data in Table 3 estimated by LS Monte Carlo together with the function* $8.2706 - 3.12026 \cdot m^{-1.1250}$ *is presented. The plot on the right shows the difference of the option values to the extrapolated value,* $\widetilde{V}_1(10^6,1,\infty) = 8.2706 - \overline{V_1}(10^6,1,m)$.



and compute $\overline{V}(\infty,\infty,\infty)$. We will limit our discussion to $L \in \{0,1,2\}$ because this should already give values accurate enough. Furthermore, multidimensional extrapolation is certainly something where little experience has been collected so far. We will do extrapolation not only as a mental exercise, but also as a way to justify our finite results which are very close to the presumable infinite limit.

To our knowledge, there has not been any theoretical error analysis of the LS Monte Carlo for MWAOs. But, we can build on a result from Stone [22]: If a regression function $\theta(x) = \mathbb{E}[Y|X=x]$ is p-smooth, then the L_2-error of a local polynomial kernel estimator converges to zero at a rate of n^{-c} with some fixed value c and n denoting the number of samples of X. This result is related to the LS Monte Carlo valuation because we use a polynomial basis in order to estimate the conditional expectation and we assume that this is the main source of error. Hence, we can rewrite $\overline{V_a}(n,L,m)$ as

$$\overline{V_a}(n,L,m) \approx \overline{V_a}(\infty,L,m) - (c_0 n^{-c_1}) \tag{21}$$

and our empirical data analysis indicates that this is a reasonable guess.

This extrapolation to $n \to \infty$ has litte impact, since values based on $n = 10^6$ are already very precise. The difference between $V(10^6,1,m)$ and $V(\infty,1,m)$ is just about one cent.

Being able to produce a series of $\overline{V_1}(\infty,L,m)$ we can continue and focus on the next parameter m. If we want to extrapolate it to infinity, we again have to find the order of the error. We look for a reasonable function which can fit the option values for different m. Figure 4 presents the LS Monte Carlo option values for different m together with the function $8.2706 - 3.12026 \cdot m^{-1.1250}$. The fit of the function is almost perfect so that we get $V(10^6,1,\infty) \approx 8.27$. The same procedure for $L=2$ leads to $V(10^6,2,\infty) \approx 8.30$.

In the end, we can present an informed guess for the value V of a continuously averaging

MWAO with properties in Table 3:

$$V \approx 8.30. \tag{22}$$

This kind of extrapolation can be useful to decrease computational effort or to increase accuracy. However, the correct description of the error and in particular the suitability of the error estimators for the parameters is open for future research.

6 Conclusion

This paper presents a simple and flexible implementation of a moving window Asian option. Despite being actively traded, no accurate algorithm has been published so far which could extract the derivative's optimal exercise strategy and its precise value. The computational difficulty stems from one of the options underlyings: an either discretely or continuously sampled moving average over a stock price path. This leads to a very high dimensionality in the mathematical definition with an exponential complexity in standard algorithms. We have shown that a straight forward approach to this problem could be found by employing Least-Squares Monte Carlo and a technique called sparse grids, which was specifically developed as a cure to the curse of dimension. The presented approach can be applied to any derivative that has the moving average as an underlying, as it is commonly plotted in stock price charts. We believe that this paper can increase the acceptance of such products. Now, there is a simple algorithm for the simple derivative.

References

[1] Forman S. Acton. *Numerical Methods that Work*, page 106. The Mathematical Association of America, August 1997.

[2] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen. *LAPACK User's Guide*, chapter Linear Least Squares (LLS) Problems. SIAM, Philadelphia, 3rd edition, 1999.

[3] Volker Bathelmann, Erich Novak, and Klaus Ritter. High dimensional polynomial interpolation on sparse grids. *Advances in Compuational Mathematics*, 12:273–288, 2000.

[4] Steffan John Berridge and J.M. (Hans) Schumacher. Pricing high-dimensional american options using local consistency conditions. *CentER Discussion Paper*, (2004-19), 2004.

[5] Ralph Bilger. Valuing American-Asian options using the Longstaff-Schwartz algorithm. Msc thesis in computational finance, Oxford University, 2003.

[6] Fischer Black and Myron Scholes. The pricing of options and corporate liabilities. *Journal of Political Economy*, 81:637–659, 1973.

[7] T. Bonk. A new algorithm for multi-dimensional adaptic numerical quadrature. In W. Hackbush, editor, *Proceedings of the 9th GAMM-Seminar, Kiel 1993*, Braunschweig, January 1994. Vieweg.

[8] Phelim P. Boyle. Options: A Monte Carlo approach. *Journal of Financial Economics*, 4(3):323–338, 1977.

[9] Hans-Joachim Bungartz. Higher order finite elements on sparse grids. *Electronic Transactions on Numerical Analysis*, 6:63–77, December 1997.

[10] Hans-Joachim Bungartz and Michael Griebel. Sparse grids. *Acta Numerica*, pages 147–269, 2004.

[11] Jacques F. Carriere. Valuation of the early-exercise price for options using simulations and nonparametric regression. *Insurance: Mathematics and Economics*, 19:19–30, 1996.

[12] John Cox, Stephen Ross, and Mark Rubenstein. Option pricing: A simplifyed approach. *Journal of Financial Economics*, 7:229–263, 1979.

[13] J. Garcke, M. Griebel, and M. Thess. Data mining with sparse grids. *Computing*, 67(3):225–253, October 2001.

[14] K. Hallatschek. Fouriertransformation auf dünnen Gitttern mit hierarchischen Basen. *Numer. Math.*, 63:83–97, 1992.

[15] Chih-Hao Kao and Yuh-Dauh Lyuu. Pricing of moving average-type options with applications. *Journal of Futures Markets*, 23(5):415–440, 2003.

[16] A.G.Z. Kemna and A.C.F. Vorst. A pricing method for options based on average asset values. *Journal of Banking and Finance*, 14:113–129, 1990.

[17] Ka Wo Lau and Yue Kuen Kwok. Pricing algorithms for options with exotic path dependence. *Journal of Derivatives*, pages 28–38, Fall 2001.

[18] Francis A. Longstaff and Eduardo S. Schwartz. Valuing American options by simulation - a simple least-squares approach. *The Review of Financial Studies*, 14(1):113–147, 2001.

[19] Christoph Reisinger. *Numerische Methoden für hochdimensionale parabolische Gleichungen am Beispiel von Optionspreisaufgaben*. Dissertation, Ruprecht-Karls-Universität, Heidelberg, June 2004.

[20] Renyuan Shao and Brian Roe. The design and pricing of fixed- and moving window contracts: An application of Asian-basket option pricing methods to the hog-finishing sector. *The Journal of Futures Markets*, 23(11):1047–1073, 2003.

[21] S. A. Smolyak. Quadrature and interpolation formulas for tensor products of certain classes of functions. *Dokl. Akad. Nauk SSSR*, 148:1042–1043, 1963.

[22] C. J. Stone. Optimal rates of convergence for nonparametric regression. *Annals of Statistics*, 10(4):1040–1053, 1982.

[23] Paul Wilmott. *Paul Wilmott on Quantitative Finance*. John Wiley & Sons Ltd., West Sussex, England, 2000.

[24] Christoph Zenger. Sparse grids. In W. Hackbusch, editor, *Notes on Numerical Fluid Mechanics*, volume 31, Braunschweig, 1991, 1990. Vieweg.

[25] Robert Zvan, Peter A. Forsyth, and Kenneth R. Vetzal. Robust numerical methods for PDE models of Asian options. *Journal of Computational Finance*, 1:39–78, 1998.

[26] Robert Zvan, Peter A. Forsyth, and Kenneth R. Vetzal. Discrete Asian barrier options. *Journal of Computational Finance*, 3:41–68, 1999.