The Sparse Grid Combination Method for Multidimensional Black-Scholes Partial Differential Equations

Ruining Wu and Christina C. Christara

Abstract A fundamental challenge in the numerical solution of multidimensional partial differential equations (PDEs) is the exponential increase of the number of unknowns as the dimensions increase, known as the curse of dimensionality. This creates difficulties in obtaining accurate solutions for even moderate-dimension problems. Researchers have used the sparse grid combination method to reduce the computational workload for multidimensional PDEs. To resolve the unstable order of convergence that arises from the nonsmooth initial conditions, typical of finance problems, researchers applied smoothing (in Fourier space) or, in the case of a Basket option, a clever coordinate transformation that aligns the line of discontinuity with a coordinate axis. We view the transformation through the lens of quantization error, which allows us to determine the minimum order of smoothing required for restoring the order of convergence, and to highlight certain features of smoothing techniques. Additionally, we present numerical results for American options and for options with payoffs that cannot be transformed to align their nonsmoothness regions with a coordinate axis.

Key words: computational finance, curse of dimensionality, sparse grids

1 Introduction

There are several option pricing problems described by multidimensional PDEs. The most basic form of the Black-Scholes equation [1], that models the price of a contingent claim on a single underlying asset, is one-dimensional (involves one spatial variable). One source of additional dimensions comes from multiple assets.

Ruining Wu
Department of Computer Science, University of Toronto e-mail: rwu@cs.toronto.edu

Christina C. Christara
Department of Computer Science, University of Toronto e-mail: ccc@cs.toronto.edu

There are options written on arbitrarily many assets, such as basket options, geometric average options, and more.

To mitigate the curse of dimensionality that arises from numerically solving multidimensional PDEs, researchers have used the sparse grid combination method [5], which uses substantially fewer degrees of freedom compared to the full grid method, and hence is more computationally efficient, at the cost of a slightly deteriorated order of convergence from $\mathcal{O}(h^\beta)$ to $\mathcal{O}(h^\beta(\log h)^{d-1})$, for β-order methods. Sparse grid methods have originally been developed for multidimensional numerical integration [14]. The success of the sparse grid combination method relies heavily on smoothness requirements, which are not normally satisfied in computational finance, as the typical initial conditions (ICs) exhibit discontinuities. Some recent works on the pricing of European options use high-order methods [6] with a grid transformation [9], which aligns the plane of discontinuity for basket options with a coordinate axis, reducing the smoothness requirements to one dimension, and simplifying smoothing of the ICs when required. Additionally, in [7], no coordinate transformation is applied, and a more general multidimensional smoothing [8] is applied to restore the order of convergence. In our work, we

- compare various smoothing techniques with or without the coordinate transformation, and highlight certain features of each,
- demonstrate, through numerical examples, that the theory of quantization error [2] for the one-dimensional problem can be applied to certain multidimensional cases to explain smoothness requirements of the ICs,
- apply the sparse grid combination method to a nonlinear PDE arising from an American option pricing problem with payoff depending on both dimensions, and
- provide examples where discontinuities in the payoff cannot be aligned with a coordinate axis, and show how averaging [11] restores the order of convergence.

2 Problem Formulation

The financial options we consider in this paper are under the Black-Scholes model. For European options, the PDE that governs the price of the option is given by

$$V_\tau = \mathscr{L}V \equiv \frac{1}{2}\sigma^2 S^2 V_{SS} + rSV_S - rV, \tag{1}$$

where V denotes the option price that we are solving for, S the price of the underlying asset, τ the reverse time counted from expiry T ($\tau = T - t$, t is forward or real time), σ the volatility, r the interest rate, and subscripts denote partial derivatives. In d dimensions, with d assets, the Black-Scholes PDE is given by

$$V_\tau = \mathscr{L}V \equiv \frac{1}{2}\sum_{i,j=1}^{d} \rho_{i,j}\sigma_i\sigma_j S_i S_j V_{S_i,S_j} + \sum_{i=1}^{d} rSV_{S_i} - rV, \tag{2}$$

where S_i and σ_i denote the price and volatility of the i-th asset, and $\rho_{i,j}$ denotes the correlation between S_i and S_j.

Payoff functions correspond to ICs. There are many different payoffs in d dimensions; we use the Basket Put, given by $V(0,S_i) = \max(K - \sum w_i S_i, 0)$, the Min-Put, given by $\max(K - \min(S_i), 0)$, and the Digital (or binary) Basket Call, given by $\mathbf{1}(w_i S_i - K > 0)$, where K is known as the *strike price*, w_i are the weights, conventionally taken to sum to 1, and $\mathbf{1}$ is the indicator function. Note that Basket Put and Min-Put payoffs are $\mathscr{C}^0$, while Digital is $\mathscr{C}^{-1}$. We are interested in the option values at $\tau = T$ and $S_i \approx K$, as this is considered the interesting region to practitioners.

2.1 American Options

In contrast to European options, American options can be exercised at any time up to expiry. This early exercise right gives rise to a linear complementarity problem, which we solve by replacing with the nonlinear PDE [4]

$$V_\tau = \mathscr{L}V + \rho \max(V^* - V, 0), \tag{3}$$

where ρ is the reciprocal of the desired accuracy ε for solving with a penalty iteration algorithm. Solving the American PDE (3) using second-order numerical methods results in a deteriorated order of convergence, as shown in [4]. The authors propose to resolve this issue using adaptive timesteps. Further research [13] has shown that a quadratic transformation of the uniform timesteps also restores the order of convergence. We prefer to use the latter in our work due to its simplicity.

3 Sparse Grid Combination Method

The sparse grid combination method [5] (henceforth, combination method) is a discretization method for mitigating the curse of dimensionality. It makes use of existing PDE solvers, each based on a different, anisotropic tensor product discretization. By combining the numerical solutions of the subproblems appropriately, error terms from dimensions which have coarser discretizations are cancelled out, resulting in an accurate approximation to the numerical solution computed by the full grid method. Additionally, since the subproblems have no interdependence on each other, parallelization comes easily.

Consider the case in two dimensions. Let $u_{i,j}$ denote the numerical solution computed by the two-dimensional grid with level i and j in the first and second variables respectively. Each increase by one in level doubles the number of gridpoints in that dimension.

In two dimensions, the combination method solution at grid level q is computed by

$$u_q^c = \sum_{i+j=q+1} u_{i,j} - \sum_{i+j=q} u_{i,j}, \tag{4}$$

and if the method for solving the subproblems admits an error expansion of the form

$$u - u_{i,j} = C_1 h_i^2 + C_2 h_j^2 + D h_i^2 h_j^2 \tag{5}$$

with h_i and h_j the stepsizes in the first and second dimensions, respectively, and where C_1, C_2 and D are bounded above by a constant independent of h_i or h_j, then, with asymptotic error expansions, it can be shown [5] that the error terms on the coarse grids in both dimensions cancel out, and only q-level errors remain. The combination formulas can be generalized to d dimensions, resulting in $\mathscr{O}((\log N)^{d-1})$ subproblems, each of size $\mathscr{O}(N)$ for a total complexity of $\mathscr{O}(N(\log N)^{d-1})$. The combination method solution for the general d-dimensional case is given by

$$u_q^c = \sum_{p=0}^{d-1} (-1)^p \binom{d-1}{p} \sum_{\sum l_i = q+(d-1)-p} u_{l_1,l_2,\ldots,l_d}. \tag{6}$$

In other words, it is a sum of d sums, and the coefficient associated with each term is given by binomial coefficients with alternating signs.

Due to the slightly deteriorated order of convergence, typical orders expected for 2D problems vary from ~ 1.4 to ~ 1.8, while for 3D problems from ~ 1.2 to ~ 1.7, assuming about 7 grid refinements.

The combination method was originally developed for elliptic PDEs. In the application to parabolic PDEs, we apply the appropriate combination formula to the solution computed at the final timestep at the respective levels. The timesteps are chosen to be the same size as that for a full grid method on the same grid level.

4 Numerical methods

In our work, we use second-order accurate methods. Specifically, we use Crank-Nicolson with Rannacher smoothing [12] (CNR) for timestepping and three-point finite difference (FD) formulas on nonuniform grids generated by smooth mappings [3] for spatial discretization. Note that while the nonuniform three-point FD approximation to the second derivative is technically first-order, because we apply it on a smooth nonuniform grid, it can be shown to be second-order accurate. Multidimensional grids are created from tensor products of one-dimensional grids.

For each spatial variable S_i, we truncate the semi-infinite domain $[0,\infty)$ to $[0,S_{i,\max}]$ where $S_{i,\max}$ is chosen sufficiently large such that the accuracy of the solution near the strike is affected minimally. For 2D problems, we set $S_{i,\max} = 8K$, and for 3D problems, we set $S_{i,\max} = 6K$.

In a multidimensional discretization, the straightforward application of CNR timestepping in the combination method may lead to spurious oscillations, so we use smoothing and other techniques described subsequently to remove the nonsmoothness of ICs. While the nonsmoothness of the ICs may also affect the convergence of the full grid method, the effect is less visible, especially on $\mathscr{C}^0$ payoffs.

To have a unique solution to the PDE problem, it is not always necessary to have boundary conditions. For computational purposes, we impose some equations on the boundary. All boundary conditions we impose are of Dirichlet type, and derived by time-discounted payoffs (i.e. $V(\tau,\partial\Omega) = V(0,\partial\Omega)\exp(-r\tau)$).

4.1 Smoothing operators and quantization error

For the smoothing of functions, we turn to their representation in Fourier space. The well-known paper [8] gives smoothing operators for various orders of smoothness; relevant here are the first- and second-order smoothings, denoted by Φ_1 and Φ_2, respectively. While these operators are presented in [8] for uniform grids, we consider the generalization of Φ_1 that appears in [11] and is known as **averaging**, defined as

$$u_s(x_j) = \frac{1}{x_{j+1/2} - x_{j-1/2}} \int_{x_{j-1/2}}^{x_{j+1/2}} u(x)\, dx, \tag{7}$$

where u is the IC, $u_s(x_j)$ is the smoothed function sampled at the gridpoints x_j, $x_{j-1/2}$ is halfway between the points x_j and x_{j-1}, and $x_{j+1/2}$ likewise. It can be seen that u_s is the average value of u in $[x_{j-1/2}, x_{j+1/2}]$, hence the name. The averaging technique can also be seen as a simple approximate extension of Φ_1 of [8] to nonuniform grids. It also generalizes easily to higher dimensions. For example, in two dimensions,

$$u_s(x_j, y_i) = \frac{1}{(x_{j+1/2} - x_{j-1/2})(y_{i+1/2} - y_{i-1/2})} \int_{x_{j-1/2}}^{x_{j+1/2}} \int_{y_{i-1/2}}^{y_{i+1/2}} u(x,y)\, dx\, dy, \tag{8}$$

and higher dimensions are analogous.

Quantization error is loosely defined as the error arising from the placement of the point of nonsmoothness on the discrete grid. In [2], for one-dimensional problems, explicit formulae are developed for the quantization error with various ICs of varying degrees of nonsmoothness. We briefly review some facts directly relevant to this paper. Let $\alpha \in (0,1]$ be the relative placement of the nonsmooth point on the grid. As shown in [2], for Heaviside (digital), the quantization error without smoothing has an $\mathcal{O}(h)$ term with coefficient $\alpha - 0.5$, and an $\mathcal{O}(h^2)$ term with coefficient dependent on α. Thus, aligning the discontinuity with a midpoint in all grid refinements results in stable $\mathcal{O}(h^2)$ convergence. Equivalently, the Φ_1-smoothed Heaviside gives stable $\mathcal{O}(h^2)$ convergence, if α remains constant in all refinements, and the Φ_2-smoothed Heaviside also gives stable $\mathcal{O}(h^2)$ convergence, regardless of α. For ramp (call, put, etc) ICs, the quantization error without smoothing has an $\mathcal{O}(h^2)$ term with coefficient dependent on α. Thus, aligning the discontinuity with a gridpoint (easy to maintain in all grid refinements in 1D) results in stable $\mathcal{O}(h^2)$ convergence. Equivalently, the Φ_1-smoothed ramp function also gives stable $\mathcal{O}(h^2)$ convergence, regardless of α. Note that, in multiple dimensions, maintaining α constant is not easy, if not impossible.

5 Numerical Experiments

To demonstrate the improved efficiency of the methods along with requirements on smoothness, we present numerical experiments for the two-dimensional American Basket Put, two-dimensional Digital Basket Call, and three-dimensional Min-Put options.

In all numerical experiments, the solution of all subproblems takes place in parallel, with enough workers, such that it is possible to distribute only one subproblem to each worker.

Throughout the section of numerical results, q denotes the grid level, with the sparse grid solution for a certain grid index q_k intended to approximate the full grid solution at the same index q_k, N_p denotes the number of subproblems, N_S denotes the number of spatial gridpoints used, N_τ denotes the number of timesteps taken (for the combination method, we count the gridpoints and timesteps from all subproblems), and convergence order is calculated with successive differences; that is,

$$\text{order} = \log_2 \left((u_q - u_{q-1})/(u_{q+1} - u_q) \right) \tag{9}$$

where u_q is the solution of the q-th grid level, measured pointwise at (K,K) in the two-dimensional case, or (K,K,K) in the three-dimensional case.

5.1 American Basket Put

Our first problem is the American Basket Put. In all tables and relevant discussion, "Sparse Grid method" refers to the combination method. The Full Grid method is not as sensitive to the unsmooth $\mathscr{C}^0$ payoff, and even with unsmoothed ICs, exhibits second-order convergence. The Sparse Grid method on the original coordinates with 2D-averaging gives compatible (if not slightly better) results. The Sparse Grid method on the transformed coordinates as given in [9] is also comparable. We point out that, since the coordinate transformation aligns the discontinuity with one coordinate axis, the Fourier transform of the IC reduces to the one-dimensional case, which is studied extensively in [2]. When using the coordinate transformation, we ensure that our one-dimensional grid in the direction orthogonal to the plane of nonsmoothness has a gridpoint (1D-alignment with gridpoint) in the location of the nonsmoothness. This allows the combination method to exhibit a stable order of convergence. In the terminology of [2], the quantization error is always present and is second order, with a constant coefficient.

5.2 Digital Basket Call

We consider the Digital Basket Call option to demonstrate our numerical method on $\mathscr{C}^{-1}$ ICs, show that the Greeks have stable convergence also, and to further show that the coordinate transformation in [9] allows us to analyze basket options using the one-dimensional case. Although the coordinate transformation method allows us, with a simple 1D-alignment with a gridpoint, to avoid smoothing for $\mathscr{C}^0$ ICs, as we have seen in the previous example, this is a feature of the nonsmooth but continuous ICs and not entirely of the transformation itself. When we have discontinuous ICs, such as this problem, we cannot avoid smoothing *even on full grids*. This is apparent from the first sub-table of Table 2, where the Full Grid method is first-order. In the second sub-table, the Full Grid method on transformed coordinates and nei-

q	N_p	N_τ	N_S	iters	value	difference	order	time (s)
Full Grid method, no smoothing								
1	1	34	1089	63	0.874869332	—	—	2.79e-01
2	1	66	4225	129	0.877385013	2.52e-03	—	2.48e+00
3	1	130	16641	258	0.878047559	6.63e-04	1.92	2.36e+01
4	1	258	66049	515	0.878225544	1.78e-04	1.90	2.58e+02
5	1	514	263169	1027	0.878265165	3.96e-05	2.17	2.46e+03
Sparse Grid method, original coordinates, 2D-averaging								
2	3	198	5379	365	0.877620427	—	—	9.30e-01
3	5	650	17029	1203	0.878118233	4.98e-04	—	2.97e+00
4	7	1806	46471	3373	0.878241795	1.24e-04	2.01	1.28e+01
5	9	4626	117641	8687	0.878276333	3.45e-05	1.85	5.18e+01
6	11	11286	284555	21153	0.878284761	8.43e-06	2.03	3.11e+02
7	13	26650	667533	50235	0.878287086	2.33e-06	1.86	1.44e+03
Sparse Grid method, transformed coordinates, 1D-alignment with gridpoint								
2	3	198	5379	382	0.876834538	—	—	2.57e+00
3	5	650	17029	1231	0.877947454	1.11e-03	—	4.73e+00
4	7	1806	46471	3429	0.878204302	2.57e-04	2.12	1.26e+01
5	9	4626	117641	8797	0.878267177	6.29e-05	2.03	5.72e+01
6	11	11286	284555	21583	0.878282833	1.57e-05	2.01	3.03e+02
7	13	26650	667533	50819	0.878286697	3.86e-06	2.02	1.51e+03

Table 1 Two-dimensional American Basket Put Option, with parameters $\sigma_i = 0.4$, $\rho = 0.2$, $r = 0.1$, $T = 1$, and $K = 10$, solved with penalty method, and quadratic transformation of time domain. In the Sparse Grid method, all subproblems are solved in parallel.

ther smoothing, nor proper 1D-alignment with a midpoint is also first-order; see also first-order results in Table 8 of [9]. But, in the third sub-table, 2D-averaging (second sub-table) restores second-order for the Full Grid method. We note that, in theory, 2D-averaging (approximation to 2D-Φ_1) may not be enough for stable second-order, but the Full Grid method is not as sensitive to nonsmoothness as the Sparse Grid method. Also, in the fourth sub-table, transformed coordinates and 1D-alignment with a midpoint give second-order errors.

Due to lack of space, we present Sparse Grid method results exclusively with smoothing, since smoothing is necessary even for Full Grid. To get second-order with the Sparse Grid method, we use transformed coordinates, *and* 1D-alignment with a midpoint in all grid refinements (fifth sub-table), which is easy to apply in one dimension, namely the dimension orthogonal to the plane of discontinuity. The second-order error can be explained mathematically, by applying the multidimensional Fourier transform to the IC. Since the IC in the new coordinate system depends on only one variable, the Fourier transform is the same as the one-dimensional transform in that variable. Therefore, the theory in [2] applies. In the original coordinate system, we cannot apply the theory of grid alignment, since the anisotropic grids from the sparse grid combination method cause gridpoints to have varying alignments with the plane of discontinuity. If we wish to keep the original coordinates with the Sparse Grid method, we need to apply 2D-Φ_2 smoothing (sixth sub-table).

The last sub-table has a different structure, because we also want to show convergence of the Greeks; we indicate the Gamma (V_{xx}). Note that N_p, N_τ and N_S are

the same as the previous sub-table. The number of iterations would be very similar, hence is is omitted. Convergence is stable and monotonic.

q	N_p	N_τ	N_S	iters	value	difference	order	time (s)
Full Grid method, original coordinates, no smoothing								
1	1	34	1089	157	0.5335624882	—	—	3.58e-02
2	1	66	4225	401	0.5413892716	7.83e-03	—	1.76e-01
3	1	130	16641	981	0.5454967719	4.11e-03	0.93	1.49e+00
4	1	258	66049	2630	0.5475973997	2.10e-03	0.97	1.70e+01
5	1	514	263169	7058	0.5486592436	1.06e-03	0.98	1.76e+02
Full Grid method, transformed coordinates, no smoothing, 1D-alignment with gridpoint								
1	1	34	1089	136	0.5316505597	—	—	3.80e-02
2	1	66	4225	301	0.5402955722	8.65e-03	—	1.53e-01
3	1	130	16641	657	0.5449069062	4.61e-03	0.91	1.09e+00
4	1	258	66049	1555	0.5472907262	2.38e-03	0.95	1.11e+01
5	1	514	263169	3623	0.5485028562	1.21e-03	0.98	9.54e+01
Full Grid method, original coordinates, 2D-averaging								
1	1	34	1089	170	0.5506961685	—	—	5.36e-02
2	1	66	4225	398	0.5499706420	-7.26e-04	—	1.95e-01
3	1	130	16641	1038	0.5497891960	-1.81e-04	2.00	1.73e+00
4	1	258	66049	2632	0.5497438271	-4.54e-05	2.00	1.74e+01
5	1	514	263169	7065	0.5497323344	-1.15e-05	1.98	1.77e+02
Full Grid method, transformed coordinates, 1D-alignment with midpoint								
1	1	34	1089	138	0.5520491862	—	—	1.12e-01
2	1	66	4225	322	0.5502341477	-1.82e-03	—	1.70e-01
3	1	130	16641	658	0.5498469489	-3.87e-04	2.23	1.08e+00
4	1	258	66049	1555	0.5497573230	-8.96e-05	2.11	1.05e+01
5	1	514	263169	3624	0.5497357514	-2.16e-05	2.05	9.32e+01
Sparse Grid method, transformed coordinates, 1D-alignment with midpoint								
2	3	198	5379	809	0.5502339155	—	—	8.77e-01
3	5	650	17029	2565	0.5498468301	-3.87e-04	—	1.14e+00
4	7	1806	46471	6796	0.5497572770	-8.96e-05	2.11	1.76e+00
5	9	4626	117641	17033	0.5497357335	-2.15e-05	2.06	5.06e+00
6	11	11286	284555	41471	0.5497304484	-5.29e-06	2.03	1.95e+01
7	13	26650	667533	92999	0.5497291349	-1.31e-06	2.01	7.35e+01
8	15	61470	1531791	206345	0.5497287984	-3.37e-07	1.96	3.33e+02
9	17	139298	3456913	450930	0.5497286947	-1.04e-07	1.70	1.59e+03
Sparse Grid method, original coordinates, 2D-Φ_2 smoothing								
	Gamma (V_{xx})				value (V)			time (s)
2	1.729238e-04	—	—		0.5501696945	—	—	2.06e-01
3	1.726895e-04	-2.34e-07	—		0.5498427171	-3.27e-04	—	3.82e-01
4	1.726071e-04	-8.24e-08	1.51		0.5497595027	-8.32e-05	1.97	1.18e+00
5	1.725810e-04	-2.61e-08	1.66		0.5497369071	-2.26e-05	1.88	3.88e+00
6	1.725732e-04	-7.83e-09	1.74		0.5497308590	-6.05e-06	1.90	1.48e+01
7	1.725709e-04	-2.31e-09	1.76		0.5497292769	-1.58e-06	1.93	6.26e+01
8	1.725701e-04	-7.79e-10	1.57		0.5497288599	-4.17e-07	1.92	2.76e+02
9	1.725699e-04	-2.09e-10	1.90		0.5497287516	-1.08e-07	1.94	1.28e+03

Table 2 Two-dimensional Digital Basket Call Option, with parameters $\sigma_x = 0.4$, $\sigma_y = 0.2$, $\rho = 0.2$, $r = 0.1$, $T = 1$, and $K = 10$. In the Sparse Grid method, all subproblems are solved in parallel.

5.3 Three-Dimensional Min-Put

Unlike basket options, the Min-Put, a put option written on the minimum of several assets, has ICs with multiple regions of discontinuity, that make alignment with a coordinate axis difficult, if not impossible. Hence, these experiments demonstrate the more general nature of the averaging technique, compared to applying transformations. In Table 3, we present three sub-tables, one showing second-order convergence of the Full Grid method, and two showing steady and erratic convergence of the Sparse Grid method with and without smoothing, respectively. In the last sub-table, we include results with minimum number of subintervals in all grids and subproblems $\min_N = 12$, and $\min_N = 24$. A larger $\min_N$ improves the convergence rate, at the cost of increased runtime [10]. For $\min_N = 12$, N_τ and N_S are the same as no smoothing, with similar runtime. For $\min_N = 24$, N_τ is doubled and N_S octupled. The efficiency of the Sparse Grid method is emphasized in the 3D problem, with the highest grid index ($q = 5$) of the Full Grid method using similar time as the highest grid index ($q = 8$) of the Sparse Grid method for 100 times the error. For the Full Grid method, we estimate that it would take about three months for $q = 8$.

q	N_p	N_τ	N_S	iters	value	difference	order	time (s)
Full Grid method, no smoothing								
1	1	14	2197	42	1.631619998	—	—	2.66e-01
2	1	26	15625	78	2.082438148	4.51e-01	—	2.25e-01
3	1	50	117649	201	2.177623887	9.52e-02	2.24	3.78e+00
4	1	98	912673	588	2.200021333	2.24e-02	2.09	9.13e+01
5	1	194	7189057	1769	2.205548944	5.53e-03	2.02	1.60e+03
Sparse Grid method, no smoothing ($\min_N = 12$)								
3	10	500	64090	1082	2.210251041	—	—	1.94e+00
4	19	1862	222247	3907	2.245322399	3.51e-02	—	1.42e+00
5	31	6014	682087	12028	2.212317996	-3.30e-02	0.09	2.74e+00
6	46	17756	1937098	34874	2.218271375	5.95e-03	2.47	1.10e+01
7	64	49280	5214448	89798	2.208918361	-9.35e-03	-0.65	6.84e+01
8	85	130730	13497241	219145	2.210013624	1.10e-03	3.09	3.57e+02
Sparse Grid method, 3D-aver., $\min_N = 12$					Sparse Grid method, 3D-aver., $\min_N = 24$			
3	2.226693054	—	—		—	—	—	—
4	2.215203163	-1.15e-02	—		2.210620071	—	—	1.37e+01
5	2.209757568	-5.45e-03	1.08		2.208408052	-2.21e-03	—	2.38e+01
6	2.208213011	-1.54e-03	1.82		2.207671564	-7.36e-04	1.59	7.16e+01
7	2.207626025	-5.87e-04	1.40		2.207493028	-1.79e-04	2.04	3.11e+02
8	2.207437289	-1.89e-04	1.64		2.207433828	-5.92e-05	1.59	1.46e+03

Table 3 Three-dimensional European Min-Put Option, with parameters $\sigma_i = 0.4$, $\rho_{i,j} = 0.2$ for $i \neq j$, $r = 0.1$, $T = 1$, and $K = 10$. In the Sparse Grid method, all subproblems are solved in parallel.

6 Conclusions

The sparse grid combination method allows numerical solutions of multidimensional PDEs to be computed on grids that have finer resolution than is feasible for

standard full grid methods. We focused on problems from computational finance, which are characterized by nonsmooth ICs, that are shown to cause trouble if the sparse grid combination method is blithely applied. Like other researchers [9, 6, 7], we turned to smoothing techniques to remedy this. We highlighted particular features. The coordinate transformation [9] eliminates discontinuities in all but one dimension, and allows the application of grid-alignment theory from [2]. Averaging [11] allows the treatment of more general ICs with discontinuities that cannot be aligned to a coordinate axis, and gives rise to errors equivalent to coordinate transformation, for $\mathscr{C}^0$ ICs. We obtained second-order accurate solutions and Greeks (without visible deterioration) for 2D problems, including American options. For 3D problems and complicated payoffs, we obtained the expected (slightly reduced) order of convergence.

Subsequent work involves the application of sparse grids to other multidimensional nonlinear PDEs, their smoothing requirements, the study of the quantization error in multiple dimensions, dimension-adaptive techniques for transformed variables, and closed-form formulae for the smoothing of common payoffs.

References

1. Black, F., Scholes, M.: The pricing of options and corporate liabilities. Journal of political economy **81**(3), 637–654 (1973)
2. Christara, C., Leung, N.C.H.: Analysis of quantization error in financial pricing via finite difference methods. SIAM J. Numer. Anal. **56**(3), 1731–1757 (2018)
3. Clarke, N., Parrott, K.: Multigrid for American option pricing with stochastic volatility. Appl. Math. Finance **6**(3), 177–195 (1999)
4. Forsyth, P., Vetzal, K.: Quadratic convergence for valuing American options using a penalty method. SIAM J. Sci. Comput. **23**(6), 2095–2122 (2002)
5. Griebel, M., Schneider, M., Zenger, C.: A combination technique for the solution of sparse grid problems. Proceedings of the IMACS International Symposium on Iterative Methods in Linear Algebra pp. 263–292 (1992)
6. Hendricks, C.: High-order methods for parabolic equations in multiple space dimensions for option pricing problems. Ph.D. thesis, Bergische Universität Wuppertal (2016)
7. Hendricks, C., Heuer, C., Ehrhardt, M., Günther, M.: High-order ADI finite difference schemes for parabolic equations in the combination technique with application in finance. J. Comput. Appl. Math. **316**, 175–194 (2017)
8. Kreiss, H.O., Thomée, V., Widlund, O.: Smoothing of initial data and rates of convergence for parabolic difference equations. Comm. Pure Appl. Math. **23**(2), 241–259 (1970)
9. Leentvaar, C., Oosterlee, C.W.: On coordinate transformation and grid stretching for sparse grid pricing of basket options. J. of Comput. Appl. Math. **222**(1), 193–209 (2008)
10. López-Salas, J.G., Pérez-Rodríguez, S., Vazquez, C.: AMFR-W numerical methods for solving high-dimensional SABR/LIBOR PDE models. SIAM J. Sci. Comput. **43**(1), B30–B54 (2021)
11. Pooley, D., Vetzal, K., Forsyth, P.: Convergence remedies for non-smooth payoffs in option pricing. J. Comput. Finance **6**(4), 25–40 (2003)
12. Rannacher, R.: Finite element solution of diffusion problems with irregular data. Numerische Mathematik **43**(2), 309–327 (1984)
13. Reisinger, C., Whitley, A.: The impact of a natural time change on the convergence of the Crank–Nicolson scheme. IMA J. Numer. Anal. **34**(3), 1156–1192 (2014)
14. Smolyak, S.A.: Quadrature and interpolation formulas for tensor products of certain classes of functions. Doklady Akademii Nauk, Rus. Acad. Sci. **148**(5), 1042–1045 (1963)