Efficient Numerical Methods for Stochastic Differential Equations in Computational Finance

Dissertation by
Juho Häppölä

In Partial Fulfillment of the Requirements

For the Degree of

Doctor of Philosophy

King Abdullah University of Science and Technology
Thuwal, Kingdom of Saudi Arabia

August 2017
The dissertation of Juho Häppölä is approved by the examination committee.

Committee Chairperson: Professor Raúl Tempone
External examiner: Professor Boualem Djehiche
Committee Members: Professors Mohamed Slim-Alouini, Diogo Gomes, Ernesto Mordecki, and Jorge Zubelli.
Efficient Numerical Methods for Stochastic Differential Equations in Computational Finance
Juho Häppölä

Stochastic Differential Equations (SDE) offer a rich framework to model the probabilistic evolution of the state of a system. Numerical approximation methods are typically needed in evaluating relevant Quantities of Interest arising from such models. In this dissertation, we present novel effective methods for evaluating Quantities of Interest relevant to computational finance when the state of the system is described by an SDE.
ACKNOWLEDGEMENTS

Great thanks to all the academics who have helped me in the preparation in this work. Co-authors Fabian Crocce, Jonas Kiessling, Christian Bayer, Håkon Hoel, and, of course, the person who has had profound impact on this work, Professor Raul Tempone. In addition numerous people have read parts of the work and provided additional feedback or just otherwise provided guidance on my path: Professor Ernesto Mordecki, Professor Boualem Djehiche, Dr. Joakim Beck, Dr. Abdul-Lateef Haji-Ali, Dr. Matteo Icardi, Sören Wolfers.

In addition to those immersed in my tiny bubble, tons of folks, and one special dog, have provided me with a support network and an interface to the outer world, that I could not have lived without: Albert, Archie, Azade, Chris, Claire, Dad, Erkki, Gillis, Grace, Hanna, Heikki, Hjörtur, Jarmo, Jill, Jose, Joose, Kevin, Lauren, Mari, Nabeelah, Niko, Outi, Oscar, Ozan, Pete, Rose, Teemu, Todd, Tony, and all the others.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Examination Committee Page</td>
<td>2</td>
</tr>
<tr>
<td>Copyright</td>
<td>3</td>
</tr>
<tr>
<td>Abstract</td>
<td>4</td>
</tr>
<tr>
<td>Acknowledgements</td>
<td>5</td>
</tr>
<tr>
<td>List of Figures</td>
<td>11</td>
</tr>
<tr>
<td>List of Tables</td>
<td>15</td>
</tr>
<tr>
<td><strong>I Introductory chapters</strong></td>
<td>16</td>
</tr>
<tr>
<td>1 Itô SDEs and differential calculus</td>
<td>17</td>
</tr>
<tr>
<td>1.1 Itô SDE</td>
<td>17</td>
</tr>
<tr>
<td>1.1.1 Stochastic process, Wiener process</td>
<td>18</td>
</tr>
<tr>
<td>1.1.2 Quantity of Interest, weak solutions</td>
<td>22</td>
</tr>
<tr>
<td>1.1.3 Partial differential equations</td>
<td>23</td>
</tr>
<tr>
<td>2 Financial applications</td>
<td>26</td>
</tr>
<tr>
<td>2.1 Models</td>
<td>28</td>
</tr>
<tr>
<td>2.1.1 Local volatility models</td>
<td>29</td>
</tr>
<tr>
<td>2.1.2 Stochastic volatility</td>
<td>32</td>
</tr>
<tr>
<td>2.1.3 Pure jump and jump diffusion models</td>
<td>33</td>
</tr>
<tr>
<td>2.1.4 Lévy models</td>
<td>36</td>
</tr>
<tr>
<td>2.1.5 Yield curve models</td>
<td>37</td>
</tr>
<tr>
<td>2.2 Risk-neutral measure</td>
<td>38</td>
</tr>
<tr>
<td>2.3 Payoff</td>
<td>41</td>
</tr>
<tr>
<td>2.3.1 Plain vanilla</td>
<td>42</td>
</tr>
<tr>
<td>2.3.2 Binary</td>
<td>44</td>
</tr>
<tr>
<td>2.3.3 Averaging payoffs</td>
<td>45</td>
</tr>
</tbody>
</table>
2.3.4 Stopping and hitting times ........................................ 45

3 Numerical methods ..................................................... 49
  3.1 Exact methods ..................................................... 49
  3.2 Forward simulation, Monte Carlo methods ...................... 50
    3.2.1 Variance reduction schemes .................................. 51
      3.2.1.1 Antithetic variables .................................. 51
      3.2.1.2 Control variates .................................... 52
      3.2.1.3 Multilevel Monte Carlo ............................ 54
    3.3 PDE methods ............................................... 54

4 Contributions of papers .............................................. 57
  4.1 Article I ....................................................... 57
  4.2 Article II ..................................................... 58
  4.3 Article III .................................................... 59

5 Concluding remarks ................................................... 60

References ............................................................... 62

II Included Articles .................................................... 67

6 Article I .............................................................. 68
  6.1 Introduction .................................................... 68
    6.1.1 Monte Carlo Methods and the Euler–Maruyama Scheme .... 70
    6.1.2 Uniform and Adaptive Time-Stepping MLMC ............... 72
      6.1.2.1 MLMC Notation .................................... 72
      6.1.2.2 Uniform Time-Stepping MLMC .................. 73
      6.1.2.3 Uniform Time-Stepping MLMC Error and Computational Complexity ........................................ 73
      6.1.2.4 MSE A Posteriori Adaptive Time-Stepping .......... 75
      6.1.2.5 An MSE Adaptive MLMC Algorithm ................ 76
      6.1.2.6 Earlier Works on Adaptivity for SDE ............. 77
    6.2 Derivation of the MSE A Posteriori Adaptive Algorithm .... 78
      6.2.1 The Error Expansion .................................. 79
        6.2.1.1 Numerical Approximation of the First Variation .. 81
      6.2.2 The Adaptive Algorithm ................................ 86
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.2.2.1 Mesh Refinement Strategy</td>
<td>88</td>
</tr>
<tr>
<td>6.2.2.2 Wiener Path Refinements</td>
<td>89</td>
</tr>
<tr>
<td>6.2.2.3 Updating the Error Indicators</td>
<td>89</td>
</tr>
<tr>
<td>6.2.2.4 Pseudocode</td>
<td>90</td>
</tr>
<tr>
<td>6.2.3 Numerical Examples</td>
<td>91</td>
</tr>
<tr>
<td>6.3 Extension of the Adaptive Algorithm to the Multilevel Setting</td>
<td>96</td>
</tr>
<tr>
<td>6.3.1 Notation and Objective</td>
<td>96</td>
</tr>
<tr>
<td>6.3.1.1 The Mesh Hierarchy</td>
<td>97</td>
</tr>
<tr>
<td>6.3.2 Error Control</td>
<td>98</td>
</tr>
<tr>
<td>6.3.2.1 The Statistical Error</td>
<td>99</td>
</tr>
<tr>
<td>6.3.2.2 The Time Discretization Error</td>
<td>101</td>
</tr>
<tr>
<td>6.3.2.3 Computational Cost</td>
<td>102</td>
</tr>
<tr>
<td>6.3.3 MLMC Pseudocode</td>
<td>102</td>
</tr>
<tr>
<td>6.4 Numerical Examples for the MLMC Algorithms</td>
<td>104</td>
</tr>
<tr>
<td>6.4.0.1 The MSE Expansion for the Adaptive Algorithm</td>
<td>110</td>
</tr>
<tr>
<td>6.4.0.2 Implementation Details and Observations</td>
<td>110</td>
</tr>
<tr>
<td>6.4.1 Computer Implementation</td>
<td>113</td>
</tr>
<tr>
<td>6.5 Conclusion</td>
<td>114</td>
</tr>
<tr>
<td>References</td>
<td>115</td>
</tr>
<tr>
<td>6.A Theoretical Results</td>
<td>121</td>
</tr>
<tr>
<td>6.A.1 Error Expansion for the MSE in 1D</td>
<td>121</td>
</tr>
<tr>
<td>6.A.2 Second, Third and Fourth Variations</td>
<td>134</td>
</tr>
<tr>
<td>6.A.3 Error Expansion for the MSE in Multiple Dimensions</td>
<td>141</td>
</tr>
<tr>
<td>6.B A Uniform Time Step MLMC Algorithm</td>
<td>144</td>
</tr>
<tr>
<td>7 Article II</td>
<td>146</td>
</tr>
<tr>
<td>7.1 Introduction</td>
<td>146</td>
</tr>
<tr>
<td>7.2 Fourier method for option pricing</td>
<td>149</td>
</tr>
<tr>
<td>7.2.1 Evaluation of the method for multiple values of ( x ) simultaneously</td>
<td>153</td>
</tr>
<tr>
<td>7.3 Error bound</td>
<td>154</td>
</tr>
<tr>
<td>7.3.1 Quadrature error</td>
<td>155</td>
</tr>
<tr>
<td>7.3.2 Frequency truncation error</td>
<td>163</td>
</tr>
<tr>
<td>7.3.3 Bound for the full error</td>
<td>164</td>
</tr>
<tr>
<td>7.3.4 Explicit error bounds</td>
<td>166</td>
</tr>
<tr>
<td>7.4 Computation and minimization of the bound</td>
<td>169</td>
</tr>
<tr>
<td>7.4.1 Call option in variance gamma model</td>
<td>169</td>
</tr>
</tbody>
</table>
7.4.2 Call options under Kou dynamics ........................................... 174
7.4.3 Binary option in the Merton model ....................................... 176
7.4.4 Call options ........................................................................ 176

7.5 Conclusion ............................................................................. 179
7.6 Acknowledgements .............................................................. 181
7.7 Declarations of Interest .......................................................... 181
References .................................................................................. 181

7.A Truncation bound in Lee’s scheme .......................................... 186

8 Article III .................................................................................. 188

8.1 Introduction ........................................................................... 189
  8.1.0.0.1 Outline ....................................................................... 192

8.2 Markovian projections and implied stopping times .................. 193
  8.2.1 Markovian projections and approximate stopping times ...... 194
  8.2.2 Implied stopping time and price bounds ............................. 201
    8.2.2.1 Lower bound ................................................................ 201
    8.2.2.2 Upper bound ................................................................ 202

8.2.3 Dimension reduction for models relevant to quantitative finance 204
  8.2.3.1 Definitions ..................................................................... 205
  8.2.3.2 Bachelier model ............................................................ 206
    8.2.3.2.1 Step 1 ...................................................................... 207
    8.2.3.2.2 Step 2 ...................................................................... 207
  8.2.3.3 Other models in reduced dimension .................................. 209
  8.2.3.4 Black-Scholes model ..................................................... 212

8.3 Numerical implementation .................................................... 215
  8.3.1 Evaluation of local volatility ................................................. 215
    8.3.1.1 Laplace approximation ............................................... 215
    8.3.1.2 Extrapolation-interpolation to projected domain .......... 219
  8.3.2 Numerical value function ................................................... 221
  8.3.3 Forward-Euler approximation ............................................. 222
  8.3.4 Error decomposition .......................................................... 224
  8.3.5 Examples .......................................................................... 227
    8.3.5.1 American put on a basket in the Bachelier model ......... 228
    8.3.5.2 3-to-1 dimensional Black-Scholes model ..................... 230
    8.3.5.3 10-to-1 dimensional Black-Scholes model ................... 233
    8.3.5.4 25-to-1 dimensional Black-Scholes model ..................... 235
LIST OF FIGURES

1.1 (1.1(a)) A single realization of a Wiener process $W(t)$ (1.1(a)) on a selection of nested temporal meshes. (1.1(b)) The corresponding realizations of a particular SDE. Note that whereas the nested approximations of $W$ coincide for different levels of approximation, the same is not the case for the trajectory of the SDE due to non-constant drift and diffusion. .................................................. 21

6.1 (Left) A sample Wiener path, $W$, generated on the coarse mesh, $\Delta t^{(0)}$, with uniform step size 1/10 (blue line). The path is thereafter Brownian bridge interpolated onto a finer mesh, $\Delta t^{(1)}$, which has uniform step size of 1/20 (green line). (Right) Euler–Maruyama numerical solutions of the Ornstein-Uhlenbeck SDE problem, $dX_t = 2(1 - X_t)dt + 0.2dW_t$, with initial condition $X_0 = 3/2$, are computed on the meshes $\Delta t^{(0)}$ (blue line) and $\Delta t^{(1)}$ (green line) using Wiener increments from the respective path resolutions. .................................................. 73

6.2 Comparison of the performance of uniform and adaptive time-stepping Euler–Maruyama numerical integration for Example 6.2.9 in terms of MSE vs. number of time steps. .................................................. 95

6.3 The ratio of the level-wise mean number of time steps $E[|\Delta t^{(\ell)}|]/N_\ell$, of MSE adaptive mesh realizations to uniform mesh realizations for Example 6.4.3 .................................................. 106

6.4 For a set of $\epsilon_T$ values, 100 realizations of the MSE adaptive multilevel estimator are computed using both MLMC algorithms for Example 6.4.1. The errors $|A_{M\ell}(\omega_i; \epsilon_T, \delta) - E[g(X_T)]|$ are respectively plotted as circles (adaptive MLMC) and triangles (uniform MLMC), and the number of multilevel estimator realizations failing the constraint $|A_{M\ell}(\omega_i; \epsilon_T, \delta) - E[g(X_T)]| < \epsilon_T$ is written above the $(\epsilon_T^{-1}, \epsilon_T)$ line. Since the confidence parameter is set to $\delta = 0.1$ and less than 10 realizations fail for any of the tested $\epsilon_T$ values, both algorithms meet the approximation goal (6.39). .................................................. 107
6.5 Average runtime vs. $\epsilon_T^{-1}$ for the two MLMC algorithms solving Example 6.4.1 .......................................................... 107

6.6 Output for Example 6.4.1 solved with the MSE adaptive and uniform time-stepping MLMC algorithms. (Top) Weak error $|E[\Delta_t g]|$ for solutions at $\epsilon_T = 10^{-3}$. (Middle) Variance $\text{Var}(\Delta_t g)$ for solutions at $\epsilon_T = 10^{-3}$. (Bottom) Average number of samples $E[M_\ell]$ .......................................................... 108

6.7 (Top) One MSE adaptive numerical realization of the SDE problem (6.48) at different mesh hierarchy levels. The blow-up singularity point is located at $\xi \approx 0.288473$ and the realizations are computed for three singularity exponent values. We observe that as the exponent, $p$, increases, the more jump at $t = \xi$ becomes more pronounced. (Bottom) Corresponding MSE adaptive mesh realizations for the different test cases .......................................................... 111

6.8 (Top) Average errors $|E[\Delta_t g]|$ for Example 6.4.3 solved with the MSE adaptive MLMC algorithm for three singularity exponent values. (Bottom) Corresponding average errors for the uniform MLMC algorithm. ........................................................................................................ 112

6.9 (Top) Variances $\text{Var}(\Delta_t g)$ for for Example 6.4.3 solved with the MSE adaptive MLMC algorithm for three singularity exponent values. (Bottom) Corresponding variances for the uniform MLMC algorithm. The more noisy data on the highest levels is due to the low number used for the initial samples, $M = 20$, and only a subset of the generated 100 multilevel estimator realizations reached the last levels. .......................................................... 113

6.10 (Top) Average number of samples $M_\ell$ for for Example 6.4.3 solved with the MSE adaptive MLMC algorithm for three singularity exponent values. (Bottom) Corresponding average number of samples for the uniform MLMC algorithm. The plotted decay rate reference lines, $c2^{-(\beta(p)+1)/\ell}$, for $M_\ell$ follow implicitly from equation (6.45) (assuming that $\beta(p) = 2(1 - p)$ is the correct variance decay rate). ........................................................................................................ 114

6.11 Average runtime vs. $\epsilon_T^{-1}$ for the two MLMC algorithms for three singularity exponent values in Example 6.4.3 .......................................................... 115

6.12 Approximation errors for both of the MLMC algorithms solving Example 6.4.3 At every $\epsilon_T$ value, circles and triangles represent the errors from 100 independent multilevel estimator realizations of the respective algorithms. ........................................................................................................ 116
<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.1</td>
<td>The true error and the error bound for evaluating at the money options for the VG model test case.</td>
</tr>
<tr>
<td>7.2</td>
<td>The true error $\mathcal{E}$ for the two VG test cases presented in Table 7.1 and the bound-minimising configurations (white circle) $(\alpha_2, \Delta \omega_2)$ for the examples.</td>
</tr>
<tr>
<td>7.3</td>
<td>The true error $E_1$ and the bound $\varepsilon$ for the dissipative Merton model, for a range of quadrature points $n$, along with the bound-minimising configurations contrasted to the true error.</td>
</tr>
<tr>
<td>8.1</td>
<td>Projected volatility $\tilde{b}_1(t,s)$ defined in (8.41) and its interpolation in space and time for the 3-to-1 dimensional Black-Scholes model (8.56). In both the figures, the plots are done for the range of essential support of the density, which expands as $t$ increases.</td>
</tr>
<tr>
<td>8.2</td>
<td>Convergence of the upper and lower bounds for the Bachelier model described in Section 8.3.5.1 and the resulting relative errors for the American and at-the-money put options.</td>
</tr>
<tr>
<td>8.3</td>
<td>Convergence of the upper and lower bounds for the dissipative Merton model described in Section 8.3.5.1 and the resulting relative errors for the American options.</td>
</tr>
<tr>
<td>8.4</td>
<td>Both European and American put option prices for the test case (8.56) and the corresponding relative errors. For comparison of the solvers, identical spatial and temporal meshes, sample sizes and number of Monte Carlo realizations are used for solving both the European and the American options.</td>
</tr>
<tr>
<td>8.5</td>
<td>The value function of the 3-to-1-dimensional Black-Scholes example (8.56) and the corresponding early exercise boundary.</td>
</tr>
<tr>
<td>8.6</td>
<td>Convergence of the upper and lower bounds for the 10-to-1 dimensional Black-Scholes model (8.58) and the resulting relative errors. As in Figure 8.4, same numerical parameters have been used for solving both the European and the American prices for ease of comparison.</td>
</tr>
<tr>
<td>8.7</td>
<td>Price uncertainty in the 10-dimensional Black-Scholes model (8.58) with varying numbers of time steps $N_t$ in the forward-Euler monte Carlo.</td>
</tr>
<tr>
<td>8.8</td>
<td>Convergence of the upper and lower bounds for the 25-to-1-dimensional Black-Scholes model and the resulting relative errors.</td>
</tr>
</tbody>
</table>
8.9 American put prices (8.9(a)) and corresponding relative errors (8.9(b))
for 43 independent randomized repetitions on evaluating the American
put on the 25-to-1-dimensional Black-Scholes model with 258 individ-
ual option price valuations for varying strike, K. In addition to the
random structure of the test problem by [6] and the parameters (8.59)
and $T = \frac{1}{4}$, we also randomize the portfolio weights. For each of the
runs, we choose $P_{1i}$ independently from an uniform distribution
$U[\frac{1}{2}, \frac{3}{2}]$
and finally rescale the weights so that $\sum_{i=1}^{25} P_{1i} = 25$. . . . . . . . . . 236

8.10 Functions $f_1$ of (8.61) (8.10(a)) and $\tilde{f}_1$ of (8.60) (8.10(b)) in blue and
their respective approximations based on the second-order Taylor ex-
pansions of their logarithms in dashed red. . . . . . . . . . . . . . . 246

8.11 Functions $f_2$ of (8.63) (8.11(a)) and $f_2$ of (8.62) (8.11(b)) in blue and
and their respective approximations based on the second-order Taylor
expansions of their logarithms in dashed red. . . . . . . . . . . . . . . 247
## LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.1</td>
<td>Observed computational cost – disregarding log((\varepsilon_T)) multiplicative factors of finite order – for the drift blow-up study in Example 6.4.3.</td>
<td>77</td>
</tr>
<tr>
<td>6.2</td>
<td>List of parameter values used by the MSE adaptive MLMC algorithm and (when required) the uniform MLMC algorithm for the numerical examples in Section 6.4.</td>
<td>105</td>
</tr>
<tr>
<td>7.1</td>
<td>The error bound for European call/put options in the VG model for select examples. Reference result (\mathcal{E}^*) from [24]</td>
<td>173</td>
</tr>
<tr>
<td>7.2</td>
<td>Numerical performance of the bound for the Kou model, with the test case in [38] (see also [13]) with the number of quadrature points set to (n = 32). The point of comparison (\mathcal{E}^*) refers to the corresponding bound computed with the method described in Chapters 6.1 to 6.4 of [24]. In the (\mathcal{E}^\dagger), the cutoff error has been evaluated using a computationally more intensive Clenshaw-Curtis quadrature instead of an asymptotic argument with an exponentially decaying upper bound for the option price.</td>
<td>174</td>
</tr>
</tbody>
</table>
Part I

Introductory chapters
Chapter 1

Itô SDEs and differential calculus

In this Section, we present some of the mathematical concepts that underlie the articles included in this thesis. The discussion contains only the core of the concepts needed to discuss the contributions of the articles. We refer the interested reader to the many works written on SDEs and their applications in finance, and their solutions [6, 37, 38, 50, 52, 57].

1.1 Itô SDE

At the heart of this thesis, in all the particular contributions we assume a forward model, or simply a model. The models of interest for this dissertation, with the slight exception of Article II of Section 7, are Stochastic Differential Equations of Itô type. Even in the case of the Article II, Itô SDEs lay the theoretical foundation of our work, and only some additional terms are included to account for a wider range of phenomena. Itô stochastic differential equations are mathematical objects that describe the random evolution of one or more, possibly infinitely many state variables as an external parameter, typically understood to be time, is evolving.

We discuss the properties of the Wiener process and how it gives rise to Itô SDEs. We also later discuss generalizations of the Itô SDEs such as jump-diffusions, to the extent they are relevant for the work at hand. We also include a brief discussion on the particular quantities one is typically interested in evaluating when using Itô SDEs to model stochastic processes.
For the bulk of this work, refrain from addressing the question of which of the infinitely many Itô SDEs is best suited for modeling a particular state variable. Instead, we assume the model given and focus on the efficient computation of relevant quantities of interest. We do note, however, that our work carries certain indirect relevance to selecting and calibrating an appropriate model in the sense that often model selection involves evaluating the same quantity of interest for a range of models and then determining which of those models best correspond to the observed reality, often through some optimization procedure.

1.1.1 Stochastic process, Wiener process

Stochastic differential equations are essentially equations of motion for dynamic systems that evolve subject to probabilistic description. Given a triplet $(\Omega, \mathcal{F}, \mathbb{P})$, with $\Omega$ a space and $\mathcal{F}$ a sigma-algebra of sets measurable with a probability measure $\mathbb{P}$, a mapping $X(t) : \Omega \rightarrow \mathbb{R}^d$, $t > 0$ is called a stochastic process in continuous time. In principle, $t > 0$ can be unbounded, but in the work at hand, we consider dynamics for $t \in [0, T]$ unless otherwise noted. Likewise, the range can be generalised from the $\mathbb{R}^d$ into a more general one, possibly in infinite dimension.

The canonical case of a continuous-time stochastic process in $\mathbb{R}^d$ is that of the Wiener process, which plays a key role in constructing many of the stochastic models used in this dissertation. The Wiener process $W(t)$ is a stochastic process with the following properties:

1. $W(t)$ is almost surely continuous and $W(0) = 0$.

2. For any $0 < t_1 < t_2 < t_3 < t_4$, the increments $W(t_2) - W(t_1)$ and $W(t_4) - W(t_3)$ are statistically independent and

3. for any $0 < t_1 < t_2$ components of $W(t_2) - W(t_1)$ are normally distributed with mean zero and variance $t_2 - t_1$. 
In many sources, the Wiener process is defined in $\mathbb{R}$ rather than $\mathbb{R}^d$, however, the main principle remains the same and the individual components of the $\mathbb{R}^d$-valued Wiener process are independent $\mathbb{R}$-valued Wiener-processes. In order to simplify our notation, we refrain from explicitly writing $\mathbf{W}$ or $\bar{\mathbf{W}}$ to distinguish the dimension of $W$. In typical cases, the dimension will be obvious from the context.

One way to understand the Wiener process is through its construction using discrete approximations. Given a strictly increasing mesh of times $\{t_i\}_{i=0}^{N_t}$, $t_0 = 0$, $t_{N_t} = T$, the values of $W(t_n)$ can be understood as the sums of random variables

$$W(t_n) = \sum_{j=1}^{N_t} \left( W(t_j) - W(t_{j-1}) \right),$$

and the full Wiener process for the interval $[0, T]$ is understood as the limit of $N_t$ tending to infinity, while maintaining that the maximum mesh spacing goes to zero,

$$\max_{0 \leq i < N} |t_{i+1} - t_i| \to 0$$

We define the Wiener process $W$ to be the resulting limit:

$$W(t) = \int_0^t dW(s)$$

A more general Itô Integral can be defined, with a volatility function $b : \mathbb{R}^{d+1} \to \mathbb{R}^{d \times k}$, this allows the definition of an Itô Integral $I_b(t)$ as

$$I_b(t) = \int_0^t b(s, I(s)) \, dW(s)$$
with the integral understood as the limit of discrete sums

\[ \tilde{I}_b(t_n) = \sum_{j=1}^{n} b(t_{j-1}, \tilde{T}(t_{j-1})) \frac{W(t_j) - W(t_{j-1})}{\sim N(0, t_j - t_{j-1})}, \]  

(1.1)

as \( \max_{0 \leq i < N} |t_{i+1} - t_i| \to 0 \) and \( n \to \infty \). The sum can be proven to converge for a Lipschitz continuous volatility \( b \) in \( L^2(\Omega) \). We call convergence in \( L^2 \) strong convergence. It should be noted that the choice of the discretisation point in the approximating sum above is important, as the limit depends on the discretisation. While other choices such as the Stratonovich rule exist, the above definition gives rise to the Itô integral which is of the most interest in financial applications. The reason for this is that the Itô integral is not forward-looking: the argument of the volatility function \( b \) in (1.1) is evaluated at \( t_{j-1} \), at a time that precedes the observation of the value of the random variable \( W(t_j) - W(t_{j-1}) \). The definition gives rise to an object that has mean zero for all times, a characteristic feature of all Itô Integrals. Introducing a drift term \( a: \mathbb{R}^{d+1} \to \mathbb{R}^d \), we may write the full integral

\[ I(t) = \int_0^t a(s, I(s)) \, ds + \int_0^t b(s, I(s)) \, dW(s) + I(0). \]

The integral above is often written in differential form as

\[ dI(t) = a(t, I(t)) \, dt + b(t, I(t)) \, dW(t), \quad I(0) = I_0 \]  

(1.2)

with the initial value \( I_0 \in \mathbb{R}^d \). This is the most typical form for expressing an SDE of Itô type. An illustration of the discretized Wiener process, as well as the corresponding solution of a particular SDE, is shown in Figure 1.1.

**Assumption 1.1.1** (Standard regularity assumptions on Itô SDEs). We say that an Itô SDE of form (1.2) satisfies the standard assumptions \([39\text{, pp. 128}]\) if the following hold:
Figure 1.1: 1.1(a): A single realization of a Wiener process $W(t)$ (1.1(a)) on a selection of nested temporal meshes. 1.1(b): The corresponding realizations of a particular SDE. Note that whereas the nested approximations of $W$ coincide for different levels of approximation, the same is not the case for the trajectory of the SDE due to non-constant drift and diffusion.

1. **Measurability** $a = a(t, x)$ and $b = b(t, x)$ are jointly $L^2$-measurable in $(t, x) \in [0, T] \times \mathbb{R}^d$.

2. **Lipschitz-continuity** There exists $C > 0$ such that
\[
\max \{|a(t, x) - a(t, y)|, |b(t, x) - b(t, y)|\} \leq C||y - x||
\]
for all $t \in [0, T]$, $x, y \in \mathbb{R}^d$.

3. **Linear growth bound** There exists $C > 0$ such that
\[
\max \{|a(t, x)|^2, |b(t, x)|^2\} \leq C^2(1 + ||x||^2)
\]
for all $t \in [0, T]$, $x, y \in \mathbb{R}^d$.

4. **Initial value** $X(0)$ is $\mathcal{F}_0$-measurable and of finite variance.

The assumptions above give sufficient, but not necessary, conditions for the strong solution of the SDE to exist. In the first manuscript (Section 6), we deal with SDEs that explicitly violate the assumptions. The strong solution to the SDE exists nevertheless, and the central contribution of the work is to propose a numerical method that provides an efficient approximation to the solution.
1.1.2 Quantity of Interest, weak solutions

For applications, one is typically interested in qualitative aspects of the solution. Often these take the form of a *quantity of interest* (QoI). This typically distills into a *payoff function* \( g : \mathbb{R} \to \mathbb{R} \) or *payoff functional* \( g : \Omega \to \mathbb{R} \). In practice, vector-valued QoIs follow as straightforward generalisations of the scalar case. We are interested in the expectation of such payoff functionals. In a more general setting, we might be interested in the distributional properties of the payoff functional evaluated on the solution of the SDE. However, in these cases, the problem can often be reduced to evaluating multiple QoIs, or a vector valued QoI. For example, one might evaluate the expectations \( \mathbb{E}[g(x)], \mathbb{E}[g^2(x)] \ldots \mathbb{E}[g^n(x)] \).

To this end, it is not necessary that the model under study has a strong solution, or that the strong solution is unique. A more relaxed solution, a *weak solution*, is sufficient. We define a weak solution to an SDE as a triple \((X,W, (\Omega,\mathcal{F},\mathbb{P}) , (\mathcal{F}_t)_{t \geq 0})\) such that

1. \((\Omega,\mathcal{F},\mathbb{P})\) is a probability space and \((\mathcal{F}_t)_{t \geq 0}\) its filtration (an increasing set of sigma-algebras such that \(\mathcal{F}_{t_1} \subset \mathcal{F}_{t_2}\) for any \(0 \leq t_1 \leq t_2\))

2. \(X\) is continuous, \((\mathcal{F}_t)\)-adapted \(\mathbb{R}^d\) valued process and \(W\) a \(d\)-dimensional Wiener process on the probability space that generates the filtration \(\mathcal{F}_t\).

3. \(\int_0^t |a(s,X(s))| + |b(s,X(s))|^2 \, ds\) is almost surely finite and \(X(t) = \int_0^t a(s,X(s)) \, ds + \int_0^t b(s,X(s)) \, dW(s)\) almost surely.

Thus, it is sufficient to study the distribution of a functional of the solution. For most problems of interest, the existence of a unique solution in the strong sense can be shown, which is enough to guarantee existence in the weak sense as well. However, in numerical solutions, the convergence rates of numerical approximations can significantly differ for approximating the solution of an SDE in the weak sense and in the strong sense.
1.1.3 Partial differential equations

As mentioned, most practical applications relating to stochastic differential equations are typically related to the expected value of a payoff functional, when evaluated on the state variables of the SDE, or the distribution of outcomes of such evaluation.

For these situations, there exist parabolic PDE’s that prove particularly useful for evaluating quantities of interest in a purely deterministic PDE setting, even though the underlying problem is stochastic in nature. The most important equations in this realm are the Kolmogorov equations: Kolmogorov forward, often called the Fokker-Planck equation and Kolmogorov backward equation, which is a particular case of the Feynman-Kac formula.

**Proposition 1.1.2** (Kolmogorov forward, or Fokker-Planck equation). Let $X(t)$ be the solution to (1.2) and assume that the density $\phi$ of $X(t)$ exists and is twice continuously differentiable. Then the density $\phi$ solves the Fokker-Planck equation in $(t,y) \in [0,T] \times \mathbb{R}^d$:

$$
\partial_t \phi(t,x,y) = \frac{1}{2} \sum_{i,j=1}^d \partial^2_{y_i y_j} \left( \left( bb^T(t,y) \right) \phi(t,x,y) \right) - \sum_{i=1}^d \partial_{y_i} \left( a_i(t,y) \phi(t,x,y) \right),
$$

$$
\phi(0,x,y) = \delta(x - y),
$$

(1.3)

where $\delta$ denotes the delta distribution.

The other relevant PDE in the analysis of SDE’s is the counterpart of the forward equation above, the Kolmogorov backward equation:

**Proposition 1.1.3** (Kolmogorov backward Equation). Let $X(t)$ be the solution to (1.2) and both the drift and the volatility bounded and smooth. Also assume the payoff
function $g$ to be bounded and smooth. Then the value function $u(t,x)$

$$u(t,x) = \mathbb{E}[g(X(T)) | X(t) = x]$$

satisfies the Kolmogorov backward Equation

$$-\partial_t u(t,y) = \frac{1}{2} \sum_{i,j=1}^{d} (bb^T)_{ij}(t,y) \partial^2_{y_iy_j} u(t,y) + \sum_{i} a_i(t,y) \partial_{y_i} u(t,y), \quad (t,y) \in [0,T] \times \mathbb{R}^d$$

$$u(T,y) = g(y). \quad (1.4)$$

A particularly useful generalisation of the Kolmogorov backward Equation is the Feynman-Kac Formula presented below:

**Proposition 1.1.4 (Feynman-Kac Formula).** Let $a, b, g$ satisfy the same conditions as in the Kolmogorov backward Equation above. Furthermore, assume a bounded, continuous functions $h, V : \mathbb{R}^{d+1} \to \mathbb{R}$. Then the function

$$u(t,y) = \mathbb{E} \left[ g(X(T)) \exp \left( \int_t^T V(s,X(s)) \, ds \right) | X(t) = y \right]$$

$$- \mathbb{E} \left[ \int_t^T h(s,X(s)) \exp \left( \int_s^T V(s',X(s')) \, ds' \right) \, ds | X(t) = y \right]$$

solves the PDE

$$-\partial_t u(t,y) = \frac{1}{2} \sum_{i,j=1}^{d} (bb^T)_{ij}(t,y) \partial^2_{y_iy_j} u(t,y) + \sum_{i} a_i(t,y) \partial_{y_i} u(t,y) + V(t,y) u(t,y) - h(t,y), \quad (t,y) \in [0,T] \times \mathbb{R}^d,$$ \quad (1.5)

$$u(T,y) = g(y), \quad y \in \mathbb{R}^d. \quad (1.6)$$
I.e. the Feynman-Kac formula generalizes the Kolmogorov backward equation to the case where one is interested in computing a *weighted* expectation of \( g(X(T)) \) with the weighting term depending on the trajectory of the solution in the interval \([0, T]\). In financial applications, this weighting can be understood as a *discount factor* that enables comparisons of cash flows happening at different times. Likewise, the Feynman-Kac formula offers a way to compute an expectation of an average of a function \( h \) over the realisations of an SDE. This is particularly useful when one is interested in valuing non-instantaneous, stochastic cash flows that take place over a prolonged period with the magnitude of the cash flow being given by \( h \). The Feynman-Kac formula also offers a stochastic interpretation of a rather general parabolic PDE, that allows the computation of point values of the PDE using simulation methods, i.e. sampling simulated realisations of the SDE.
Chapter 2

Financial applications

Arguably the best-known application for SDEs is that of finance. As the prices of financial securities are known to be notoriously hard, if not impossible, to predict, the SDEs provide a framework to describe the probabilistic evolution of such quantities, or state variables, as time passes. The earliest of such works dates back to the early 20th century and Bachelier, who analyzed prices of options assuming that the prices of financial securities, in his case the prices of common stock, follow the normal distribution.

Assuming a distribution of possible outcomes for a price of a given security, the question of fair price arises. Assuming a contract whose payoff is linked to the performance of a financial security, what is the price for such contract such that in expectation, the contract pays off exactly that price. Questions about such fair value pricing became much better formalized in later years with the advent of Black-Scholes equation and its extensions. The question of evaluating expected, or average returns of a contract also explains the heavy use of the Kolmogorov-backward equation and Feynman-Kac formula (Equations (1.4) and (1.6)) introduced in the previous chapter.

Of course, the use of SDEs is not only limited to modeling randomly evolving stock prices, or even prices of other securities. They have been heavily used in modeling the evolution of interest rates, currency exchange rates, or even the evolution of non-observable parameters such as modelling the volatility parameter $b$ of \( \text{(1.2)} \) itself as a random process, given by an SDE. The SDEs have also successfully been used to model infinite-dimensional quantities, such as a yield curve specified not only as a
scalar value but a scalar quantity defined on the positive real line.

Likewise, the SDE approach has been generalised to various quantities of interest beyond simple averages of a payoff function evaluated at the state variable at a given time $T > 0$. For example, the likelihood of ruin (also called the lookback value in the option pricing jargon) represented by

$$E\left[1_{\inf\{0 \leq t \leq T : X(t) < c}\}\right] \quad (2.1)$$

, with $1$ denoting an indicator function,

$$1_A(x) = \begin{cases} 1, \text{if } x \in A \\ 0, \text{otherwise} \end{cases},$$

is naturally of interest. For a discussion on the extrema generated by an SDE and its approximation with multi-level methods discussed later in Section 3.2.1.3, we refer to [26, Chapter 5.2]

Also, the construction of the full density, or rather the Cumulative Density Function (CDF) of the process at a fixed time $T$ can be of interest. This can be achieved solving

$$E\left[1_{X(T) > c}\right] \quad (2.2)$$

for a sequence of values for $c$ and interpolating for the full CDF of $X(T)$. For the interested reader, we recommend the analysis of constructing the CDF in [28].

Below, we give a flavor of the problems that have been analyzed using the mathematical concepts relevant to the thesis. Following the spirit of Article II, we devote Section 2.1 to different phenomena being modelled and a variety of stochastic forward models used. The discussion is followed by Section 2.2 that is devoted to the con-
cept of risk-neutral probability measure. This concept is essential when the models of Section 2.1 are employed to price financial derivative instruments such as options. In Section 2.3, we briefly discuss the different kinds of QoIs or payoffs that are often considered within the field. The discussion is not particularly exhaustive, and we refer the reader to the variety of works written on the topic \cite{60,62}.

2.1 Models

The literature of SDE models widely used in financial modeling is wide in both the methods of modelling as well as the quantities being modelled. On the one hand, empirical evidence has called to question the models used in the early days of mathematical finance such as the Bachelier and Black-Scholes models. Generalisations and extensions have been made to the early models to account for phenomena such as discontinuous dynamics, the frequent occurrence of large fluctuations and heteroscedasticity. We discuss the shortcomings of the traditional Black-Scholes model in Section 2.1.1 and proceed to introduce more elaborate models in the subsequent sections.

In addition to the models themselves, their field of application has widened significantly. In addition to pricing financial securities such as shares of stock, SDE models have been used to model rates of exchange for currencies and abstract quantities like volatility indices. One particularly interesting and complex object to model is interest rates. In the case of interest rates, the need to consistently model the dynamics of the yield curve naturally arises: One may borrow money for a fixed period, say for a year, or only for six months and refinance one’s debt for yet another six months. Under ideal conditions, one would expect that these choices yield the same result on average, i.e. arbitrage is not permitted. The Heath-Jarrow-Morton framework provides a consistent framework for modeling the interest rates for all possible expiration dates of bonds. Later, similar approach has been adopted for the modeling of commodity
futures that also have multiple possible expiry times.

2.1.1 Local volatility models

Arguably the first theoretically justified and consistent option pricing model, the Black-Scholes model, features many desirable qualities. Taking essentially one free parameter, the volatility $\sigma$ (the drift parameter $\mu$ turns out to be of secondary importance in derivative pricing due to reasons explained in Section 2.2) it gives rise to the dynamics of a single stock price through the SDE

$$dS(t) = \mu S(t) \, dt + \sigma S(t) \, dW(t). \quad (2.3)$$

Due to the simplicity of the model and the fact that the logarithm of the state variable $S(t)$ is normally distributed, the prices of options for any strike $K$ and maturity $T$ reduce to the numerical evaluation of the error function that can be carried out with arbitrary precision with negligible computational complexity.

Given the fixed parameters $T$ and $K$ the Black-Scholes model gives an easy to evaluate mapping from the scalar volatilities $\sigma$ volatilities to option prices $C(\sigma, T, K)$, where

$$C(\sigma, T, K) = \mathbb{E}_Q \left[ e^{-rT} (S(T) - K)^+ | S(0) = S_0 \right], \quad (2.4)$$

and $r \in \mathbb{R}$ is understood as the instantaneous interest rate and $\mathbb{Q}$ the so-called risk-neutral probability measure discussed later in Section 2.2. It can be shown that the function $C$ is increasing in $\sigma$ and continuous and thus invertible. Equally, one may define the inverse mapping $\sigma_i (C_{T, K}, T, K)$. Now if one assumes the model (2.3), it is a sensible question to ask how the one free parameter $\sigma$ should be set so that it best corresponds to observed reality, i.e. prices. One obvious choice is to study the past evolution of the stock price and to infer how large the fluctuations in the logarithm
of the asset price have been. The resulting estimate is so-called historical volatility. The method of estimating the historical volatility is straightforward, but it also has one severe shortcoming inherent in the name: it is not forward-looking. Historical volatility remains unchanged even if the market consensus about the magnitude of the future uncertainty is altered dramatically. To achieve a forward-looking estimate, one may determine the implied volatility: to determine which value of $\sigma$ best produces the prices of traded securities visible to every market participant. Doing the simple inversion $\sigma_i$ for a set of options, though, one will soon observe that the value $\sigma_i$ is dependent on which expiry date $T$ and strike $K$ options one uses for the inversion. From this, we may draw the conclusion that trying to model the dynamics of stock prices using a simple scalar volatility $\sigma$ is insufficient. The market consensus (i.e. prices) typically predicts that extreme events are relatively more likely than the Black-Scholes model implies. Also, one typically observes an implied volatility that is decreasing in time to maturity $T$. For a discussion on the behavior of this volatility surface we refer the reader to [43] and [24].

Despite this evident shortcoming, the Black-Scholes model remains an important concept in the field of finance, and many numerical methods are benchmarked using it as a test case. Indeed, the multi-dimensional variant is the central example studied in Article III in Chapter 8. Many of the subsequent advances in modelling asset prices are in one way or another extensions and generalizations of the long-known and well-studied Black-Scholes model (2.3), with additional terms introduced to account for the inherent insufficiency of the Black-Scholes model. Below, we discuss some of these extensions.

The first, and arguably the simplest, extension to the Black-Scholes model is choosing the volatility to be a function of both the price and time, $\sigma = \sigma (t, S(t))$. This way, it is possible to produce option prices consistent to practically any observed prices [19]. Some times accurate prices of options are not available, or there is a
need to limit oneself to a particular subclass of local volatility models for ease of computation. For this purpose, there is a multitude of parametric local volatility models able to produce desired distributional characteristics. One of the perhaps best known and widely used ones is the Constant Elasticity of Variance (CEV) that introduces an additional exponent $\gamma$ and defines the local volatility as

$$\sigma(t, S(t)) = S^\gamma(t) \sigma.$$ 

Another local volatility model commonly used to model short-term interest rates is given by the Cox-Ingersoll-Ross (CIR) model. The CIR model features the volatility structure familiar from the CEV model, with mean reversion:

$$dY(t) = (\xi Y(t) + \zeta) \, dt + \nu \sqrt{Y(t)} \, dW(t).$$

In essence, Cox Ingersoll and Ross defined a model in which the short-term variance of the short rate $Y(t)$ is given by a CEV model with the critical exponent $\gamma = \frac{1}{2}$. However, the drift is such that the evolution always tends towards the long-term average $-\frac{\zeta}{\xi}$.

The mean reverting drift of the CIR model is a feature well-known and widely used in the study of SDEs. The corresponding volatility process,

$$dY(t) = \xi (Y(t) + \zeta) \, dt + \nu dW(t), \tag{2.5}$$

was introduced decades earlier and is known as the Ornstein-Uhlenbeck process. It is one of the few SDE models for which a closed form solution has been given. In the financial literature, the process is called Vasicek model, and used, like the CIR model, to model short-term interest rates with the mean level $\zeta$ often time-dependent.
2.1.2 Stochastic volatility

Another class of SDEs that produces desired volatility smiles is the class of stochastic volatility models. These two or higher dimensional models feature a one-dimensional process for the state variable and one or more components that form a separate process for the instantaneous volatility. Since the instantaneous volatility can assume higher values, stochastic volatility models not only give rise to the desired volatility smile but can also reproduce heteroscedasticity, the phenomenon of fluctuations in the state variable appearing to be clustered. This feature is commonly observed in the markets [17, 41].

Probably the most widely used stochastic volatility model is that of Heston, defined as

\[
\begin{align*}
    dS(t) &= \mu S(t) \, dt + \sqrt{v(t)} S(t) \, dW_1(t) \\
    dv(t) &= \kappa (\theta - v(t)) \, dt + \sigma \sqrt{v(t)} \, dW_2(t)
\end{align*}
\]

with the Wiener processes \( W_1 \) and \( W_2 \) having an arbitrary correlation coefficient \(-1 < \rho < 1\). Essentially, the volatility is modelled by a CIR process correlated with the asset price process, which is given as a generalised CEV process with exponent \( \gamma = \frac{1}{2} \). Due to the precise characteristics of the model, the characteristic function, or the Fourier transform of the density function is known exactly [34]. This feature allows expressing the price of European Options as a one-dimensional integral – a feature typically seen in Lévy models discussed in the sequel.

Another widely used Stochastic volatility generalisation of the CEV model is the SABR model, so named for the stochastic nature and the three parameters, \( \alpha, \beta, \rho \)
used to parametrise it. In the SABR model, the asset price dynamics is given as

\[ dS(t) = a(t) S^\beta(t) \, dW_1(t), \]

\[ da(t) = \alpha a(t) \, dW_2(t). \]

The parameter $-1 < \rho < 1$ not explicitly visible in the formulation is understood as the correlation coefficient between the two Wiener processes, $W_1$ and $W_2$. I.e. the asset price is modelled by a CEV process with exponent $\beta$ and the volatility process is defined by the Black-Scholes model. Like in the case of the Heston model, an arbitrary value for the correlation of the Wiener process is permitted [30].

2.1.3 Pure jump and jump diffusion models

Another, complementary way of addressing the shortcomings of the Black-Scholes model is introduction discontinuities or jumps. Whereas in stochastic volatility models one achieves the feature of sudden fluctuations in excess to those expected by the normal distribution, by instantaneous high value of the stochastic volatility, in Jump modelling rapid changes in the state variables are achieved by introducing a Poisson process that models the arrivals of sudden, large fluctuations in one or more state variables. For a review of jump processes and their use in financial modelling, we refer the interested reader to [59]. For the purpose of our this thesis, we focus mainly on a particular subclass of jump processes and jump-diffusion processes called Lévy processes.

For a stochastic process $X$ featuring jumps, the difference $|X(t + \epsilon) - X(t)|$ for $0 \leq t < t + \epsilon \leq T$ is not controlled by the magnitude of $\epsilon$. Assuming the existence of a transition density $\psi$, one can write for a Markov process $X$ in $\mathbb{R}$, the expectation
\( u(t, x) = \mathbb{E}[g(X_T) | X(t) = x] \) as

\[
    u(t, x) = \int \frac{P(X(t + \epsilon) \in [z, z + dz] | X(t) = x)}{\psi(t, x, z)} \, u(t + \epsilon, z) \, dz.
\]

Since the transition density \( \psi(t, x, z) \) can have support for \( |x - y| \) large, even in the limit of \( \epsilon \to 0 \), the resulting backward equation has non-local integral terms to account for the discontinuities in the dynamics.

Due to a theorem by Courrège \[15\] the generator \( \mathcal{L} \) of a pure jump process can be written as

\[
    (\mathcal{L}f)(t, x) = \int_{\mathbb{R}^d \setminus \{0\}} (f(x + y) - f_x(x) - \phi(x, y)(y_i - x_i)f(x)) \, \mu(t, x, dy)
\]

for any \( f \in C_c^\infty(\mathbb{R}^d) \), with \( \phi \) being a local unit, i.e.

1. \( \phi(x, y) = 1 \) in a neighbourhood of the diagonal \( D = \{(x, x) : x \in \mathbb{R}^d \} \).
2. for each compact \( K \subset \mathbb{R}^d \), the mapping \( y \to \phi(x, y), x \in K \) has compact support.

and the measure \( \mu \) is a Lévy kernel, i.e. a Borel measure on \( \mathbb{R}^d \) such that

1. the mapping \( x \to \int_{\mathbb{R}^d \setminus \{x\}} |x - y|^2 f(y) \, \mu(x, dy) < \infty \) is Borel measurable and locally bounded for each locally continuous, compactly supported \( f \).
2. for all \( x \in \mathbb{R}^d \) and for every neighbourhood \( V_x \) of \( x \), \( \mu(x, \mathbb{R}^d \setminus \{V_x\}) \)

Backward equations featuring such integral terms are often called partial integro-differential equations or PIDEs. \[14,55\]

Perhaps the simplest case of a jump process would be one where the jump measure \( \mu \) is finite. In that case, one may select \( \phi(x, y) = \delta(x - y) \). For example, to describe a pure jump process that features jumps arriving as a Poisson process with intensity
with the jumps being of unit size with equal probabilities in each direction on the real line. For such a process, \( \mu(x, dy) = \lambda \delta(x-y+1) + \delta(x-y-1) \) This gives

\[
\mathcal{L}f(x) = \frac{\lambda}{2} (f(x+1) + f(x-1) - 2f(x))
\]

and the Kolmogorov backward equation (1.4) for the conditional expectation \( u(t, x) = E[g(X_T) | X_t = x] \) in the pure jump framework reads

\[
\partial_t u(t, x) = (\mathcal{L}u)(t, x) \\
u(T, x) = g(x)
\]

for \( (t, x) \in [0, T] \times \mathbb{R}^d \).

The operator above is linear as in the case of Kolmogorov backward equation corresponding to the Black-Scholes model (2.3). When modelling processes that have a discontinuous jump as the naive example above, as well as a continuous component, the resulting generator is a mere sum of the generators corresponding to the diffusion and pure-jump processes respectively.

In the discussion here, we have treated time \( t \) as a continuous variable. It should be noted that in practice, this time \( t \) is measured in terms of the markets. I.e. time evolves when markets are open, and in the night time, physical time passes, but market time does not. However, information relevant to the prices does occasionally arrive in the night time causing fluctuations in the prices. In terms of market time \( t \), these fluctuations appear as sudden movements, jumps. The need to include discontinuous dynamics in the realm of financial models was acknowledged first by Merton [48] and the literature has expanded to include a variety of pure jump models (see [4, 12, 46]) as well as models that feature a continuous and a jump component (see, for example, [25, 12]). We note that in addition to introducing discontinuous dynamics, jumps can also give rise to the volatility smile mentioned earlier in Section 2.1.1.
In this thesis, we discuss a specific kind of pure jump and jump diffusion models in one state variable, namely those called Lévy models, which we discuss below.

2.1.4 Lévy models

One particularly useful and widely used class of models for financial modeling is that of Lévy models (see \cite{2}). These models are pure-jump and jump-diffusion models that have useful characteristics, the most important one of which is the Lévy-Khintchine formula, which allows to transform the PIDE into an ODE in Fourier space, reducing the problem of solving a PIDE into a simple Fourier transform integral, besides Article II, this approach has been widely used in a variety of financial topics (\cite{13,22,44,45,47}).

A Lévy process in $\mathbb{R}^d$ can be characterised as the triple $(\gamma, \sigma, \nu)$, with constant drift $\gamma \in \mathbb{R}^d$, constant volatility $\sigma \in \mathbb{R}^{d \times k}$ and the Lévy jump measure $\nu$:

$$\nu(dy) = \mu(x, dy).$$

In other words, Lévy processes are jump-diffusion processes such that the drift, volatility and the jump measure are independent of the process, or translationally invariant.

In introducing the Fokker-Planck or Kolmogorov forward equation, we assumed that the process has a density $\phi$. Even if such a density exists, solving it through PDE methods may be difficult especially in high dimension. The central appeal for using Lévy models is that the Fourier representation of the density is explicitly known through an identity known as the Lévy-Khintchine formula.

**Lévy-Khintchine formula** Let a Lévy process $X(t)$ be define by its characteristic triplet $(\gamma, \sigma, \nu)$. Then the characteristic function $\varphi$ of the process is given
by

\[ \varphi_{X(t) \theta} = \mathbb{E} \left[ \exp \left( i \theta \cdot X(t) \right) \right] \]
\[ = \exp \left( t \left( i \gamma \cdot \theta - \frac{\theta^T \sigma \sigma^T \theta}{2} + \int_{\mathbb{R}^d \setminus \{0\}} e^{i \theta \cdot y} - 1 - i \theta \cdot y 1_{|y| < 1} \nu(dy) \right) \right). \quad (2.6) \]

For a proof of the formula, we refer the reader to [2]. With the notable exception of the Heston model, the characteristic function is typically not known for models other than the Lévy models. For most of the Lévy models there is no need to evaluate the formula, as the characteristic functions are available in the literature. Using the procedure in [31], the expected payoffs can be evaluated using the Parseval identity. This property has also been extended to higher dimension for spread option pricing [40].

### 2.1.5 Yield curve models

So far, we have discussed models in a low number of dimensions. One class of SDE models of particular interest in financial applications that are particularly interesting for computations, however, is the set of yield curve models. We have introduced multiple models that are originally introduced to model the short-term interest rates. For fixed income investments, however, one is often interested in modelling the whole yield curve: one needs a probabilistic model for what is the interest rate when committing to lend a given sum at time \( t_1 \) to be repaid at a time \( t_2 \) for arbitrary \( 0 < t_1 < t_2 \).

Towards this end, we introduce the concept of a yield curve. One particular way to model such an object is through a Heath-Jarrow-Morton (HJM) model

\[ df(t, s) = \mu(t, s) dt + \Sigma(t, s) dW(t) \quad (2.7) \]

with the drift and volatility functions \( \mu(t, s) \) and \( \Sigma(t, s) \) being both \( \mathcal{F}_t \)-adapted. \[33\]

In practice, the HJM model allows one to model complicated dynamics of the
forward curves, through defining a few fluctuating modes $\beta^{(j)}$, $j = 1, 2, 3...$ and writing the dynamics of (2.7) as

$$df(t, T) = \alpha(t, T) \, dt + \sum_{j=1}^{d} \sigma_j(t, T) \, dW_j(t)$$

with the modes representing, for example, the overall level of $f$, the slope of it, the convexity, etc. In extreme cases, and for complicated markets, as many as a dozen modes might be required to model the forward curve satisfactorily [7]. As pointed out in [10, 11], the framework also allows for an infinite number of nodes, as studied in [3, 5].

### 2.2 Risk-neutral measure

In our discussion of financial models and pricing derivative securities, we have left implicit the probability measure. For all of our contributions and practically all derivative pricing problems, the relevant pricing measure is not that naturally imposed by the forward model such as (2.3), but an equivalent martingale measure, also referred to as pricing measure or risk-neutral measure.

From a mathematical standpoint, the distinction is often omitted, and in the works included in this dissertation, we often assume that the models under study are already such that the dynamics of the model gives rise to the risk-neutral measure. However, in calibration and interpretation of the results, the distinction is of importance. We describe the key concepts central of the pricing measures here.

Tasks of portfolio management are typically carried in the physical measure, i.e. the measure specified by the forward model. For derivatives pricing, however, the pricing of the derivative security is closely linked to the cost of hedging the derivative asset. We are interested in the question of what it costs to form a replicating portfolio that has exactly the payoff of the derivative being priced.
First, let us give an example of a derivative pricing problem in a local volatility problem that elucidates the concept of risk-neutral pricing measure. After that, we briefly discuss the concept in a slightly more general setting.

Assume that we are interested in pricing a derivative product that returns \( g(x) \) to the holder at time \( T \). To facilitate the analysis, assume that

1. Transaction costs are zero
2. Short-selling is allowed
3. The asset prices change continuously
4. One can take an arbitrary position (any real number) in the underlying security

Further, let us assume that the asset dynamics is given by

\[
dX(t) = \mu(t, X(t)) \, dt + \sigma(t, X(t)) \, dW(t),
\]

\[
X(0) = x_0.
\]

Now assume that there is a derivative paying the payoff \( g \) at the end of the contract period \( t = T \). Also, assume that the price of this derivative \( u \) is given as a function twice differentiable in space and once in time. Assume one sells such a derivative for its fair price \( u \) and invests the received money into the stock on which the derivative is written, and the rest into the bank to accrue interest. Let the amount of money invested in the stock be \( u_x \) and the remaining \( u - u_x \) is invested in the bank account, which compounds continuously as

\[
\text{dB}(t) = rB(t) \, dt.
\]

Let us consider the case of \textit{delta hedging} of the option, so-named for the options pricing jargon in which the derivative, or sensitivity, of the option with respect to the
value of the underlying, is called the delta of the option. Let us look at the value of a portfolio $Y$ that contains one option sold short, and a hedging position of $u_x$ units of the stock and $u - u_x$ units of cash in the bank. This portfolio evolves as

$$\frac{dY(t)}{dt} = -\left(u_t + u_x\mu + \frac{\sigma^2}{2} u_{xx}\right) dt - u_x\sigma dW(t) + u_x \mu dt + u_x \sigma dW(t)$$

$$+ r(u - u_x) B(t) dt.$$

$$= -\left(u_t + \frac{\sigma^2}{2} u_{xx}\right) + r(u - u_x) dt.$$

I.e. the evolution of the portfolio is deterministic: the terms containing $dW(t)$ cancel out. To preclude arbitrage all deterministic quantities must evolve similarly in time. If this was not the case, one might buy the one asset and sell another and enjoy arbitrarily high profits arising from the differential. The portfolio $Y(t)$ must therefore satisfy

$$\frac{dY(t)}{dt} = r\left(-u + u_x S(t) + (u - u_x) B(t)\right) dt.$$

We thus have that

$$u_t + \frac{\sigma^2}{2} u_{xx} - ru + u_x S(t) = 0.$$

Noting that trajectories of $S$ are dense in $\mathbb{R}$ gives the Black-Scholes equation in $\mathbb{R}$

$$u_t + \frac{\sigma^2}{2} u_{xx} - ru + u_x x = 0. \quad (2.8)$$

It is noteworthy that in the expression above, the drift of the process, $\mu$ does not appear at all. In fact, using Feynman-Kac formula, we interpret the function $u$ as a
discounted expectation of a quantity modelled by another SDE:

\[
\begin{align*}
    dX(t) &= r(t) X(t) \, dt + \sigma dW(t), \\
    X(0) &= x_0, \\
    u(t, x) &= \mathbb{E} [\exp (-r(T-t)) g(X(T)) | X(t) = x].
\end{align*}
\] (2.9) (2.10)

the dynamics (2.9) is the dynamics of the Black-Scholes under the risk-neutral measure. The change between the original, physical and the risk-neutral measure is enabled by the Girsanov theorem of measure change. However, for the purposes of this work, we, unless otherwise noted, study models in which the probability measure in question is already the risk-neutral measure, or in which the dynamics is risk neutral, i.e.

\[
    \mathbb{E} [e^{-r(t-s)} X(t) | \mathcal{F}_s] = X(s), \quad 0 \leq s \leq t \leq T. \tag{2.11}
\]

We note that in the case of jump-diffusion models, the assumption on continuous changes is explicitly violated. In these cases, the risk-neutral measure will not be unique. However, a particular risk-neutral measure can be achieved by adjusting the drift so that (2.11) is satisfied, and this choice of drift is discussed in more detail in the Section.

### 2.3 Payoff

There are many kinds of relevant properties that one may seek to evaluate. These properties are typically given in the form of a payoff function or a payoff functional such that the quantity of interest is the expected value of the payoff. Depending on the circumstances, the payoff can be either a simple function evaluated at a state variable of the SDE at a given time, or a more elaborate payoff functional such as
the ones evaluated in Asian, or American options pricing. In this Section, we list some of the common payoff types and their relevance, putting the contributions in the research manuscripts in context.

2.3.1 Plain vanilla

The archetype payoff type, the put and call options, that can be priced with the Black-Scholes equation (2.8) are widely traded on the market. The call option gives the holder of the option the right to buy the underlying asset at a pre-determined strike price $K$. Naturally, if the price of the option is below $K$, a rational agent will not choose to exercise her right.

However, if the price of the underlying security is above $K$, the holder of the option will exercise, making the difference of the price of the underlying asset and the strike price, giving the call option the payoff

$$g_{\text{call}}(x) = (x - K)^+.$$ 

Likewise, a put option works identically, but giving the holder the opportunity to sell, rather than buy the underlying asset:

$$g_{\text{put}}(x) = (K - x)^+.$$ 

Sometimes it is more convenient to write the dynamics of the logarithm of the asset price rather than the price itself, as is the case in the second research manuscript as well as many other works [44]. In these cases the payoffs are given as

$$g_{\log\text{-call}}(x) = (e^x - K)^+,$$

$$g_{\log\text{-put}}(x) = (K - e^x)^+.$$  

(2.12)
The call and put options are widely quoted on various options markets and thus considered the plain vanilla derivative products, as opposed to the more exotic path-dependent, and over-the-counter (OTC) products are custom tailored for particular investment needs. For modelling purposes, these openly quoted options serve as a good tool for model calibration, as being able to observe option prices for various expiry times $T$ and strikes $K$ gives a lot of information about the risk-neutral dynamics that equilibrium between market participants implies. These options also serve as tools to replicate or hedge more complex structures, such as the cash flows of large corporations. For example, knowing precisely the prices of derivatives of all strikes $K \in \mathbb{R}$ allows one to extract the whole risk-neutral distribution of the future prices at the time of the expiry of the options \cite{056}.

A particular feature of the European put and call options is the ability to replicate one using the other. This property is called the put-call-parity and is useful in a variety of cases, for example when one of the options has practically zero value, which might cause instabilities in numerical solutions.

Denote the price of the put option as $P$ and the call option by $C$ we can determine that the payoff of the difference of a European put and a European call yields the payoff

$$g(x) = (K - x)^+ - (x - K)^+ = 1_{x<K} (K - x) - 1_{x\geq K} (x - K) = (K - x).$$

I.e. the portfolio containing a long put and a short call at the same expiry and strike yields a payoff of $K - x$ regardless of the events in the market. Similarly to the proof of the Black-Scholes model, we determine that all deterministic portfolios have the same dynamics to avoid arbitrage and conclude that

$$P - C = \exp \left( - \int_0^T r(s) \, ds \right) (K - x).$$
In Article II of Section [?], we introduce a damping parameter $\alpha$, such that log-put or log-call payoff in (2.12) multiplied by a damping factor $e^{\alpha x}$ is in the function space $L^1(\mathbb{R})$ so that the Fourier-transformed of this new, modified, payoff exists. It turns out that the Fourier-transformed, damped payoffs have the same functional form, with the only difference being in the values of the damping parameter for which the transform is defined. Thus, the discussion of the optimal parameter selection in Article II can be seen as a Fourier-space manifestation of the put-call parity. The appropriate choice of numerical parameters can be interpreted as the question of whether it is more feasible to evaluate put or call prices, as one will naturally give the other [44, Remark 7.4].

2.3.2 Binary

Another simple payoff is the binary payoff, that is simply defined as the indicator function of a measurable set $A$. Binary options pay off a set amount in case the event $A$ occurs and nothing if it does not. Despite the notoriety of some fraudulent activity, these securities serve a purpose on the financial markets and various binary options are also quoted on the liquid, open markets such as NYSE and CBOE.

The payoff of these binary options (also known as all-or-nothing options, digital options or fixed return options (FRO)) takes the simple form

$$g(x) = 1_{x \in A},$$

with typically $A = \{x \in \mathbb{R} : x < K\}$ or $A = \{x \in \mathbb{R} : x > K\}$.

Due to the non-continuity of the payoff function, the pricing of binary options is often more challenging than familiar puts and calls. For example, when using Fourier methods on Lévy processes like in our second article, the discontinuity of the payoff results in a slower decay of the Fourier coefficients of the payoff. This, in turn, makes
the numerical Fourier inversion more prone to errors than the evaluation of classical options.

Another manifestation of this non-smoothness is evident in the multilevel Monte Carlo methods, that traditionally require smoothness off the payoff and require additional work when confronted with non-smooth payoffs [27]. However, when analyzed properly, the knowledge of all the binary option values

\[ B_K = \mathbb{E}[1_{X_T > K} | X(0) = x] \]

is equivalent to knowing the full risk-neutral density of \( X_T \).

### 2.3.3 Averaging payoffs

At the other extreme to the binary options, the concept of averaging payoffs and Asian options deserve mention. For example, in case of the Asian options, the payoff is defined by a functional of the whole trajectory, and one is interested in the average trajectories in time

\[ \mathbb{E}\left[g\left(\int_0^T X(s) \, ds\right) | X(0) = x\right]. \]

These options are of particular interest to investors and producers who produce or consume the underlying asset on a continuous basis. From a numerical analysis standpoint, these options offer an opportunity to exploit the smoothness of the payoff compared to the point-wise evaluation of the state at a time \( T \).

### 2.3.4 Stopping and hitting times

One payoff functional that is of particular interest in finance, and is more closely discussed in the third work in this thesis is the one related to stopping times. For \textit{American} options, the holder of the option can exercise the option at any time leading
up to the expiration of the contract at time \( T \). The time at which an agent chooses to exercise her option is called a stopping time, often denoted by \( \tau \) in the financial literature.

A stopping time \( \tau \) is a random variable \( \tau : \Omega \rightarrow [0, T] \), such that

\[
\{ \omega \in \Omega : \tau (\omega) \leq t \} \in \mathcal{F}_t,
\]

i.e. it is possible to determine whether the stop has happened before \( t \) using only the information generated by the process up to time \( t \). Generalizations where the stopping times that can be unbounded are possible, and are often relevant for the study of options that never expire, i.e. perpetual options. However, we shall omit them here.

The pricing of American options using a model is based on the assumption that the holder of the option exercises it in an optimal fashion given the model. I.e. the pricing problem becomes

\[
u_A (t, x) = \sup_{\tau \in \mathcal{T}_t} \mathbb{E} \left[ e^{-\int_t^\tau r(s) ds} g(X(\tau)) \mid X(t) = x \right],
\]

\[
\mathcal{T}_t = \{ \tau : \Omega \rightarrow [t, T] : \{ \omega \in \Omega : \tau (\omega) \leq s \} \in \mathcal{F}_s \}.
\]

In other words, one is evaluating the maximal expected discounted value that the holder can extract if she is given an option that has not yet been exercised at time \( t \), when the state of the system is \( x \), under the appropriate pricing measure. Conversely, the price of an American option can be seen as the maximal expected loss that the underwriter of said option can lose.

Since for all the models of interest here, the dynamics of the system is Markovian, i.e. it only depends on the time \( t \) and the state \( x \), the decision of whether to exercise or not depends on those variables. Thus, for our work, the concept of a hitting time can be used interchangeably with the concept of the stopping time.
Definition 2.3.1 (Hitting time for an SDE). Let $X$ be a stochastic process in $\mathbb{R}^d$ and $A \subset \mathbb{R}^{1+d}$. Then the mapping

$$
\pi_A = \inf \{ t \in \mathbb{R} : (t, X(t)) \in A \}
$$

is called the hitting time of $X$ into $A$.

The price function of the American pricing problem naturally gives rise to a hitting time. Since the control in the dynamics is straightforward, the domain $[0, T] \times \mathbb{R}$ can be decomposed into two distinct subsets, the subset in which it is optimal to exercise, and the subset where it is not. The former of these gives rise to the optimal stopping time, which is also a hitting time, setting

$$
A = \{ (t, x) : u_A(t, x) = g(x) \}
$$

gives the optimal stopping time as a hitting time. At terminal time $t = T$, the value of the option is worth its intrinsic value:

$$
u_A(T, x) = g(x),$$

for $x \in \mathbb{R}^d$. This results in the hitting time $\pi_A$ being always bounded. We exploit this parallel between stopping times and hitting times in our third manuscript, where we solve a different optimal stopping problem $\pi_A$ instead of the original, high-dimensional, optimal stopping problem. However, this different problem still defines a region where the value function and the payoff function coincide, giving us a valid hitting time, which is also a valid stopping time.

For European options it is possible that the value of the option is exceeded by the payoff function. This is not possible for American options, since it is always possible
to extract at least the intrinsic value of the option, as determined by the payoff

\[ u_A(t, x) \geq g(x), \quad \forall (t, x) \in [0, T] \times \mathbb{R}^d. \]

In the continuation region where \( u_A(t, x) > g(x) \), the dynamics is that of the European option and the usual Black-Scholes equation \([2.8]\) holds giving us that for American options

\[ (u_A(t, x) - g(x)) \left( \partial_t u_A + \frac{\sigma^2}{2} \partial^2_{xx} u_A - ru_A + x \partial_x u_A \right) = 0, \]

for \( x \in \mathbb{R}^d \) and \( t \in [0, T] \).

Combining the holding and exercise regions, we have the formulation for the price \( u_A \) of the American option

\[ u_A(t, x) - g(x) \geq 0, \]

\[ \left( \partial_t u_A + \frac{\sigma^2}{2} \partial^2_{xx} u_A - ru_A + x \partial_x u_A \right) \leq 0, \quad (2.13) \]

\[ (u_A(t, x) - g(x)) \left( \partial_t u_A + \frac{\sigma^2}{2} \partial^2_{xx} u_A - ru_A + x \partial_x u_A \right) = 0 \]

for \( x \in \mathbb{R}^d \) and \( t \in [0, T] \). This is the strong form of the equation one is interested in, when solving American options \([1]\), and the fundamental object of our inquiry in the third work of this thesis.
Chapter 3

Numerical methods

Having introduced the relevant equations (1.4) and (1.6), the question of efficient solution methods to those equations arises. In this Section, we first discuss in 3.1 some models exhibiting known closed-form solutions, which provide useful toy models to analyze problems. We then discuss briefly the two main categories of solution methods used to solve the PDEs. In 3.2 we discuss the Monte Carlo simulation methods analyzed in more detail in Article I of Section 6. In 3.3, we discuss the solution of PDEs using the PDE solution strategies known from various fields of science and engineering beyond the scope of this thesis.

3.1 Exact methods

Few models exhibit a known closed-form solution, and even fewer are of practical use in the field of finance. Most of these are either Gaussian or can be transformed into Gaussian using a change of variables. For a listing of explicitly solvable models, we refer reader to [39, Chapter 4.4]

The most relevant exactly solvable models for this dissertation are mainly the Ornstein-Uhlenbeck model, the closely related Bachelier model, and the Black-Scholes model. The solution for the vector-valued version of the Ornstein-Uhlenbeck process (2.5),

\[ dY(t) = A(Y(t) + m)dt + BdW(t), \]
is given as

\[ \mathbf{Y}(t) = \exp(\mathbf{A}t) \mathbf{Y}(0) + (1 - \exp(-\mathbf{A}t)) \mathbf{m} + \mathbf{B} \int_0^t \exp(-\mathbf{A}(t-s)) \mathrm{dW}(s). \]

From this, the Bachelier model solution follows as a special case for which \( \mathbf{A} = r\mathbf{1} \) and \( \mathbf{m} = 0 \).

The third relevant model, the Black-Scholes model, is most conveniently described as the exponential transform of the Ornstein Uhlenbeck process. Selecting \( \mathbf{A} = 0 \) and \( \mathbf{m} = r\mathbf{1} \), one gets a model for the logarithm of the Black-Scholes asset price in the risk-neutral measure.

### 3.2 Forward simulation, Monte Carlo methods

Given the definition of an Itô integral, arguably the most natural way to evaluate numerically expectations of an integral is that of a forward-Euler simulation sampling (1.1) directly.

This approach has some inherent benefits starting from the ease of implementation and versatility. Using Monte Carlo simulation, one can often solve problems with minimal regularity assumptions on the forward model as well as the payoff functional. In many cases, Monte Carlo simulation can also be used to avoid the so-called curse of dimensionality, where the computational cost of the solution method scales exponentially in the number of dimensions. However, as a tradeoff, the performance of such methods is fundamentally limited by the Central Limit Theorem, meaning that in order to solve the problem subject to computational error tolerance \( \epsilon_T \), a computational cost of at least \( \epsilon_T^{-2} \) is incurred. Below we outline some central aspects of estimating QoIs using an estimator based on Monte Carlo simulation. For a comprehensive review on the topic, we refer the reader to the numerous works written on Monte Carlo methods in general [32, 53, 54], and in the field of computational finance
In particular \[8\,[29],[35].\]

In forward-Euler methods, we approximate (1.2) on a mesh \(\{t_n\}_{n=0}^N\) as

\[
X(t_{n+1}) = X(t_n) + a(t_n, X(t_n)) \Delta t_n + b(t, X(t_n)) \Delta W_n
\]

\[
\hat{X}(t_0) = X(0),
\]

\[
\Delta t_n = t_{n+1} - t_n
\]

\[
\Delta W_n = W(t_{n+1}) - W(t_n).
\]

It can be shown [39] under regularity assumption 1.1.1 that the approximation behaves as

\[
\mathbb{E}[\hat{X}(t_N) - X(t_N)] = \mathcal{O}(\max_n |t_{n+1} - t_n|)
\]

\[
\mathbb{E}[|\hat{X}(t_N) - X(t_N)|^2] = \mathcal{O}(\max_n |t_{n+1} - t_n|^2).
\]

While other time-stepping schemes like the Milstein scheme (see [49]) exist, their scope of applicability is often limited, and the one has to resort to the rather low-order error of the forward-Euler integration given above.

This lower order of convergence is further deteriorated, when sampling for hitting times, or maxima of trajectories, which are quantities of interest in the third article of this thesis.

### 3.2.1 Variance reduction schemes

#### 3.2.1.1 Antithetic variables

While the convergence of the Monte Carlo estimators is set by CLT, as the number of samples, \(M\) increases, there are ways of reducing the constant coefficient that multiplies this asymptotic rate. One such way is the use of antithetic variables. When using antithetic variables, one generates independent realisations \(\theta, \tilde{\theta}\), where \(\tilde{\theta}\)
is the antithetic pair of \( \theta \) that has the same distribution as \( \tilde{\theta} \). One particular choice of the antithetic variable is to use the Brownian motion \( W \) to generate \( \theta \) and its counterpart \(-W\) to generate \( \tilde{\theta} \), exploiting the symmetry of the Wiener process with respect to reflections. Instead of forming the statistical estimator as

\[
A_{sl} = \sum_{m=1}^{M} \frac{\theta_m}{M},
\]

one averages over the antithetic realisations

\[
A_a = \sum_{m=1}^{M} \frac{\theta_m + \tilde{\theta}_m}{2M}.
\]

The variance of the first estimator, \( A_{sl} \), is given as \( \frac{\text{var}(\theta)}{M} \). The variance of the antithetic estimator \( A_a \) is \( \frac{1}{2M} \left( \text{var}(\theta) + \text{cov}(\theta, \tilde{\theta}) \right) \). I.e., we have that if \( \theta \) and \( \tilde{\theta} \) are perfectly negatively correlated, the estimator \( A_a \) has zero variance. For example, determining the intercept of an affine function of \( W(T) \), the antithetic variable gives perfect results: only one realisation is needed to nail down the parameter. On the other hand, a higher computational cost is incurred when evaluating both \( \theta \) and \( \tilde{\theta} \). The cost of the estimator \( A_a \) can be nearly twice as high as that of \( A_{sl} \), as the solution of \( \tilde{\theta} \) is equally costly as \( \theta \). The cost of random number generation is saved.

### 3.2.1.2 Control variates

Another popular technique is that of control variates. In this method, one uses an auxiliary random variable, the control variate, which is correlated with the quantity sampled and has a known mean.

At the heart of the method is the simple identity

\[
E[ A_{sl} ] = E \left[ \sum_{m=1}^{M} \frac{\theta_m}{M} \right] = E \left[ \sum_{m=1}^{M} \frac{\theta_m + \beta(\psi_m - E[\psi_m])}{M} \right] = E[ A_{cv} ],
\]
where $\beta \in \mathbb{R}$. Adding a zero-mean random variable to the sampler, one can significantly reduce the variance of the estimator.

Variance of the resulting estimator $A_{cv}$ is given as

$$\text{var} (\theta) + \beta^2 \text{var} (\psi) + 2 \beta \text{cov} (\theta, \psi)$$

$$\frac{1}{M}$$

differentiating the numerator with respect to $\beta$ and taking the root, we find a critical point at

$$\beta = -\rho \sqrt{\frac{\text{var} (\theta)}{\text{var} (\psi)}}$$

with $\rho$ denoting the correlation between $\theta$ and $\psi$ and the resulting variance of the estimator is

$$\text{var} (A_{cv}) = \text{var} (A_{sl}) (\rho - 1) (\rho + 1)$$

Once again, if we find a control variate $\psi$ that perfectly correlates with the random samples $\theta$, either negatively or positively, the estimator has zero variance. For all intermediate values for the correlation parameter $\rho$, the variance of the estimator is reduced too. However, we do note that, like in the case of antithetic variables, the cost of forming the estimator $A_{cv}$ is higher than the cost of the corresponding estimator $A_{sl}$, which requires no evaluation of the random realizations $\psi$.

A multitude of stochastic modeling problems relevant to financial applications are diffusive in nature, and it is often the case that the diffusion constant does not depend too wildly on the state of the system. As a result, it is often beneficial to use control variates formed by the exactly solvable models discussed above. For example, the log-normal random variables are well-approximated by normal ones, the sum of different components of a multivariate log-normal is not too badly approximated by
3.2.1.3 Multilevel Monte Carlo

A particular variant of the control variate scheme is that the multilevel Monte Carlo method by Giles [26]. In the multilevel Monte Carlo method, one generates a sequence of approximations to the true problem, labeled by a discrete parameter \( \ell \in \mathbb{N} \). Assuming that the method converges to the true solution as \( \ell \) tends to infinity, in the strong sense, one can reap significant computational gains by forming the Monte Carlo estimator as a telescoping sum

\[
A_{ML} = \sum_{m=1}^{M_0} \frac{g\left(X^{(0)}(\omega_m)\right)}{M_0} + \sum_{\ell=1}^{L} \sum_{m=1}^{M_{\ell}} \frac{g\left(X^{(\ell)}(\omega_m)\right) - g\left(X^{(\ell-1)}(\omega_m)\right)}{M_{\ell}},
\]

i.e. one samples the same realization on two different levels of approximation \( \ell \) and uses the difference of the payoff functional to form the estimator. The mean of the estimator matches the mean, of the finest level of approximation, \( \ell = L \). However, with appropriate choice of the sample sizes, \( M_\ell \), most of the samples are done at coarse levels of approximation, resulting in computational gains. This approach has been generalized to so-called multi-index Monte Carlo, in which the discretization is characterized by multiple discretization parameters \( \ell_1, \ell_2, \ldots, \ell_n \) [31]. The multilevel Monte Carlo approach also provides an essential building block for the Article I in Section [?], in which a particular adaptive approximation scheme is built for a low-regularity SDEs.

### 3.3 PDE methods

A particular class of methods that work particularly well when the underlying state space is of low dimension is that of PDE methods. These methods side-step the probabilistic nature of the problem and aim to solve for the relevant quantities of interest
by discretizing the relevant Black-Scholes or Hamilton-Jacobi-Bellman equations directly.

The literature on PDEs is vast (see, for example, [21, 23, 58]), and as a result there are many solution methods, such as the Finite Difference [1, Chapter 3] [18] and Finite Element method [1, Chapter 4], well suited for solving the PDEs arising from the field of finance.

In this thesis, the contributions of the second article fall within the realm of PDE methods, with the slight exception that for the jump-diffusion processes the relevant equations have an additional non-local term that corresponds to the jumps present in the dynamics.

Furthermore, in the third article, we use discretized PDE methods, and Finite Differences, in particular, to solve for the approximate early-exercise region, that is used to construct a lower bound for a basket option.

Perhaps the simplest method to understand is that of finite differences. The state space is divided into a mesh, and the spatial differential operators are replaced by their respective finite-difference approximations. The result is a high-dimensional ODE, on which one imposes an appropriate time stepping method.

The other well-known and widely used method is that of Finite differences. In this method, one focuses on the variational, or weak formulation of the respective Equations (1.3) and (1.6). Rather than solving the variational problem in its original function (Sobolev) space $V$, one restricts the solution to a subspace $V_h$ and finds a solution to the variational problem within that subspace. The resulting matrix equation is solved in time much like the Finite differences solution, using an appropriate time-stepper on a temporal mesh. The FEM approach allows the user to exploit a vast body of literature regarding error control, and to choose basis functions that are well-suited for the approximation of the value function. In particular, in multi-dimensional settings, one may choose a basis that is sparse. In practice, this can be
done in many ways, one of them through the definition of univariate bases \( V^\ell \) and instead of constructing the multivariate basis as

\[
V_d = V^\ell \otimes V^\ell \otimes \cdots \otimes V^\ell = \sum_{0 \leq k_1, k_2, \ldots, k_d \leq \ell} Y^{k_1} \otimes Y^{k_2} \otimes \cdots \otimes Y^{k_d}, \quad (3.3)
\]

one looks for a space with lower total dimension,

\[
\tilde{V}_d = \sum_{0 \leq k_1 + k_2 + \cdots + k_d \leq \ell} Y^{k_1} \otimes Y^{k_2} \otimes \cdots \otimes Y^{k_d}. \quad (3.4)
\]

Whereas the basis \( V_d \) of (3.3) includes basis functions such that the indices \( k_i \) are simultaneously high along all directions of, \( \tilde{V}_d \) allows basis functions such that the sum of indices is bounded from above. For example, the unidirectional bases \( Y^{k_i} \) include basis functions whose derivatives are bounded by \( k_i \). Thus, the reduced basis of (3.4) includes only basis functions that do not have a large derivative along multiple directions simultaneously. Assuming that the functions satisfy this assumption of vanishing high-order cross-derivatives one can achieve a good approximation of the true function being approximated, thus avoiding the curse of dimensionality and achieving computational gains when compared to methods that use the full basis \( V_d \).
Chapter 4

Contributions of papers

The theoretical contributions of this dissertation are given in three research manuscripts. In this section we outline the individual contributions and briefly provide some context of the field that these works contribute towards.

4.1 Article I

In the first work, we focus on Euler–Maruyama numerical solutions on SDEs. We derive a formal mean square error (MSE) expansion for the numerical bias introduced for the time stepping method. This mean square, or strong error, and its asymptotic convergence to zero is of key importance for multilevel Monte Carlo methods. In the work, we use the mean square error expansion as a basis for the adaptive time stepping Euler–Maruyama integrator and use it as a part of the adaptive multilevel Monte Carlo solver that is used for weak approximations of SDEs. The resulting method is well-suited for dealing with low-regularity problems, which we demonstrate through numerical examples.

In particular, we demonstrate how the developed MLMC method outperforms the corresponding uniform time stepping MLMC method by orders of magnitude. We demonstrate how the output of our developed method has an error which is bounded with high probability by $TOL > 0$ at the cost rate $O(TOL^{-2}\log(TOL)^4)$. We note that the achieved rate is near-optimal for the test problems in the sense that even if one could sample the values of forward-solutions exactly at cost $O(1)$, the cost of a
Monte Carlo solution would still be $O(TOL^{-2})$ due to the sampling of independent, identically distributed stochastic variables. In our method, the need for a time-stepper results only in logarithmic factors in the complexity of the method, as well as a larger constant factor in the complexity.

Aspects of the work have been presented in MCQMC2014 in Leuven, Belgium, research seminar of the Weierstrass Institute, Berlin, Germany in 2014, SIAM UQ 2016 in Lausanne, Switzerland, and finally, MCQMC2016 in Stanford University, USA.

4.2 Article II

In this work, we provide an analysis of the error committed when using a Fourier method to price European options when the underlying asset follows an exponential Lévy dynamic. The price of the option is described by a partial integro-differential equation (PIDE), which, using a Fourier transformation, can be cast into an ordinary differential equation (ODE) that can be solved analytically exploiting the known characteristic exponent of the Lévy process. The analytical and exact solution of the Fourier-transformed ODE needs to be inverted into real space for the solution to be of practical use. Our analysis provides a novel bound for the error committed in the numerical Fourier inversion.

Our bound has a particularly useful structure, for it can be decomposed into a product of two terms: one arising from the payoff of the option being priced and the other from the dynamics of the system. Furthermore, the bound works independently of the asymptotic behavior of the option prices at extreme asset prices. We present how the numerical parameters of the Fourier inversion can be selected in a way that minimizes the bound. This is demonstrated with a wide variety of examples, both jump-diffusion, and pure-jump. The resulting bounds are comparable or superior to previous bounds presented in the literature.
The results of the work were presented in Asian Quantitative Finance Conference 2016 in Osaka, Japan.

4.3 Article III

This work addresses the problem of pricing American basket options in a multivariate setting, which includes among others, the Bachelier and the Black-Scholes models. In high dimensions, nonlinear partial differential equation methods for solving the problem become prohibitively costly due to the curse of dimensionality. Instead, this work proposes to use a stopping rule that depends on the dynamics of a low-dimensional Markovian projection of the given basket of assets. It is shown that the ability to approximate the original value function by a lower-dimensional approximation is a feature of the dynamics of the system and is unaffected by the path-dependent nature of the American basket option. Assuming that we know the density of the forward process and using the Laplace approximation, we first efficiently evaluate the diffusion coefficient corresponding to the low-dimensional Markovian projection of the basket. Then, we approximate the optimal early-exercise boundary of the option by solving a Hamilton-Jacobi-Bellman partial differential equation in the projected, low-dimensional space. The resulting near-optimal early-exercise boundary is used to produce an exercise strategy for the high-dimensional option, thereby providing a lower bound for the price of the American basket option. A corresponding upper bound is also provided. These bounds allow assessing the accuracy of the proposed pricing method. Indeed, our approximate early-exercise strategy provides a straightforward lower bound for the American basket option price. Following a duality argument due to Rogers, we derive a corresponding upper bound solving only the low-dimensional optimal control problem. Numerically, we show the feasibility of the method using baskets with dimensions up to fifty. In these examples, the resulting option price relative errors are only of the order of few percent.
Chapter 5

Concluding remarks

In conclusion, Itô differential equations and jump diffusion equations offer a rich family of dynamic stochastic models that is particularly well-suited for the modeling of financial uncertainties.

Since many of these models do not have an explicit analytic solution, the need for efficient numerical approximations is profound. In the included research articles, we have focused on different classes of such numerical approximations. The fields of applications can be seen as different generalizations on the classical Black-Scholes model.

The scope of application of the first article of this thesis is that of low-regularity problems: models in which the drift in particular features diverges that deteriorate the convergence rates of the solutions. We have derived a formal mean-square error expansion for forward-Euler approximations of SDEs and implemented an adaptive time-stepper based on the mean-square error expansion. This time stepper focuses refinements of the temporal mesh in regions where the refinement reduces the mean-square error the most. The resulting multilevel Monte Carlo estimator is shown to outperform its uniform time-step counterpart by orders of magnitude in low-regularity examples while maintaining the comparable performance to the classical results in regular problems.

For the second article focuses on the problem raised very shortly after the seminal work by Black-Scholes: discontinuous dynamics. Focusing on a particular case of models featuring discontinuous dynamics, namely that of Lévy models, the problem
of European option pricing can be cast into an ODE problem using Fourier transform. We have analyzed the error committed in the inversion from Fourier space into the real space and given a bound for the error. We have also discussed the use of our proposed bound as a guide for the selection of the numerical parameters for the Fourier inversion, resulting in fast and robust computation with error satisfying a pre-described tolerance.

In the third article, the scope of application is generalized in two senses. Firstly, we look into optimal stopping problems arising from pricing American options. Secondly, we are interested in the case of basket options, where the underlying stopping rules are high-dimensional and the well-established difficulties of the numerical approximation of stopping times are aggravated by the curse of dimensionality. We have derived an upper and a lower bound for the price of American options using the technique of Markovian projection. In relevant numerical examples, the duality gap is of the order of few percent of the price of the option of interest, with the computational complexity of the method being polynomial in the dimension of the forward model.
REFERENCES


Part II

Included Articles
Chapter 6

Article I

Abstract

A formal mean square error expansion (MSE) is derived for Euler–Maruyama numerical solutions of stochastic differential equations (SDE). The error expansion is used to construct a pathwise, \textit{a posteriori}, adaptive time-stepping Euler–Maruyama algorithm for numerical solutions of SDE, and the resulting algorithm is incorporated into a multilevel Monte Carlo (MLMC) algorithm for weak approximations of SDE. This gives an efficient MSE adaptive MLMC algorithm for handling a number of low-regularity approximation problems. In low-regularity numerical example problems, the developed adaptive MLMC algorithm is shown to outperform the uniform time-stepping MLMC algorithm by orders of magnitude, producing output whose error with high probability is bounded by $\epsilon_T > 0$ at the near-optimal MLMC cost rate $\mathcal{O}(\epsilon_T^{-2} \log(\epsilon_T)^4)$.

6.1 Introduction

SDE models are frequently applied in mathematical finance $[12, 28, 29]$, where an observable may, for example, represent the payoff of an option. SDE are also used to model the dynamics of multiscale physical, chemical or biochemical systems $[11, 25, 30, 32]$, where, for instance, concentrations, temperature and energy may be sought observables.

Given a filtered, complete probability space $(\Omega, \mathcal{F}, (\mathcal{F}_t)_{0 \leq t \leq T}, P)$, we consider the
Itô SDE

\[ dX_t = a(t, X_t)dt + b(t, X_t)dW_t, \quad t \in (0, T], \]
\[ X_0 = x_0, \]

(6.1)

where \( X : [0, T] \times \Omega \to \mathbb{R}^{d_1} \) is a stochastic process with randomness generated by a \( K \)-dimensional Wiener process, \( W : [0, T] \times \Omega \to \mathbb{R}^{d_2} \), with independent components, \( W = (W^{(1)}, W^{(2)}, \ldots, W^{(d_2)}) \), and \( a : [0, T] \times \mathbb{R}^{d_1} \to \mathbb{R}^{d_1} \) and \( b : [0, T] \times \mathbb{R}^{d_1} \to \mathbb{R}^{d_1 \times d_2} \) are the drift and diffusion coefficients, respectively. The considered filtration \( \mathcal{F}_t \) is generated from the history of the Wiener process \( W \) up to time \( t \) and the possible outcomes of the initial data \( X_0 \), and succeedingly completed with all \( \mathbb{P} \)-outer measure zero sets of the sample space \( \Omega \). That is

\[ \mathcal{F}_t := \sigma(\{W_s\}_{0 \leq s \leq t}) \vee \sigma(X_0) \]

where the operation \( \mathcal{A} \vee \mathcal{B} \) denotes the \( \sigma \)-algebra generated by the pair of \( \sigma \)-algebras \( \mathcal{A} \) and \( \mathcal{B} \), i.e., \( \mathcal{A} \vee \mathcal{B} := \sigma(\mathcal{A}, \mathcal{B}) \), and \( \overline{\mathcal{A}} \) denotes the \( \mathbb{P} \)-outer measure null-set completion of \( \mathcal{A} \),

\[ \overline{\mathcal{A}} := \mathcal{A} \vee \left\{ A \subset \Omega \mid \inf_{\hat{A} \in \{\hat{A} \in \mathcal{A} \mid \hat{A} \supset A\}} \mathbb{P}(\hat{A}) = 0 \right\}. \]

The contributions of this work are twofold. First, an \textit{a posteriori} adaptive time-stepping algorithm for computing numerical realizations of SDE using the Euler–Maruyama method is developed. And second, for a given observable \( g : \mathbb{R}^{d_1} \to \mathbb{R} \), we construct a mean square error (MSE) adaptive time-stepping multilevel Monte Carlo (MLMC) algorithm for approximating the expected value, \( \mathbb{E}[g(X_T)] \), under the following constraint:

\[ \mathbb{P}(|\mathbb{E}[g(X_T)] - A| \leq \epsilon_T) \geq 1 - \delta. \]

(6.2)

Here, \( A \) denotes the algorithm’s approximation of \( \mathbb{E}[g(X_T)] \) (examples of which are
given in Equations (6.1) and (6.6) and $\epsilon_T$ and $\delta > 0$ are accuracy and confidence constraints, respectively.

The rest of this paper is organized as follows: First, in Section 6.1.1, we review the Monte Carlo methods and their use with the Euler–Maruyama integrator. This is followed by discussion of Multilevel Monte Carlo methods and adaptivity for SDEs. The theory, framework and numerical examples for the MSE adaptive algorithm is presented in Section 6.2. In Section 6.3, we develop the framework for the MSE adaptive MLMC algorithm and present implementational details in algorithms with pseudocode. In Section 6.4 we compare the performance of the MSE adaptive and uniform MLMC algorithms in a couple of numerical examples, one of which is a low-regularity SDE problem. Finally, we present brief conclusions followed by technical proofs and the extension of the main result to higher-dimensional problems in the appendices.

6.1.1 Monte Carlo Methods and the Euler–Maruyama Scheme.

Monte Carlo (MC) methods provide a robust and typically non-intrusive way to compute weak approximations of SDE. The convergence rate of MC methods does not depend on the dimension of the problem; for that reason, MC is particularly effective on multi-dimensional problems. In its simplest form, an approximation by the MC method consists of the following two steps:

(A.1) Compute $M$ independent and identically distributed numerical realizations, $\overline{X}_T(\omega_m)$, of the SDE (6.1).

(A.2) Approximate $E[g(X_T)]$ by the sample average

$$A := \sum_{m=1}^{M} \frac{g(\overline{X}_T(\omega_m))}{M}.$$
Regarding ordinary differential equations (ODE), the theory for numerical integrators of different orders for scalar SDE is vast. Provided sufficient regularity, higher order integrators generally yield higher convergence rates \[22\]. With MC methods it is straightforward to determine that the goal \[6.2\] is fulfilled at the computational cost \(O\left(\epsilon^{-2-1/\alpha}T\right)\), where \(\alpha \geq 0\) denotes the weak convergence rate of the numerical method, as defined in Equation \[6.4\].

As a method of temporal discretization, the Euler–Maruyama scheme is given by

\[
X_{t_{n+1}} = X_{t_n} + a(t_n, X_{t_n})\Delta t_n + b(t_n, X_{t_n})\Delta W_n, \tag{6.3}
\]

using time steps \(\Delta t_n = t_{n+1} - t_n\) and Wiener increments \(\Delta W_n = W_{t_{n+1}} - W_{t_n} \sim \mathcal{N}(0, \Delta t_n I_d)\), where \(I_d\) denotes the \(d \times d\) identity matrix. In this work, we will focus exclusively on Euler–Maruyama time-stepping. The Euler–Maruyama scheme, which may be considered the SDE-equivalent of the forward-Euler method for ODE, has, under sufficient regularity, first-order weak convergence rate

\[
|\mathbb{E}[g(X_T) - g(X_T)]| = O\left(\max_n \Delta t_n\right), \tag{6.4}
\]

and also first-order MSE convergence rate

\[
\mathbb{E}\left[(g(X_T) - g(X_T))^2\right] = O\left(\max_n \Delta t_n\right), \tag{6.5}
\]

cf. \[22\]. For multi-dimensional SDE problems, higher order schemes are generally less applicable, as either the diffusion coefficient matrix has to fulfill a rigid commutativity condition, or Levy areas, required in higher order numerical schemes, have to be accurately approximated to achieve better convergence rates than those obtained with the Euler–Maruyama method \[22\].
6.1.2 Uniform and Adaptive Time-Stepping MLMC

MLMC is a class of MC methods that uses a hierarchy of subtly correlated and increasingly refined realization ensembles to reduce the variance of the sample estimator. In comparison with single-level MC, MLMC may yield orders of magnitude reductions in the computational cost of moment approximations. MLMC was first introduced by Heinrich \cite{14, 15} for approximating integrals that depend on random parameters. For applications in SDE problems, Kebaier \cite{21} introduced a two-level MC method and demonstrated its potential efficiency gains over single-level MC. Giles \cite{8} thereafter developed an MLMC algorithm for SDE, exhibiting even higher potential efficiency gains. Presently, MLMC is a vibrant and growing research topic, (cf. \cite{3, 4, 9, 10, 13, 26, 34}, and references therein).

6.1.2.1 MLMC Notation

We define the multilevel estimator by

\[
A_{ML} := \sum_{\ell=0}^{L} \sum_{i=1}^{M_{\ell}} \frac{\Delta_{\ell} g(\omega_{i,\ell})}{M_{\ell}},
\]

(6.6)

where

\[
\Delta_{\ell} g(\omega) := \begin{cases} 
  g\left(\bar{X}_{T}^{(0)}(\omega)\right), & \text{if } \ell = 0, \\
  g\left(\bar{X}_{T}^{(\ell)}(\omega)\right) - g\left(\bar{X}_{T}^{(\ell-1)}(\omega)\right), & \text{otherwise}.
\end{cases}
\]

Here, the positive integer, \(L\), denotes the final level of the estimator, \(M_{\ell}\) is the number of sample realizations on the \(\ell\)-th level, and the realization pair, \(\bar{X}_{T}^{(\ell)}(\omega_{i,\ell})\) and \(\bar{X}_{T}^{(\ell-1)}(\omega_{i,\ell})\), are generated by the Euler–Maruyama method (6.3) using the same Wiener path, \(W(\omega_{i,\ell})\), sampled on the respective meshes, \(\Delta t^{(\ell)}\) and \(\Delta t^{(\ell-1)}\), (cf. Figure 6.1). For consistency, we also introduce the notation \(W^{(\ell)}(\omega)\) for the Wiener path restricted to the mesh \(\Delta t^{(\ell)}(\omega)\).
Figure 6.1: (Left) A sample Wiener path, $W$, generated on the coarse mesh, $\Delta t^{(0)}$, with uniform step size $1/10$ (blue line). The path is thereafter Brownian bridge interpolated onto a finer mesh, $\Delta t^{(1)}$, which has uniform step size of $1/20$ (green line). (Right) Euler–Maruyama numerical solutions of the Ornstein-Uhlenbeck SDE problem, $dX_t = 2(1 - X_t)dt + 0.2dW_t$, with initial condition $X_0 = 3/2$, are computed on the meshes $\Delta t^{(0)}$ (blue line) and $\Delta t^{(1)}$ (green line) using Wiener increments from the respective path resolutions.

6.1.2.2 Uniform Time-Stepping MLMC

In the uniform time-stepping MLMC introduced in [8], the respective SDE realizations $\{X_T^{(\ell)}\}_\ell$ are constructed on a hierarchy of uniform meshes with geometrically decaying step size, $\min \Delta t^{(\ell)} = \max \Delta t^{(\ell)} = T/N_\ell$, and $N_\ell = c^\ell N_0$ with $c \in \mathbb{N} \setminus \{1\}$ and $N_0$ a finite integer. For simplicity, we consider the uniform time-stepping MLMC method with $c = 2$.

6.1.2.3 Uniform Time-Stepping MLMC Error and Computational Complexity

By construction, the multilevel estimator is telescoping in expectation, i.e., $E[A_{MC}] = E\left[g(X_T^{(L)})\right]$. Using this property, we may conveniently bound the multilevel approximation error:

$$|E[g(X_T)] - A_{MC}| \leq \left|E\left[g(X_T) - g(X_T^{(L)})\right]\right| + \left|E\left[g(X_T^{(L)}) - A_{MC}\right]\right| =: \mathcal{E}_T + \mathcal{E}_S.$$
The approximation goal (6.2) is then reached by ensuring that the sum of the bias, $\mathcal{E}_T$, and the statistical error, $\mathcal{E}_S$, is bounded from above by $\epsilon_T$, e.g., by the constraints $\mathcal{E}_T \leq \epsilon_T/2$ and $\mathcal{E}_S \leq \epsilon_T/2$, (see Section 6.3.2 for more details on the MLMC error control). For the MSE error goal,

$$\mathbb{E}[(\mathbb{E}[g(X_T)] - A_{\mathcal{M}{\mathcal{C}}})^2] \leq \epsilon_T^2,$$

the following theorem states the optimal computational cost for MLMC:

**Theorem 6.1.1** (Computational cost of deterministic MLMC; Cliffe et al. [4]). Suppose there are constants $\alpha, \beta, \gamma$ such that $\alpha \geq \min(\beta, \gamma) \frac{2}{\alpha}$ and

1. $|\mathbb{E}[g(X_T^{(\ell)}) - g(X_T)]| = \mathcal{O}(N_\ell^{-\alpha})$,
2. $\text{Var}(\Delta_\ell g) = \mathcal{O}(N_\ell^{-\beta})$,
3. $\text{Cost}(\Delta_\ell g) = \mathcal{O}(N_\ell^\gamma)$.

Then, for any $\epsilon_T < e^{-1}$, there exists an $L$ and a sequence $\{M_\ell\}_{\ell=0}^L$ such that

$$\mathbb{E}[(A_{\mathcal{M}{\mathcal{C}}} - \mathbb{E}[g(X_T)])^2] \leq \epsilon_T^2, \quad (6.7)$$

and

$$\text{Cost}(A_{\mathcal{M}{\mathcal{C}}}) = \begin{cases} 
\mathcal{O}(\epsilon_T^{-2}), & \text{if } \beta > \gamma, \\
\mathcal{O}(\epsilon_T^{-2} \log(\epsilon_T)^2), & \text{if } \beta = \gamma, \\
\mathcal{O}(\epsilon_T^{-2+\frac{\beta-\gamma}{\alpha}}), & \text{if } \beta < \gamma.
\end{cases} \quad (6.8)$$

In comparison, the computational cost of achieving the goal (6.7) with single-level MC is $\mathcal{O}(\epsilon_T^{-2-\gamma/\alpha})$. Theorem 6.1.1 thus shows that for any problem with $\beta > 0$, MLMC will asymptotically be more efficient than single-level MC. Furthermore, the performance gain of MLMC over MC is particularly apparent in settings where $\beta \geq \gamma$. The latter property is linked to the contributions of this work. In low-regularity SDE
problems, e.g., Example 6.4.3 below and \[1, 35\], the uniform time-stepping Euler–Maruyama results in convergence rates for which $\beta < \gamma$. More sophisticated integrators can preserve rates such that $\beta \geq \gamma$.

**Remark 6.1.2.** Similar accuracy vs. complexity results to Theorem 6.1.1, requiring slightly stronger moment bounds, have also been derived for the approximation goal \((6.2)\) in the asymptotic setting when $\epsilon_T \downarrow 0$, cf. \[5, 16\].

### 6.1.2.4 MSE A Posteriori Adaptive Time-Stepping

In general, adaptive time-stepping algorithms seek to fulfill one of two equivalent goals \[2\]:

(B.1) Provided a computational budget $N$ and a norm $\| \cdot \|$, determine the possibly non-uniform mesh, which minimizes the error $\| g(X_T) - g(\overline{X}_T) \|$.

(B.2) Provided an error constraint $\| g(X_T) - g(\overline{X}_T) \| \leq \epsilon_T$, determine the possibly non-uniform mesh, which achieves the constraint at the minimum computational cost.

Evidently, the refinement criterion of an adaptive algorithm depends on the error one seeks to minimize. In this work, we consider adaptivity goal (B.1) with the error measured in terms of the MSE. This error measure is suitable for MLMC algorithms as it often will lead to improved convergence rates, $\beta$ (since $\text{Var}(\Delta \ell g) \leq E[\Delta \ell g^2]$), which by Theorem 6.1.1 may reduce the computational cost of MLMC. In Theorem 6.2.1, we derive the following error expansion for the MSE of Euler–Maruyama numerical solutions of the SDE \((6.1)\):

$$
E \left[ (g(X_T) - g(\overline{X}_T))^2 \right] = E \left[ \sum_{n=0}^{N-1} \overline{p}_n \Delta t_n^2 + o(\Delta t_n^2) \right],
$$

\((6.9)\)

where the error density, $\overline{p}_n$, is a function of the local error and sensitivities from the dual solution of the SDE problem, as defined in \[6.26\]. The error expansion \((6.9)\) is
an a posteriori error estimate for the MSE, and in our adaptive algorithm, the mesh is refined by equilibration of the expansion’s error indicators

\[ r_n := \rho_n \Delta t_n^2, \quad \text{for} \quad n = 0, 1, \ldots, N - 1. \]  

6.1.2.5 An MSE Adaptive MLMC Algorithm

Using the described MSE adaptive algorithm, we construct an MSE adaptive MLMC algorithm in Section 6.3. The MLMC algorithm generates SDE realizations, \( \{X_T^{(\ell)}\}_\ell \), on a hierarchy of pathwise adaptively refined meshes, \( \{\Delta t^{(\ell)}\}_\ell \). The meshes are nested, i.e., for all realizations \( \omega \in \Omega \),

\[ \Delta t^{(0)}(\omega) \subset \Delta t^{(1)}(\omega) \subset \ldots \Delta t^{(\ell)}(\omega) \subset \ldots, \]

with the constraint that the number of time steps in \( \Delta t^{(\ell)} \), \(|\Delta t^{(\ell)}|\), is bounded by \( 2N_\ell \):

\[ |\Delta t^{(\ell)}| < 2N_\ell = 2^{\ell+2}N_{-1}. \]

Here, \( N_{-1} \) denotes the pre-initial number of time steps; it is a bounded integer set in advance of the computations. This corresponds to the hierarchy setup for the uniform time-stepping MLMC algorithm in Section 6.1.2.2.

The potential efficiency gain of adaptive MLMC is experimentally illustrated in this work using the drift blow-up problem

\[ dX_t = \frac{rX_t}{|t - \xi|^p} \, dt + \sigma X_t \, dW_t, \quad X_0 = 1. \]

This problem is addressed in Example 6.4.3 for the three different singularity exponents \( p = 1/2, 2/3 \) and \( 3/4 \), with a pathwise, random singularity point \( \xi \sim U(1/4, 3/4) \), an observable \( g(x) = x \), and a final time \( T = 1 \). For the given singularity
exponents, we observe experimental deteriorating convergence rates, $\alpha = (1 - p)$ and $\beta = 2(1 - p)$, for the uniform time-stepping Euler–Maruyama integrator, while for the adaptive time-step Euler–Maruyama we observe $\alpha \approx 1$ and $\beta \approx 1$. Then, as predicted by Theorem 6.1.1 we also observe an order of magnitude difference in computational cost between the two algorithms (cf. Table 6.1).

Table 6.1: Observed computational cost – disregarding $\log(\epsilon T)$ multiplicative factors of finite order – for the drift blow-up study in Example 6.4.3.

<table>
<thead>
<tr>
<th>Singularity exponent $p$</th>
<th>Adaptive MLMC</th>
<th>Uniform MLMC</th>
</tr>
</thead>
<tbody>
<tr>
<td>1/2</td>
<td>$\epsilon_T^{-2}$</td>
<td>$\epsilon_T^{-2}$</td>
</tr>
<tr>
<td>2/3</td>
<td>$\epsilon_T^{-2}$</td>
<td>$\epsilon_T^{-3}$</td>
</tr>
<tr>
<td>3/4</td>
<td>$\epsilon_T^{-2}$</td>
<td>$\epsilon_T^{-4}$</td>
</tr>
</tbody>
</table>

6.1.2.6 Earlier Works on Adaptivity for SDE

Gaines’ and Lyons’ work [7] is one of the seminal contributions on adaptive algorithms for SDE. They present an algorithm that seeks to minimize the pathwise error of the mean and variation of the local error conditioned on the $\sigma$-algebra generated by (i.e., the values at which the Wiener path has been evaluated in order to numerically integrate the SDE realization) $\{W_{t_n}\}_{n=1}^{N}$. The method may be used in combination with different numerical integration methods, and an approach to approximations of potentially needed Levy areas is proposed, facilitated by a binary tree representation of the Wiener path realization at its evaluation points. As for a posteriori adaptive algorithms, the error indicators in Gaines’ and Lyons’ algorithm are given by products of local errors and weight terms, but, unlike in a posteriori methods, the weight terms are computed from a priori estimates, making their approach a hybrid one.

Szepessy et al. [31] introduced a posteriori weak error based adaptivity for the Euler–Maruyama algorithm with numerically computable error indicator terms. Their
development of weak error adaptivity took inspiration from Talay and Tubaro’s seminal work [33], where an error expansion for the weak error was derived for the Euler–Maruyama algorithm when uniform time steps were used. In [16], Szepessy et al.’s weak error adaptive algorithm was used in the construction of a weak error adaptive MLMC algorithm. To the best of our knowledge, the present work is the first on MSE a posteriori adaptive algorithms for SDE both in the MC- and MLMC setting.

Among other adaptive algorithms for SDE, many have refinement criterions based only or primarily on estimates of the local error. For example in [17], where the step-size depends on the size of the diffusion coefficient for a MSE Euler–Maruyama adaptive algorithm; in [23], the step-size is controlled by the variation in the size of the drift coefficient in the constructed Euler–Maruyama adaptive algorithm, which preserves the long-term ergodic behavior of the true solution for many SDE problems; and in [19], a local error based adaptive Milstein algorithm is developed for solving multi-dimensional chemical Langevin equations.

6.2 Derivation of the MSE A Posteriori Adaptive Algorithm

In this section, we construct an MSE a posteriori adaptive algorithm for SDE whose realizations are numerically integrated by the Euler–Maruyama algorithm (6.3). Our goal is, in rough terms, to obtain an algorithm for solving the SDE problem (6.1) that for a fixed number of intervals $N$, determines the time-stepping, $\Delta t_0, \Delta t_1, \ldots, \Delta t_{N-1}$ such that the MSE, $E\left[\left( g(X_T) - g(X_T) \right)^2 \right]$ is minimized. That is,

$$E\left[\left( g(X_T) - g(X_T) \right)^2 \right] \rightarrow \min!, \quad N \text{ given} \quad (6.11)$$

The derivation of our adaptive algorithm consists of two steps. First, an error expansion for the MSE is presented in Theorem 6.2.1. Based on the error expansion, we thereafter construct a mesh refinement algorithm. At the end of the section, we
apply the adaptive algorithm to a few example problems.

6.2.1 The Error Expansion

Let us now present a leading-order error expansion for the MSE \((6.11)\) of the SDE problem \((6.1)\) in the one-dimensional (1D) setting, i.e., when \(X_t \in \mathbb{R}\) and the drift and diffusion coefficients are respectively of the form \(a : [0, T] \times \mathbb{R} \to \mathbb{R}\) and \(b : [0, T] \times \mathbb{R} \to \mathbb{R}\). An extension of the MSE error expansion to multi-dimensions is given in Appendix \(6.A.3\). To state the error expansion Theorem, some notation is needed. Let \(X^x_{s,t}\) denote the solution of the SDE \((6.1)\) at time \(s \geq t\), when the initial condition is \(X_t = x\) at time \(t\), i.e.,

\[
X^x_{s,t} := x + \int_t^s a(u, X_u)du + \int_t^s b(u, X_u)dW_u, \quad s \in [t, T],
\]

and in light of this notation, \(X_t\) is shorthand for \(X^x_{0,0}\). For a given observable \(g\), the payoff-of-flow map function is defined by \(\varphi(t, x) = g(X^x_{t,t})\). We also make use of the following function space notation

\[
C(U) := \{f : U \to \mathbb{R} | f \text{ is continuous}\},
\]

\[
C_b(U) := \{f : U \to \mathbb{R} | f \text{ is continuous and bounded}\},
\]

\[
C^k_b(\mathbb{R}) := \left\{f : \mathbb{R} \to \mathbb{R} | f \in C(\mathbb{R}) \text{ and } \frac{d^j}{dx^j} f \in C_b(\mathbb{R}) \text{ for all integers } 1 \leq j \leq k \right\},
\]

\[
C^{k_1, k_2}_b([0, T] \times \mathbb{R}) := \left\{f : [0, T] \times \mathbb{R} \to \mathbb{R} | f \in C([0, T] \times \mathbb{R}) \text{ and } \partial^{j_1}_{t} \partial^{j_2}_{x} f \in C_b([0, T] \times \mathbb{R}) \text{ for all integers s.t. } j_1 \leq k_1 \text{ and } 1 \leq j_1 + j_2 \leq k_2 \right\}.
\]

We are now ready to present our mean square expansion result, namely,

**Theorem 6.2.1** (1D MSE leading-order error expansion). Assume that drift and diffusion coefficients and input data of the SDE \((6.1)\) fulfill

\((R.1)\) \(a, b \in C^{2,4}_b([0, T] \times \mathbb{R}),\)
(R.2) there exists a constant \( C > 0 \) such that
\[
|a(t, x)|^2 + |b(t, x)|^2 \leq C(1 + |x|^2), \quad \forall x \in \mathbb{R} \text{ and } \forall t \in [0, T],
\]

(R.3) \( g' \in C^3_b(\mathbb{R}) \) and there exists a \( k \in \mathbb{N} \) such
\[
|g(x)| + |g'(x)| \leq C(1 + |x|^k), \quad \forall x \in \mathbb{R}, \quad (6.13)
\]

(R.4) for the initial data, \( X_0 \in \mathcal{F}_0 \) and \( \mathbb{E}[|X_0|^p] < \infty \) for all \( p \geq 1 \).

Assume further the mesh points \( 0 = t_0 < t_1 < \ldots < t_N = T \)

(M.1) are stopping times for which \( t_n \in \mathcal{F}_{t_{n-1}} \) for \( n = 1, 2, \ldots, N \),

(M.2) for all mesh realizations, there exists a deterministic integer, \( \bar{N} \), and a \( c_1 > 0 \) such that \( c_1 \bar{N} \leq N \leq \bar{N} \) and a \( c_2 > 0 \) such that \( \max_{n \in \{0, 1, \ldots, N-1\}} \Delta t_n < c_2 \bar{N}^{-1} \),

(M.3) and there exists a \( c_3 > 0 \) such that for all \( p \in [1, 8] \) and \( n \in \{0, 1, \ldots, \bar{N} - 1\} \)
\[
\mathbb{E}[\Delta t_n^{2p}] \leq c_3 (\mathbb{E}[\Delta t_n^2])^p.
\]

Then, as \( \bar{N} \) increases,
\[
\mathbb{E}\left[\left(g(X_T) - g(\overline{X}_T)\right)^2\right] = \sum_{n=0}^{\bar{N}-1} \mathbb{E}\left[\varphi_x(t_n, \overline{X}_{t_n}) \left(\frac{b_x b}{2}(t_n, \overline{X}_{t_n}) \Delta t_n^2 + o(\Delta t_n^2)\right)\right], \quad (6.14)
\]

where we have defined \( t_n = T \) and \( \Delta t_n = 0 \) for all \( n \in \{N, N+1, \ldots, \bar{N}\} \). And replacing the first variation, \( \varphi_x(t_n, \overline{X}_n) \), by the numerical approximation, \( \varphi_{x,n} \), as defined in (6.25), yields the following to leading order all-terms-computable error
expansion:

\[
E \left[ (g(X_T) - g(\overline{X}_T))^2 \right] = \sum_{n=0}^{\hat{N}-1} E \left[ \frac{\varphi^2_{x,n}(b_x b)^2}{2} (t_n, \overline{X}_{t_n}) \Delta t_n^2 + o(\Delta t_n^2) \right].
\] (6.15)

**Remark 6.2.2.** In condition (M.2) of the above theorem we have introduced \( \hat{N} \) to denote the deterministic upper bound for the number of time steps in all mesh realizations. Moreover, from this point on the mesh points \( \{t_n\}_n \) and time steps \( \{\Delta t_n\}_n \) are defined for all indices \( \{0, 1, \ldots, \hat{N}\} \) with the natural extension \( t_n = T \) and \( \Delta t_n = 0 \) for all \( n \in \{N+1, \ldots, \hat{N}\} \). In addition to ensuring an upper bound on the complexity of a numerical realization and that \( \max_n \Delta t_n \to 0 \) as \( \hat{N} \to \infty \), replacing the random \( N \) (the smallest integer value for which \( t_N = T \) in a given mesh) with the deterministic \( \hat{N} \) in the MSE error expansion (6.15) simplifies our proof of Theorem 6.2.1.

**Remark 6.2.3.** For most SDE problems on which it is relevant to apply a posteriori adaptive integrators, at least one of the regularity conditions (R.1), (R.2), and (R.3) and the mesh adaptedness assumption (M.1) in Theorem 6.2.1 will not be fulfilled. In our adaptive algorithm, the error expansion (6.15) is interpreted in a formal sense and only used to facilitate the systematic construction of a mesh refinement criterion. When applied to low-regularity SDE problems where some of the conditions (R.1), (R.2), or (R.3), do not hold, the actual leading-order term of the error expansion (6.15) may contain other or additional terms besides \( \varphi^2_{x,n} \frac{(b_x b)^2}{2} (t_n, \overline{X}_{t_n}) \) in the error density. Example 6.4.3 presents a problem where ad hoc additional terms are added to the error density.

### 6.2.1.1 Numerical Approximation of the First Variation

The first variation of the flow map, \( \varphi(t,x) \), is defined by

\[
\varphi_x(t,x) = \partial_x g(X_t^{x,t}) = g'(X_t^{x,t}) \partial_x X_T^{x,t}
\]
and the first variation of the path itself, $\partial_x X^{x,t}_s$, is the solution of the linear SDE

$$
\begin{align*}
&d(\partial_x X^{x,t}_s) = a_x(s, X^{x,t}_s)\partial_x X^{x,t}_s ds + b_x(s, X^{x,t}_s)\partial_x X^{x,t}_s dW_s, \quad s \in (t, T], \\
&\partial_x X^{x,t}_t = 1.
\end{align*}
$$

(6.16)

To describe conditions under which the terms $g'(X^{x,t}_s)$ and $\partial_x X^{x,t}_s$ are well defined, let us first recall that if $X^{x,t}_s$ solves the SDE (6.12) and

$$
\mathbb{E} \left[ \int_t^T |X^{x,t}_s|^2 ds \right] < \infty,
$$

then we say that there exists a solution to the SDE; and if $\tilde{X}^{x,t}_s$ is another solution of the SDE with the same initial condition, then we say the solution is pathwise unique provided that

$$
P \left( \sup_{s \in [t, T]} \left| X^{x,t}_s - \tilde{X}^{x,t}_s \right| > 0 \right) = 0.
$$

Lemma 6.2.4. Assume the regularity assumptions (R.1), (R.2), (R.3), and (R.4) in Theorem 6.2.1 hold, and that for any fixed $t \in [0, T]$, $x \in \mathcal{F}_t$ and $\mathbb{E}[|x|^{2p}] < \infty$, for all $p \in \mathbb{N}$. Then there exist pathwise unique solutions $X^{x,t}_s$ and $\partial_x X^{x,t}_s$ to the respective SDE (6.12) and (6.16) for which

$$
\max \left\{ \mathbb{E} \left[ \sup_{s \in [t, T]} |X^{x,t}_s|^{2p} \right], \mathbb{E} \left[ \sup_{s \in [t, T]} |\partial_x X^{x,t}_s|^{2p} \right] \right\} < \infty, \quad \forall p \in \mathbb{N}.
$$

Furthermore, $\varphi_x(t, x) \in \mathcal{F}_T$ and

$$
\mathbb{E} \left[ |\varphi_x(t, x)|^{2p} \right] < \infty, \quad \forall p \in \mathbb{N}.
$$

Proof. By writing $(Y^{(1)}_s, Y^{(2)}_s) := (X^{x,t}_s, \partial_x X^{x,t}_s)$, (6.12) and (6.16) together form a
system of SDE:

\[
\begin{align*}
    dY_s^{(1)} &= a(s, Y_s^{(1)})ds + b(s, Y_s^{(1)})dW_s \\
    dY_s^{(2)} &= a_x(s, Y_s^{(1)})Y_s^{(2)}ds + b_x(s, Y_s^{(1)})Y_s^{(2)}dW_s
\end{align*}
\]  

(6.17)

for \( s \in (t, T] \) and with initial condition \( Y_t = (x, 1) \). By the Lipschitz continuity and the linear growth bound of this system’s drift and diffusion coefficients, there exists a pathwise unique solution of the SDE (6.17) for which

\[
E \left[ \sup_{s \in [t,T]} |Y_s|^{2p} \right] < \infty, \quad \forall p \in \mathbb{N},
\]

(cf. [22, Theorems 4.5.3 and 4.5.4 and Exercise 4.5.5]). As solutions of the Itô SDE, \( X_{T}^{x,t}, \partial_x X_{T}^{x,t} \in \mathcal{F}_{T} \), and since we assume that \( g' \in C^3_b(\mathbb{R}) \), we know that \( g' \) is Borel measurable and so is the mapping \( f : \mathbb{R}^2 \to \mathbb{R} \) defined by \( f(x, y) = xy \). From this we conclude that \( \varphi_x(x, t) = f(g'(X_T^{x,t}), \partial_x X_T^{x,t}) \in \mathcal{F}_T \) and, by (6.A.2), Hölder’s and Minkowski’s inequalities, that for any \( p \in \mathbb{N} \),

\[
E \left[ |\varphi_x(t, x)|^{2p} \right] \leq C \sqrt{E \left[ \left( 1 + |X_T^{x,t}|^k \right)^{4p} \right] E \left[ |\partial_x X_T^{x,t}|^{4p} \right]} < \infty.
\]

\[\square\]

To obtain an all-terms-computable error expansion in Theorem 6.2.1 which will be needed to construct an \textit{a posteriori} adaptive algorithm, the first variation of the flow map, \( \varphi_x \), is approximated by the first variation of the Euler–Maruyama numerical solution,

\[
\varphi_{x,n} := g'(X_T)\partial_{X_{t_n}}X_T.
\]
Here, for \( k > n \), \( \partial_{X_{t_n}} X_{t_k} \) is the solution of the Euler–Maruyama scheme

\[
(\partial_{X_{t_n}} X)_{t_{j+1}} = (\partial_{X_{t_n}} X)_{t_j} + a_x(t_j, X_{t_j}) (\partial_{X_{t_n}} X)_{t_j} \Delta t_j + b_x(t_j, X_{t_j}) (\partial_{X_{t_n}} X)_{t_j} \Delta W_j,
\]

for \( j = n, n+1, \ldots k-1 \) and with the initial condition \( \partial_{X_{t_n}} X_{t_n} = 1 \), which is coupled to the numerical solution of the SDE, \( X_{t_j} \).

**Lemma 6.2.5.** If the assumptions (R.1), (R.2), (R.3), (R.4), (M.1) and (M.2) in Theorem 6.2.1 hold, then the numerical solution \( X \) of (6.3) converges in mean square sense to the solution of the SDE (6.1),

\[
\max_{1 \leq n \leq N} \left( E \left[ |X_{t_n} - X_{t_n}|^{2p} \right] \right)^{1/2p} \leq C \tilde{N}^{-1/2},
\]

and

\[
\max_{1 \leq n \leq N} E \left[ |X_{t_n}|^{2p} \right] < \infty, \quad \forall p \in \mathbb{N}. \tag{6.20}
\]

For any fixed \( 1 \leq n \leq \tilde{N} \), the numerical solution \( \partial_{X_{t_n}} X \) of (6.18) converges in mean square sense to \( \partial_x X_{X_{t_n}, t_n} \),

\[
\max_{n \leq k \leq N} \left( E \left[ \left| \partial_{X_{t_n}} X_{t_k} - \partial_x X_{X_{t_n}, t_n} \right|^{2p} \right] \right)^{1/2p} \leq C \tilde{N}^{-1/2}, \tag{6.21}
\]

and

\[
\max_{n \leq k \leq N} E \left[ \left| \partial_{X_{t_n}} X_{t_k} \right|^{2p} \right] < \infty, \quad \forall p \in \mathbb{N}. \tag{6.22}
\]

Furthermore, \( \varphi_{x,n} \in F_T \) and

\[
E \left[ \left| \varphi_{x,n} \right|^{2p} \right] < \infty, \quad \forall p \in \mathbb{N}. \tag{6.23}
\]

**Proof.** The system \( \Upsilon_{t_k} := (X_{t_k}, \partial_{X_{t_n}} X_{t_k}) \) provides solutions approximating the SDE (6.17).
that are generated by the Euler–Maruyama scheme

\[
Y_{t_k+1}^{(1)} = Y_{t_k}^{(1)} + a(t_k, Y_{t_k}^{(1)}) \Delta t_k + b(t_k, Y_{t_k}^{(1)}) \Delta W_k, \\
Y_{t_k+1}^{(2)} = Y_{t_k}^{(2)} + a_x(t_k, Y_{t_k}^{(1)}) Y_{t_k}^{(2)} \Delta t_k + b_x(t_k, Y_{t_k}^{(1)}) Y_{t_k}^{(2)} \Delta W_k,
\]

(6.24)

for \( k \geq n \) and with initial condition \( \bar{Y}_{t_n} = (\bar{X}_{t_n}, 1) \). The assumptions of [22] Theorem 10.6.3 and the remark following it] are fulfilled for the SDE (6.17), which implies the strong convergence of \( \bar{Y} \) to \( Y \) and that the inequalities (6.19), (6.20), (6.21), and (6.22) hold. That \( \bar{\varphi}_{x,n} \in \mathcal{F}_T \) and that inequality (6.23) holds can be shown by a similar argument as in the proof of Lemma 6.2.4.

From the SDE (6.24), it is clear that

\[
\partial_{X_n} \bar{X}_T = \prod_{k=n}^{N-1} \left( 1 + a_x(t_k, \bar{X}_{t_k}) \Delta t_k + b_x(t_k, \bar{X}_{t_k}) \Delta W_k \right),
\]

and this implies that \( \bar{\varphi}_{x,n} \) solves the backward scheme

\[
\bar{\varphi}_{x,n} = c_x(t_n, \bar{X}_{t_n}) \bar{\varphi}_{x,n+1}, \quad n = N - 1, N - 2, \ldots, 0,
\]

(6.25)

with the initial condition \( \bar{\varphi}_{x,N} = g'(\bar{X}_T) \) and the shorthand notation

\[
c(t_n, \bar{X}_{t_n}) := \bar{X}_{t_n} + a(t_n, \bar{X}_{t_n}) \Delta t_n + b(t_n, \bar{X}_{t_n}) \Delta W_n.
\]

The backward scheme (6.25) is convenient from a computational perspective since it implies that the set of points, \( \{\bar{\varphi}_{x,n}\}_{n=0}^{N} \), can be computed at the same cost as that
of one-path realization, \( \{X_{t_n}\}_{n=0}^N \), which can be verified as follows

\[
\varphi_{x,n} = g'(X_T) \prod_{k=n}^{N-1} c_x(t_k, X_{t_k})
\]

\[
= c_x(t_n, X_{t_n}) g'(X_T) \prod_{k=n+1}^{N-1} c_x(t_k, X_{t_k})
\]

\[
= c_x(t_n, X_{t_n}) g'(X_T) \partial_{t_{n+1}} X_T
\]

\[
= c_x(t_n, X_{t_n}) \varphi_{x,n+1}.
\]

### 6.2.2 The Adaptive Algorithm

Having derived computable expressions for all terms in the error expansion, we next introduce the error density

\[
\rho_n := \varphi_{x,n}^{2} \frac{(b_x b)^2}{2} (t_n, X_{t_n}), \quad n = 0, 1, \ldots, N - 1,
\]

and, for representing the numerical solution’s error contribution from the time interval \((t_n, t_{n+1})\), the error indicators

\[
\tau_n := \rho_n \Delta t_n^2, \quad n = 0, 1, \ldots, N - 1.
\]

The error expansion (6.15) may then be written as

\[
E \left[ (g(X_T) - g(X_T))^2 \right] = \sum_{n=0}^{N-1} E[\tau_n + o(\Delta t_n^2)].
\]

The final goal of the adaptive algorithm is minimization of the leading order of the MSE in (6.28), namely, \( E\left[\sum_{n=0}^{N-1} \tau_n\right] \), which (for each realization) is approached by minimization of the error expansion realization \( \sum_{n=0}^{N-1} \tau_n \). An approximately optimal
choice for the refinement procedure can be derived by introducing the Lagrangian

\[ \mathcal{L}(\Delta t, \lambda) = \int_0^T \rho(s)\Delta t(s) ds + \lambda \left( \int_0^T \frac{1}{\Delta t(s)} ds - \tilde{N} \right), \]  

(6.29)

for which we seek to minimize the pathwise squared error

\[ (g(X_T) - g(\overline{X}_T))^2 = \int_0^T \rho(s)\Delta t(s) ds \]

under the constraint that

\[ \int_0^T \frac{1}{\Delta t(s)} ds = \tilde{N}, \]

for a fixed number of time steps, \( \tilde{N} \), and the implicit constraint that the error indicators are equilibrated,

\[ \overline{r}_n = \overline{\rho}_n \Delta t_n^2 = \frac{(g(X_T) - g(\overline{X}_T))^2}{\tilde{N}}, \quad n = 0, 1, \ldots, \tilde{N} - 1. \]  

(6.30)

Minimizing (6.29) yields

\[ \Delta t_n = \sqrt{\frac{(g(X_T) - g(\overline{X}_T))^2}{\tilde{N} \rho(t_n)}} \quad \text{and} \quad \text{MSE}_{\text{adaptive}} \leq \frac{1}{\tilde{N}} \mathbb{E} \left[ \left( \int_0^T \sqrt{\rho(s)} ds \right)^2 \right], \]

(6.31)

where the above inequality follows from using Hölder’s inequality,

\[
\mathbb{E} \left[ (g(X_T) - g(\overline{X}_T))^2 \right] = \frac{1}{\sqrt{\tilde{N}}} \mathbb{E} \left[ |g(X_T) - g(\overline{X}_T)| \int_0^T \sqrt{\rho(s)} ds \right] \\
\leq \frac{1}{\sqrt{\tilde{N}}} \sqrt{\mathbb{E} \left[ |g(X_T) - g(\overline{X}_T)|^2 \right]} \sqrt{\mathbb{E} \left[ \left( \int_0^T \sqrt{\rho(s)} ds \right)^2 \right]}. 
\]

In comparison, we notice that if a uniform mesh is used, the MSE becomes

\[ \text{MSE}_{\text{uniform}} = \frac{T}{\tilde{N}} \mathbb{E} \left[ \int_0^T \rho(s) ds \right]. \]  

(6.32)
A consequence of observations (6.31) and (6.32) is that for many low-regularity problems, for instance, if \( \rho(s) = s^{-p} \) with \( p \in [1, 2) \), adaptive time-stepping Euler–Maruyama methods may produce more accurate solutions (measured in the MSE) than are obtained using the uniform time-stepping Euler–Maruyama method under the same computational budget constraints.

### 6.2.2.1 Mesh Refinement Strategy

To equilibrate the error indicators (6.30), we propose an iterative mesh refinement strategy to identify the largest error indicator and then refining the corresponding time step by halving it. To compute the error indicators prior to refinement, the algorithm first computes the numerical SDE solution, \( \overline{X}_{t_n} \), and the corresponding first variation \( \varphi_{x,n} \) (using equations (6.3) and (6.25) respectively) on the initial mesh, \( \Delta t^{(0)} \). Thereafter, the error indicators \( \tau_n \) are computed by Equation (6.27) and the mesh is refined a prescribed number of times, \( N_{\text{refine}} \), as follows:

1. **Find the largest error indicator**

   \[ n^* := \arg \max_n \tau_n, \quad (6.33) \]

   and refine the corresponding time step by halving

   \[ (t_{n^*}, t_{n^*+1}) \rightarrow \left( t_{n^*}, \frac{t_{n^*} + t_{n^*+1}}{2} \right), \quad (6.34) \]

   and increment the number of refinements by one.

2. **Update the values of the error indicators**, either by recomputing the whole problem or locally by interpolation, cf. Section 6.2.2.3.

3. **Go to step (C.4) if** \( N_{\text{refine}} \) **mesh refinements have been made; otherwise, return**
(C.4) (Postconditioning) Do a last sweep over the mesh and refine by halving every
time step that is strictly larger than $\Delta t_{\text{max}}$, where $\Delta t_{\text{max}} = O\left(\hat{N}^{-1}\right)$ denotes
the maximum allowed step size.

The postconditioning step (C.4) ensures that all time steps become infinitesimally
small as the number of time steps $N \to \infty$ with such a rate of decay that condition
(M.2) in Theorem 6.2.1 holds and is thereby one of the necessary conditions from
Lemma 6.2.5 to ensure strong convergence for the numerical solutions of the MSE
adaptive Euler–Maruyama algorithm. However, the strong convergence result should
primarily be interpreted as a motivation for introducing the postconditioning step
(C.4) since Theorem 6.2.1’s assumption (M.1), namely that the mesh points are stop-
ping times for which $t_n \in \mathcal{F}_{t_{n-1}}$, will not hold in general for our adaptive algorithm.

6.2.2.2 Wiener Path Refinements

When a time step is refined, as described in (6.34), the Wiener path must be refined
correspondingly. The value of the Wiener path at the midpoint between $W_{t_n^*}$ and
$W_{t_{n^*+1}}$ can be generated by Brownian bridge interpolation,

$$W_{t_{n^*+1}}^{\text{new}} = \frac{W_{t_n^*} + W_{t_{n^*+1}}}{2} + \xi \frac{\sqrt{\Delta t_{n^*}}}{2},$$

(6.35)

where $\xi \sim N(0,1)$, cf. [27]. See Figure 6.1 for an illustration of Brownian bridge
interpolation applied to numerical solutions of an Ornstein-Uhlenbeck SDE.

6.2.2.3 Updating the Error Indicators

After the refinement of an interval, $(t_n^*, t_{n+1}^*)$, and its Wiener path, error indicators
must also be updated before moving on to determine which interval is next in line for
refinement. There are different ways of updating error indicators. One expensive but
more accurate option is to recompute the error indicators completely by first solving the forward problem (6.3) and the backward problem (6.25). A less costly but also less accurate alternative is to update only the error indicators locally at the refined time step by one forward and backward numerical solution step, respectively:

\[
X_{tn+1}^{new} = X_{tn} + a(t_n, X_{tn}) \Delta t_n^{new} + b(t_n, X_{tn}) \Delta W_{n}^{new},
\]

\[
\overline{\phi}_{x,n+1} = c_x(t_n, X_{tn}) \overline{\phi}_{x,n+1}.
\]

(6.36)

Thereafter, we compute the resulting error density, \(\rho_{n+1}^{new}\), by Equation (6.26), and finally update the error locally by

\[
\tau_n = \rho_n (\Delta t_n) \overline{\phi}_{n}^{new}, \quad \tau_n^{+1} = \rho_{n+1} (\Delta t_{n+1}^{new}) \overline{\phi}_{n+1}^{new}.
\]

(6.37)

As a compromise between cost and accuracy, we here propose the following mixed approach to updating error indicators post refinement: With \(N_{\text{refine}}\) denoting the prescribed number of refinement iterations of the input mesh, let all error indicators be completely recomputed every \(\tilde{N} = \mathcal{O}(\log(N_{\text{refine}}))\)-th iteration, whereas for the remaining \(N_{\text{refine}} - \tilde{N}\) iterations, only local updates of the error indicators are computed. Following this approach, the computational cost of refining a mesh holding \(N\) time steps into a mesh of \(2N\) time steps becomes \(\mathcal{O}(N \log(N)^2)\). Observe that the asymptotically dominating cost is to sort the mesh’s error indicators \(\mathcal{O}(\log(N))\) times. To anticipate the computational cost for the MSE adaptive MLMC algorithm, this implies that the cost of generating an MSE adaptive realization pair is

\[
\text{Cost}(\Delta t) = \mathcal{O}(\ell^2 2^\ell).
\]

6.2.2.4 Pseudocode

The mesh refinement and the computation of error indicators are presented in Algorithms 1 and 2, respectively.
Algorithm 1 meshRefinement

**Input:** Mesh $\Delta t$, Wiener path $W$, number of refinements $N_{\text{refine}}$, maximum time step $\Delta t_{\text{max}}$

**Output:** Refined mesh $\Delta t$ and Wiener path $W$.

Set the number of re-computations of all error indicators to a number $\tilde{N} = O(\log(N_{\text{refine}}))$ and compute the refinement batch size $\hat{N} = \lceil N_{\text{refine}} / \tilde{N} \rceil$.

for $i = 1$ to $\tilde{N}$ do

- Completely update the error density by applying $[\tau, X, \varphi_x, \rho] = \text{computeErrorIndicators}(\Delta t, W)$.

  if $N_{\text{refine}} > 2\hat{N}$ then

    - Set the below for-loop limit to $J = \hat{N}$.
  
  else

    - Set $J = N_{\text{refine}}$.
  
  end if

for $j = 1$ to $J$ do

- Locate the largest error indicator $\tau_{n^*}$ using Equation (6.33).

  - Refine the interval $(t_{n^*}, t_{n^*} + 1)$ by the halving (6.34), add a midpoint value $W_{n^* + 1}$ to the Wiener path by the Brownian bridge interpolation (6.35), and set $N_{\text{refine}} = N_{\text{refine}} - 1$.

  - Locally update the error indicators $r_{n^*}^{\text{new}}$ and $r_{n^* + 1}^{\text{new}}$ by the steps (6.36) and (6.37).

end for

end for

Do a final sweep over the mesh and refine all time steps of the input mesh which are strictly larger than $\Delta t_{\text{max}}$.

Algorithm 2 computeErrorIndicators

**Input:** mesh $\Delta t$, Wiener path $W$.

**Output:** error indicators $\tau$, path solutions $X$ and $\varphi_x$, error density $\rho$.

Compute the SDE path $X$ using the Euler–Maruyama algorithm (6.3).

Compute the first variation $\varphi_x$ using the backward algorithm (6.25).

Compute the error density $\rho$ and error indicators $\tau$ by the formulas (6.26) and (6.27), respectively.

6.2.3 Numerical Examples

To illustrate the procedure for computing error indicators and the performance of the adaptive algorithm, we now present four SDE example problems. To keep matters relatively elementary, the dual solutions, $\varphi_x(t)$, for these examples are derived not
from \textit{a posteriori} but \textit{a priori} analysis. This approach results in adaptively generated mesh points which for all problems in this section will contain mesh points which are stopping times for which \( t_n \in \mathcal{F}_{t_{n-1}} \) for all \( n \in \{1, 2, \ldots, N\} \). In Examples 6.2.6, 6.2.7 and 6.2.8 it is straightforward to verify that the other assumptions of the respective single- and multi-dimensional MSE error expansions of Theorems 6.2.1 and 6.A.2 hold, meaning that the adaptive approach produces numerical solutions whose MSE to leading order are bounded by the respective error expansions (6.14) and (6.67).

**Example 6.2.6.** We consider the classical geometric Brownian motion problem

\[
\begin{align*}
    dX_t &= X_t dt + X_t dW_t, \quad X_0 = 1, \\
\end{align*}
\]

for which we seek to minimize the MSE

\[
    E[(X_T - \bar{X}_T)^2] = \min!, \quad N \text{ given,} \quad (6.38)
\]

at the final time, \( T = 1 \), (cf. the goal (B.1)). One may derive that the dual solution of this problem is of the form

\[
    \varphi_x(X_t, t) = \partial_{X_t} X_t^{X_t,t} = \frac{X_T}{X_t},
\]

which leads to the error density

\[
    \rho(t) = \frac{(b_x b)^2(X_t, t) (\varphi_x(X_t, t))^2}{2} = \frac{X_T^2}{2}.
\]

We conclude that uniform time-stepping is optimal. A further reduction of the MSE could be achieved by allowing the number of time steps to depend on the magnitude of \( X_T^2 \) for each realization. This is however outside the scope of the considered refinement goal (B.1), where we assume the number of time steps, \( N \), is fixed for all realizations.
and would be possible only to a very weak degree under the slight generalization of (B.1) given in assumption (M.2) of Theorem 6.2.1.

**Example 6.2.7.** Our second example is the two-dimensional (2D) SDE problem

\[ dW_t = 1dW_t, \quad W_0 = 0, \]
\[ dX_t = W_tdW_t, \quad X_0 = 0. \]

Here, we seek to minimize the MSE \( E[(X_T - \bar{X}_T)^2] \) for the observable

\[ X_T = \int_0^T W_t dW_t \]

at the final time \( T = 1 \). With the diffusion matrix represented by

\[ b((W_t, X_t), t) = \begin{bmatrix} 1 \\ W_t \end{bmatrix}, \]

and observing that

\[ \partial_{X_t} X_t^{X_t,t} = \partial_{X_t} \left( X_t + \int_t^T W_s dW_s \right) = 1, \]

it follows from the error density in multi-dimensions in Equation (6.65) that \( \rho(t) = \frac{1}{2} \).

We conclude that uniform time-stepping is optimal for this problem as well.

**Example 6.2.8.** Next, we consider the three-dimensional (3D) SDE problem

\[ dW_t^{(1)} = 1dW_t^{(1)}, \quad W_0^{(1)} = 0, \]
\[ dW_t^{(2)} = 1dW_t^{(2)}, \quad W_0^{(2)} = 0, \]
\[ dX_t = W_t^{(1)} dW_t^{(2)} - W_t^{(2)} dW_t^{(1)}, \quad X_0 = 0, \]
where \( W_t^{(1)} \) and \( W_t^{(2)} \) are independent Wiener processes. Here, we seek to minimize the MSE \( \mathbb{E}[(X_T - \overline{X}_T)^2] \) for the Levy area observable

\[
X_T = \int_0^T (W_t^{(1)} dW_t^{(2)} - W_t^{(2)} dW_t^{(1)}),
\]

at the final time, \( T = 1 \). Representing the diffusion matrix by

\[
b((W_t, X_t), t) = \begin{pmatrix}
1 & 0 \\
0 & 1 \\
-W_t^{(1)} & W_t^{(2)}
\end{pmatrix},
\]

and observing that

\[
\partial_{X_t} X_t^{X_t, t} = \partial_{X_t} \left( X_t + \int_t^T (W_s^{(1)} dW_s^{(2)} - W_s^{(2)} dW_s^{(1)}) \right) = 1,
\]

it follows from Equation (6.65) that \( \rho(t) = 1 \). We conclude that uniform time-stepping is optimal for computing Levy areas.

Example 6.2.9. As the last example, we consider the 2D SDE

\[
\begin{align*}
    dW_t &= 1dW_t, & W_0 &= 0, \\
    dX_t &= 3(W_t^2 - t)dW_t, & X_0 &= 0.
\end{align*}
\]

We seek to minimize the MSE (6.38) at the final time \( T = 1 \). For this problem, it may be shown by Itô calculus that the pathwise exact solution is \( X_T = W_T^3 - 3W_T T \). Representing the diffusion matrix by

\[
b((W_t, X_t), t) = \begin{pmatrix}
1 \\
3(W_t^2 - t)
\end{pmatrix},
\]
equation (6.65) implies that \( \rho(t) = 18W_t^2 \). This motivates the use of discrete error indicators, \( \mathcal{T}_n = 18W_n^2\Delta t_n^2 \), in the mesh refinement criterion. For this problem, we may not directly conclude that the error expansion (6.67) holds since the diffusion coefficient does not fulfill the assumption in Theorem 6.A.2. Although we will not include the details here, it is easy to derive that \( \partial_x^j X_T^{x,t} = 0 \) for all \( j > 1 \) and to prove that the MSE leading-order error expansion also holds for this particular problem by following the steps of the proof of Theorem 6.2.4. In Figure 6.2, we compare the uniform and adaptive time-stepping Euler–Maruyama algorithms in terms of MSE vs. the number of time steps, \( N \). Estimates for the MSE for both algorithms are computed by MC sampling using \( M = 10^6 \) samples. This is a sufficient sample size to render the MC estimates’ statistical error negligible. For the adaptive algorithm, we have used the following input parameter in Algorithm 1: uniform input mesh, \( \Delta t \), with step size \( 2/N \) (and \( \Delta t_{\text{max}} = 2/N \)). The number of refinements is set to \( N_{\text{refine}} = N/2 \). We observe that the algorithms have approximately equal convergence rates, but, as expected, the adaptive algorithm is slightly more accurate than the uniform time-stepping algorithm.

Figure 6.2: Comparison of the performance of uniform and adaptive time-stepping Euler–Maruyama numerical integration for Example 6.2.9 in terms of MSE vs. number of time steps.
6.3 Extension of the Adaptive Algorithm to the Multilevel Setting

In this section, we incorporate the MSE adaptive time-stepping algorithm presented in the preceding section into an MSE adaptive MLMC algorithm for weak approximations. First, we shortly recall the approximation goal and important concepts for the MSE adaptive MLMC algorithm, such as the structure of the adaptive mesh hierarchy and MLMC error control. Thereafter, the MLMC algorithm is presented in pseudocode form.

6.3.1 Notation and Objective

For a tolerance, $\epsilon_T > 0$, and confidence, $0 < 1 - \delta < 1$, we recall that our objective is to construct an adaptive time-stepping MLMC estimator, $A_{ML}$, which meets the approximation constraint

$$\mathbb{P}(\left| E[g(X_T)] - A_{ML} \right| \leq \epsilon_T) \geq 1 - \delta. \quad (6.39)$$

We denote the multilevel estimator by

$$A_{ML} := \sum_{\ell=0}^{L} \sum_{i=1}^{M_{\ell}} \frac{\Delta_{\ell}g(\omega_{i,\ell})}{M_{\ell}},$$

where

$$\Delta_{\ell}g(\omega) := \begin{cases} g\left(\overline{X}_T^{[0]}(\omega)\right), & \text{if } \ell = 0, \\ g\left(\overline{X}_T^{[\ell]}(\omega)\right) - g\left(\overline{X}_T^{[\ell-1]}(\omega)\right), & \text{else.} \end{cases}$$

Section 6.1.2.5 presents further details on MLMC notation and parameters.
6.3.1.1 The Mesh Hierarchy

A realization, $\Delta t g(\omega_{i,\ell})$, is generated on a nested pair of mesh realizations

$$\ldots \subset \Delta t^{\ell} (\omega_{i,\ell}) \subset \Delta t^{(\ell)} (\omega_{i,\ell}).$$

Subsequently, mesh realizations are generated step by step from a prescribed and deterministic input mesh, $\Delta t^{(-1)}$, holding $N-1$ uniform time steps. First, $\Delta t^{(-1)}$ is refined into a mesh, $\Delta t^{(0)}$, by applying Algorithm 1, namely

$$[\Delta t^{(0)}, W^{(0)}] = \text{meshRefinement} (\Delta t^{(-1)}, W^{(-1)}, N_{\text{refine}} = N_{-1}, \Delta t_{\text{max}} = N^{-1}_0).$$

The mesh refinement process is iterated until meshes $\Delta t^{(\ell-1)}$ and $\Delta t^{(\ell-1)}$ are produced, with the last couple of iterations being

$$[\Delta t^{(\ell-1)}, W^{(\ell-1)}] = \text{meshRefinement} (\Delta t^{(\ell-2)}, W^{(\ell-2)}, N_{\text{refine}} = N_{\ell-2}, \Delta t_{\text{max}} = N^{-1}_{\ell-1}),$$

and

$$[\Delta t^{(\ell)}, W^{(\ell)}] = \text{meshRefinement} (\Delta t^{(\ell-1)}, W^{(\ell-1)}, N_{\text{refine}} = N_{\ell-1}, \Delta t_{\text{max}} = N^{-1}_{\ell}).$$

The output realization for the difference $\Delta t g(\omega_{i,\ell}) = g\left(\bar{X}_T^{(\ell)} (\omega_{i,\ell})\right) - g\left(\bar{X}_T^{(\ell-1)} (\omega_{i,\ell})\right)$ is thereafter generated on the output temporal mesh and Wiener path pairs, $(\Delta t^{(\ell-1)}, W^{(\ell-1)})$ and $(\Delta t^{(\ell)}, W^{(\ell)})$.

For later estimates of the computational cost of the MSE adaptive MLMC algorithm, it is useful to have upper bounds on the growth of the number of time steps in the mesh hierarchy, $\{\Delta t^{(\ell)}\}_\ell$, as $\ell$ increases. Letting $|\Delta t|$ denote the number of time steps in a mesh, $\Delta t$ (i.e., the cardinality of the set $\Delta t = \{\Delta t_0, \Delta t_1, \ldots\}$), the
following bounds hold
\[ N_\ell \leq |\Delta t^{(\ell)}| < 2N_\ell \quad \forall \ell \in \mathbb{N}_0. \]

The lower bound follows straightforwardly from the mesh hierarchy refinement procedure described above. To show the upper bound, notice the maximum number of mesh refinements going from a level \( \ell - 1 \) mesh, \( \Delta t^{(\ell-1)} \) to a level \( \ell \) mesh, \( \Delta t^{(\ell)} \) is \( 2N_{\ell-1} - 1 \). Consequently,
\[
|\Delta t^{(\ell)}| \leq |\Delta t^{(-1)}| + \sum_{j=0}^{\ell-1} \text{Maximum number of refinements going from } \Delta t^{(j-1)} \text{ to } \Delta t^{(j)}
\]
\[
\leq N_{\ell-1} + 2 \sum_{j=0}^{\ell-1} N_{j-1} - (\ell + 1) < 2N_\ell.
\]

**Remark 6.3.1.** For the telescoping property \( E[A_{\mathcal{M}_\ell}] = E[g(X_T^{(\ell)})] \) to hold, it is not required that the adaptive mesh hierarchy is nested, but non-nested meshes make it more complicated to compute Wiener path pairs \((W^{(\ell-1)}, W^{(\ell)})_\omega\). In the numerical tests leading to this work, we tested both nested and non-nested adaptive meshes and found both options performing satisfactorily.

### 6.3.2 Error Control

The error control for the adaptive MLMC algorithm follows the general framework of a uniform time-stepping MLMC, but for the sake of completeness, we recall the error control framework for the setting of weak approximations. By splitting
\[
|E[g(X_T)] - A_{\mathcal{M}_\ell}| \leq \left| E[g(X_T) - g(X_T^{(L)})] \right| =_{x_T} E[g(X_T^{(L)}) - A_{\mathcal{M}_\ell}] =_{x_S}
\]
and
\[
\epsilon_T = TOL_T + TOL_S, \quad (6.40)
\]
we seek to implicitly fulfill (6.39) by imposing the stricter constraints

\[ \mathcal{E}_T \leq \text{TOL}_T, \quad \text{the time discretization error}, \quad (6.41) \]

\[ P(\mathcal{E}_S \leq \text{TOL}_S) \geq 1 - \delta, \quad \text{the statistical error}. \quad (6.42) \]

### 6.3.2.1 The Statistical Error

Under the moment assumptions stated in [6], Lindeberg’s version of the Central Limit Theorem yields that as \( \epsilon_T \downarrow 0 \),

\[ \frac{\mathcal{A}_{MC} - \mathbb{E}[g(X_T^{(L)})]}{\sqrt{\text{Var}(\mathcal{A}_{MC})}} \xrightarrow{D} N(0, 1). \]

Here, \( \xrightarrow{D} \) denotes convergence in distribution. By construction, we have

\[ \text{Var}(\mathcal{A}_{MC}) = \sum_{\ell=0}^{L} \frac{\text{Var}(\Delta^\ell g)}{M^\ell}. \]

This asymptotic result motivates the statistical error constraint

\[ \text{Var}(\mathcal{A}_{MC}) \leq \frac{\text{TOL}_S^2}{C_C^2}, \quad (6.43) \]

where \( C_C(\delta) \) is the confidence parameter chosen such that

\[ 1 - \frac{1}{\sqrt{2\pi}} \int_{-C_C}^{C_C} e^{-x^2/2} dx = (1 - \delta), \quad (6.44) \]

for a prescribed confidence \( (1 - \delta) \).

Another important question is how to distribute the number of samples, \( \{M^\ell\}_{\ell} \), on the level hierarchy such that both the computational cost of the MLMC estimator is minimized and the constraint \( (6.43) \) is met. Letting \( C^\ell \) denote the expected cost of generating a numerical realization \( \Delta^\ell g(\omega^\ell) \), the approximate total cost of generating
the multilevel estimator becomes

$$
C_{ML} := \sum_{\ell=0}^{L} C_{\ell} M_{\ell}.
$$

An optimization of the number of samples at each level can then be found through minimization of the Lagrangian

$$
\mathcal{L}(M_0, M_1, \ldots, M_L, \lambda) = \lambda \left( \sum_{\ell=0}^{L} \frac{\text{Var}(\Delta_{\ell} g)}{M_{\ell}} - \frac{TOL_s^2}{C^2} \right) + \sum_{\ell=0}^{L} C_{\ell} M_{\ell},
$$

yielding

$$
M_{\ell} = \left[ \frac{C_{\ell}^2}{TOL_s^2} \sqrt{\frac{\text{Var}(\Delta_{\ell} g)}{C_{\ell}}} \sum_{\ell=0}^{L} \sqrt{C_{\ell} \text{Var}(\Delta_{\ell} g)} \right], \quad \ell = 0, 1, \ldots, L.
$$

Since the cost of adaptively refining a mesh, $\Delta t^{(\ell)}$, is $O(N_{\ell} \log(N_{\ell})^2)$, as noted in Section 6.2.2.3, the cost of generating an SDE realization, is of the same order: $C_{\ell} = O(N_{\ell} \log(N_{\ell})^2)$. Representing the cost by its leading-order term and disregarding the logarithmic factor, an approximation to the level-wise optimal number of samples becomes

$$
M_{\ell} = \left[ \frac{C_{\ell}^2}{TOL_s^2} \sqrt{\frac{\text{Var}(\Delta_{\ell} g)}{N_{\ell}}} \sum_{\ell=0}^{L} \sqrt{N_{\ell} \text{Var}(\Delta_{\ell} g)} \right], \quad \ell = 0, 1, \ldots, L. \tag{6.45}
$$

**Remark 6.3.2.** In our MLMC implementations, the variances, $\text{Var}(\Delta_{\ell} g)$, in equation (6.45) are approximated by sample variances. To save memory in our parallel computer implementation, the maximum permitted batch size for a set of realizations, $\{\Delta_{\ell} g(\omega_{i,\ell})\}_i$, is set to 100,000. For the initial batch consisting of $M_{\ell} = \hat{M}$ samples, the sample variance is computed by the standard approach,

$$
\mathcal{V}(\Delta_{\ell} g; M_{\ell}) = \frac{1}{M_{\ell} - 1} \sum_{i=1}^{M_{\ell}} (\Delta_{\ell} g(\omega_{i,\ell}) - A(\Delta_{\ell} g; M_{\ell}))^2.
$$
Thereafter, for every new batch of realizations, \( \{ \Delta_\ell g(\omega_i, \ell) \}_{i=M_{\ell+1}}^{M_{\ell}+M} \) (\( M \) here denotes an arbitrary natural number smaller or equal to 100,000), we incrementally update the sample variance,

\[
V(\Delta_\ell g; M_\ell + M) = \frac{M_\ell}{M_\ell + M} \times V(\Delta_\ell g; M_\ell) + \frac{1}{(M_\ell + M - 1)} \sum_{i=M_{\ell+1}}^{M_{\ell}+M} (\Delta_\ell g(\omega_i, \ell) - A(\Delta_\ell g; M_\ell + M))^2,
\]

and update the total number of samples on level \( \ell \) accordingly, \( M_\ell = M_\ell + M \).

### 6.3.2.2 The Time Discretization Error

To control the time discretization error, we assume that a weak order convergence rate, \( \alpha > 0 \), holds for the given SDE problem when solved with the Euler–Maruyama method, i.e.,

\[
\left| E\left[ g(X_T) - g\left( X_T^{(L)} \right) \right] \right| = \mathcal{O}\left( N^{-\alpha}_L \right),
\]

and we assume that the asymptotic rate is reached at level \( L - 1 \). Then

\[
\left| E\left[ g(X_T) - g\left( X_T^{(L)} \right) \right] \right| = \left| \sum_{\ell=L+1}^{\infty} E[\Delta_\ell g] \right| \leq |E[\Delta_L g]| \sum_{\ell=1}^{\infty} 2^{-\alpha \ell} = \frac{|E[\Delta_L g]|}{2^{\alpha} - 1}.
\]

In our implementation, we assume the weak convergence rate, \( \alpha \), is known prior to sampling and, replacing \( E[\Delta_L g] \) with a sample average approximation in the above inequality, we determine \( L \) by the following stopping criterion:

\[
\max \left( 2^{-\alpha} |A(\Delta_{L-1} g; M_{L-1})|, |A(\Delta_L g; M_L)| \right) \leq \text{TOL}_T,
\]

(cf. Algorithm 3). Here we implicitly assume that the statistical error in estimating the bias condition is not prohibitively large.

A final level \( L \) of order \( \log(\text{TOL}_T^{-1}) \) will thus control the discretization error.
6.3.2.3 Computational Cost

Under the convergence rate assumptions stated in Theorem 6.1.1, it follows that the cost of generating an adaptive MLMC estimator, $A_{ML}$, fulfilling the MSE approximation goal $E[(A_{ML} - E[g(X_T)])^2] \leq \epsilon_T^2$ is bounded by

$$C_{ML} = \sum_{\ell=0}^{L} M_\ell C_\ell \leq \begin{cases} O\left(\epsilon_T^{-2}\right), & \text{if } \beta > 1, \\ O\left(\epsilon_T^{-2} \log(\epsilon_T)^4\right), & \text{if } \beta = 1, \\ O\left(\epsilon_T^{-2+\frac{2-1}{\alpha}} \log(\epsilon_T)^2\right), & \text{if } \beta < 1. \end{cases} \quad (6.47)$$

Moreover, under the additional higher moment approximation rate assumption

$$E\left[\left|g\left(\frac{X_T}{\bar{X}_T}\right) - g(X_T)\right|^{2+\nu}\right] = O\left(2^{-\beta+\nu/2}\right),$$

the complexity bound (6.47) also holds for fulfilling criterion (6.2) asymptotically as $\epsilon_T \downarrow 0$, (cf. [5]).

6.3.3 MLMC Pseudocode

In this section, we present pseudocode for the implementation of the MSE adaptive MLMC algorithm. In addition to Algorithms 1 and 2 presented in Section 6.2.2.4, the implementation consists of Algorithms 3 and 4. Algorithm 3 describes how the stopping criterion for the final level $L$ is implemented and how the multilevel estimator is generated, and Algorithm 4 describes the steps for generating a realization $\Delta_\ell g$.

Remark 6.3.3. For each increment of $L$ in Algorithm 3, all realizations $\Delta_\ell g$ that have been generated up to that point are reused in later computations of the multilevel estimator. This approach, which is common in MLMC, (cf. [8]), seems to work fine in practice although the independence between samples is then lost. Accounting for the lack of independence complicates the convergence analysis.
Algorithm 3 mlmcEstimator

**Input:** TOL\_T, TOL\_S, confidence \(\delta\), initial mesh \(\Delta t^{-1}\), initial number of mesh steps \(N_{-1}\), input weak rate \(\alpha\), initial number of samples \(\tilde{M}\).

**Output:** Multilevel estimator \(A_{\mathcal{M}L}\).

Compute the confidence parameter \(C_C(\delta)\) by (6.44).

Set \(L = -1\).

**while** \(L < 2\) or (6.46), using the input \(\alpha\) for the weak rate, is violated **do**

Set \(M_L = \tilde{M}\), generate a set of realizations \(\{\Delta_t g(\omega_{i,\ell})\}_{i=1}^{M_L}\) by applying adaptiveRealizations(\(\Delta t^{-1}\)).

**for** \(\ell = 0\) to \(L\) **do**

Compute the sample variance \(V(\Delta_t g; M_\ell)\).

end for

**for** \(\ell = 0\) to \(L\) **do**

Determine the number of samples \(M_\ell\) by (6.45).

if new value of \(M_\ell\) is larger than the old value **then**

Compute additional realizations \(\{\Delta_t g(\omega_{i,\ell})\}_{i=M_\ell+1}^{M_{new}}\) by applying adaptiveRealizations(\(\Delta t^{-1}\)).

end if

end for

end while

Compute \(A_{\mathcal{M}L}\) from the generated samples by using formula (6.6).

Algorithm 4 adaptiveRealization

**Input:** Mesh \(\Delta t^{-1}\).

**Outputs:** One realization \(\Delta_t g(\omega)\)

Generate a Wiener path \(W^{-1}\) on the initial mesh \(\Delta t^{-1}\).

**for** \(j = 0\) to \(\ell\) **do**

Refine the mesh by applying \([\Delta t^{(j)}, W^{(j)}] = \text{meshRefinement}(\Delta t^{(j-1)}, W^{(j-1)}, N_{\text{refine}} = N_{j-1}, \Delta t_{\text{max}} = N_{j-1}^{-1})\). 

end for

Compute Euler–Maruyama realizations \((\overline{X}_T^{(\ell-1)}, \overline{X}_T^{(\ell)})(\omega)\) using the mesh pair \((\Delta t^{(\ell-1)}, \Delta t^{(\ell)})(\omega)\) and Wiener path pair \((W^{(\ell-1)}, W^{(\ell)})(\omega)\), cf. (6.3), and return the output 

\[\Delta_t g(\omega) = g\left(\overline{X}_T^{(\ell)}(\omega)\right) - g\left(\overline{X}_T^{(\ell-1)}(\omega)\right).\]
6.4 Numerical Examples for the MLMC Algorithms

To illustrate the implementation of the MSE adaptive MLMC algorithm and to show its robustness and potential efficiency gain over the uniform MLMC algorithm, we present two numerical examples in this section. The first example considers a geometric Brownian motion SDE problem with sufficient regularity, such that there is very little (probably nothing) to gain by introducing adaptive mesh refinement. The example is included to show that in settings where adaptivity is not required, the MSE adaptive MLMC algorithm is not excessively more expensive than the uniform MLMC algorithm. In the second example, we consider an SDE with a random time drift coefficient blow-up of order $t^{-p}$ with $p \in [0.5, 1)$. The MSE adaptive MLMC algorithm performs progressively more efficiently than does the uniform MLMC algorithm as the value of the blow-up exponent $p$ increases. We should add, however, that although we observe numerical evidence for the numerical solutions converging for both examples, all of the assumptions in Theorem 6.2.1 are not fulfilled for our adaptive algorithm, when applied to either of the two examples. We are therefore not able to prove theoretically that our adaptive algorithm converges in these examples.

For reference, the implemented MSE adaptive MLMC algorithm is described in Algorithms 6.1–4, the standard form of the uniform time-stepping MLMC algorithm that we use in these numerical comparisons is presented in Algorithm 5, Appendix 6.B, and a summary of the parameter values used in the examples is given in Table 6.2. Furthermore, all average properties derived from the MLMC algorithms that we plot for the considered examples in Figures 6.3–6.12 below are computed from 100 multilevel estimator realizations, and, when plotted, error bars are scaled to one sample standard deviation.

**Example 6.4.1.** We consider the geometric Brownian motion

$$dX_t = X_t dt + X_t dW_t, \quad X_0 = 1,$$
Table 6.2: List of parameter values used by the MSE adaptive MLMC algorithm and (when required) the uniform MLMC algorithm for the numerical examples in Section 6.4.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description of parameter</th>
<th>Example 6.4.1</th>
<th>Example 6.4.3</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\delta$</td>
<td>Confidence parameter, cf. (6.39).</td>
<td>0.1</td>
<td>0.1</td>
</tr>
<tr>
<td>$\epsilon_T$</td>
<td>Accuracy parameter, cf. (6.39).</td>
<td>$[10^{-3}, 10^{-1}]$</td>
<td>$[10^{-3}, 10^{-1}]$</td>
</tr>
<tr>
<td>TOL$_S$</td>
<td>Statistical error tolerance, cf. (6.39).</td>
<td>$\epsilon_T/2$</td>
<td>$\epsilon_T/2$</td>
</tr>
<tr>
<td>TOL$_T$</td>
<td>Bias error tolerance, cf. (6.39).</td>
<td>$\epsilon_T/2$</td>
<td>$\epsilon_T/2$</td>
</tr>
<tr>
<td>$\Delta t^{(-1)}$</td>
<td>Pre-initial input uniform mesh having the following step size.</td>
<td>1/2</td>
<td>1/2</td>
</tr>
<tr>
<td>$N_0$</td>
<td>Number of time steps in the initial mesh $\Delta t^{(0)}$.</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>$\hat{N}(\ell)$</td>
<td>The number of complete updates of the error indicators in the MSE adaptive algorithm, cf. Algorithm 1.</td>
<td>$\left\lfloor \log(\ell+2) \over \log(2) \right\rfloor$</td>
<td>$\left\lfloor \log(\ell+2) \over \log(2) \right\rfloor$</td>
</tr>
<tr>
<td>$\Delta t_{\text{max}}(\ell)$</td>
<td>Maximum permitted time step size.</td>
<td>$N_{\ell}^{-1}$</td>
<td>$N_{\ell}^{-1}$</td>
</tr>
<tr>
<td>$\Delta t_{\text{min}}$</td>
<td>Minimum permitted time step size (due to the used double-precision binary floating-point format).</td>
<td>$2^{-51}$</td>
<td>$2^{-51}$</td>
</tr>
<tr>
<td>$\hat{M}$</td>
<td>Number of first batch samples for a (first) estimate of the variance $\text{Var}(\Delta \ell g)$.</td>
<td>100</td>
<td>20</td>
</tr>
<tr>
<td>$\alpha_U$</td>
<td>Input weak convergence rate used in the stopping rule (6.46) for uniform time step Euler–Maruyama numerical integration.</td>
<td>1</td>
<td>$(1 - p)$</td>
</tr>
<tr>
<td>$\alpha_A$</td>
<td>Input weak convergence rate used in the stopping rule (6.46) for the MSE adaptive time step Euler–Maruyama numerical integration.</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>
where we seek to fulfill the weak approximation goal \((6.2)\) for the observable, \(g(x) = x\), at the final time, \(T = 1\). The reference solution is \(E[g(X_T)] = e^T\). From Example 6.2.6, we recall that the MSE minimized in this problem by using uniform time steps. However, our a posteriori MSE adaptive MLMC algorithm computes error indicators from numerical solutions of the path and the dual solution, which may lead to slightly non-uniform output meshes. In Figure 6.3, we study how close to uniform the MSE adaptive meshes are by plotting the level-wise ratio, \(E[|\Delta t^{(\ell)}|]/N_\ell\), where we recall that \(|\Delta t^{(\ell)}|\) denotes the number of time steps in the mesh, \(\Delta t^{(\ell)}\), and that a uniform mesh on level \(\ell\) has \(N_\ell\) time steps. As the level, \(\ell\), increases, \(E[|\Delta t^{(\ell)}|]/N_\ell\) converges to 1, and to interpret this result, we recall from the construction of the adaptive mesh hierarchy in Section 6.3 that if \(|\Delta t^{(\ell)}| = N_\ell\), then the mesh, \(\Delta t^{(\ell)}\), is uniform. We thus conclude that for this problem, the higher the level, the more uniform the MSE adaptive mesh realizations generally become.

Figure 6.3: The ratio of the level-wise mean number of time steps \(E[|\Delta t^{(\ell)}|]/N_\ell\) of MSE adaptive mesh realizations to uniform mesh realizations for Example 6.4.3.

Since adaptive mesh refinement is costly and since this problem has sufficient regularity for the first-order weak and MSE convergence rates \((6.4)\) and \((6.5)\) to hold, respectively, one might expect that MSE adaptive MLMC will be less efficient than the uniform MLMC. This is verified in Figure 6.5, which shows that the runtime of the MSE adaptive MLMC algorithm grows slightly faster than the uniform MLMC.
algorithm and that the cost ratio is at most roughly 3.5, in favor of uniform MLMC.
In Figure 6.4, the accuracy of the MLMC algorithms is compared, showing that both algorithms fulfill the goal (6.2) reliably. Figure 6.6 further shows that both algorithms have roughly first-order convergence rates for the weak error $|E[\Delta g]|$ and the variance $\text{Var}(\Delta g)$, and that the decay rates for $M_1$ are close to identical. We conclude that although MSE adaptive MLMC is slightly more costly than uniform MLMC, the algorithms perform comparably in terms of runtime for this example.

Figure 6.4: For a set of $\epsilon_T$ values, 100 realizations of the MSE adaptive multilevel estimator are computed using both MLMC algorithms for Example 6.4.1. The errors $|\mathcal{A}_{MC}(\omega_i; \epsilon_T, \delta) - E[g(X_T)]|$ are respectively plotted as circles (adaptive MLMC) and triangles (uniform MLMC), and the number of multilevel estimator realizations failing the constraint $|\mathcal{A}_{MC}(\omega_i; \epsilon_T, \delta) - E[g(X_T)]| < \epsilon_T$ is written above the $(\epsilon_T^{-1}, \epsilon_T)$ line. Since the confidence parameter is set to $\delta = 0.1$ and less than 10 realizations fail for any of the tested $\epsilon_T$ values, both algorithms meet the approximation goal (6.39).

Figure 6.5: Average runtime vs. $\epsilon_T^{-1}$ for the two MLMC algorithms solving Example 6.4.1.
Figure 6.6: Output for Example 6.4.1 solved with the MSE adaptive and uniform time-stepping MLMC algorithms. (Top) Weak error $|E[\Delta \epsilon g]|$ for solutions at $\epsilon_T = 10^{-3}$. (Middle) Variance $\text{Var}(\Delta \epsilon g)$ for solutions at $\epsilon_T = 10^{-3}$. (Bottom) Average number of samples $E[M_\ell]$.

Remark 6.4.2. The reason why we are unable to prove theoretically that the numerical solution of this problem computed with our adaptive algorithm asymptotically converges to the true solution is slightly subtle. The required smoothness conditions in Theorem 6.2.1 are obviously fulfilled, but due to the local update of the error indicators in our mesh refinement procedure, (cf. Section 6.2.2.3), we cannot prove that the mesh points will asymptotically be stopping times for which $t_n \in F_{t_{n-1}}$ for all $n \in \{1, 2, \ldots, N\}$. If we instead were to use the version of our adaptive algorithm that recomputes all error indicators for each mesh refinement, the definition of the
error density (6.26) implies that, for this particular problem, it would take the same value, \( \bar{\rho}_n = \prod_{k=0}^{N-1} c_x(t_k, X_{t_k})^2 / 2 \), for all indices, \( n \in \{0,1,\ldots,N\} \). The resulting adaptively refined mesh would then become uniform and we could verify convergence, for instance, by using Theorem 6.2.1. Connecting this to the numerical results for the adaptive algorithm that we have implemented here, we notice that the level-wise mean number of time steps ratio, \( E[\Delta t^{(\ell)}]/N_{\ell} \), presented in Figure 6.3 seems to tend towards 1 as \( \ell \) increases, a limit ratio that is achieved only if \( \Delta t^{(\ell)} \) is indeed a uniform mesh.

**Example 6.4.3.** We next consider the SDE

\[
\begin{align*}
    dX_t &= rf(t; \xi)X_t \, dt + \sigma X_t \, dW_t \\
    X_0 &= 1,
\end{align*}
\]

with the low-regularity drift coefficient, \( f(t; \xi) = |t - \xi|^{-p} \), interest rate, \( r = 1/5 \), volatility, \( \sigma = 0.5 \), and observable, \( g(x) = x \), at the final time \( T = 1 \). A new singularity point, \( \xi \in U(1/4, 3/4) \), is sampled for each path, and it is independent from the Wiener paths, \( W_t \). Three different blow-up exponent test cases are considered, \( p = (1/2, 2/3, 3/4) \), and to avoid blow-ups in the numerical integration of the drift function component, \( f(\cdot; \xi) \), we replace the fully explicit Euler–Maruyama integration scheme with the following semi-implicit scheme:

\[
X_{t_{n+1}} = X_{t_n} + \begin{cases} 
rf(t_n; \xi)X_{t_n} \Delta t_n + \sigma X_{t_n} \Delta W_n, & \text{if } f(t_n; \xi) < 2f(t_{n+1}; \xi), \\
rf(t_{n+1}; \xi)X_{t_n} \Delta t_n + \sigma X_{t_n} \Delta W_n, & \text{else}.
\end{cases}
\]

(6.49)

For \( p \in [1/2, 3/4] \) it may be shown that for any singularity point, any path integrated by the scheme (6.49) will have at most one drift-implicit integration step. The reference
mean for the exact solution is given by

\[ E[X_T] = 2 \int_{1/4}^{3/4} \exp \left( \frac{r(x^{1-p} + (1-x)^{1-p})}{1-p} \right) \, dx, \]

and in the numerical experiments, we approximate this integral value by quadrature to the needed accuracy.

### 6.4.0.1 The MSE Expansion for the Adaptive Algorithm

Due to the low-regularity drift present in this problem, the resulting MSE expansion will also contain drift-related terms that formally are of higher order. From the proof of Theorem 6.2.1, equation (6.60), we conclude that, to leading order the MSE is bounded by

\[ E \left[ | \bar{X}_T - X_T |^2 \right] \leq E \left[ \sum_{n=0}^{N-1} \varphi_{x,n}^2 \left( N(a_t + a_x a)^2 \Delta t_n + (b_x b)^2 \right) (t_n, \bar{X}_t_n; \xi) \Delta t_n^2 \right]. \]

This is the error expansion we use for the adaptive mesh refinement (in Algorithm 1) in this example. In Figure 6.7, we illustrate the effect that the singularity exponent, \( p \), has on SDE and adaptive mesh realizations.

### 6.4.0.2 Implementation Details and Observations

Computational tests for the uniform and MSE adaptive MLMC algorithms are implemented with the input parameters summarized in Table 6.2. The weak convergence rate, \( \alpha \), which is needed in the MLMC implementations’ stopping criterion (6.46), is estimated experimentally as \( \alpha(p) = (1-p) \) when using the Euler–Maruyama integrator with uniform time steps, and roughly \( \alpha = 1 \) when using the Euler–Maruyama integrator with adaptive time steps, (cf. Figure 6.8). We further estimate the variance convergence rate to \( \beta(p) = 2(1-p) \), when using uniform time-stepping, and roughly to \( \beta = 1 \) when using MSE adaptive time-stepping, (cf. Figure 6.9). The low
Figure 6.7: (Top) One MSE adaptive numerical realization of the SDE problem (6.48) at different mesh hierarchy levels. The blow-up singularity point is located at $\xi \approx 0.288473$ and the realizations are computed for three singularity exponent values. We observe that as the exponent, $p$, increases, the more jump at $t = \xi$ becomes more pronounced. (Bottom) Corresponding MSE adaptive mesh realizations for the different test cases.

weak convergence rate for uniform MLMC implies that the number of levels $L$ in the MLMC estimator will be become very large, even with fairly high tolerances. Since computations of realizations on high levels are extremely costly, we have, for the sake of computational feasibility, chosen a very low value, $\hat{M} = 20$, for the initial number of samples in both MLMC algorithms. The respective estimators’ use of samples, $M_\ell$, (cf. Figure 6.10), shows that the low number of initial samples is not strictly needed for the the adaptive MLMC algorithm, but for the sake of fair comparisons, we have chosen to use the same parameter values in both algorithms.

From the rate estimates of $\alpha$ and $\beta$, we predict the computational cost of reaching
Figure 6.8: (**Top**) Average errors $|E[\Delta \ell g]|$ for Example 6.4.3 solved with the MSE adaptive MLMC algorithm for three singularity exponent values. (**Bottom**) Corresponding average errors for the uniform MLMC algorithm.

the approximation goal (6.39) for the respective MLMC algorithms to be

$$\text{Cost}_{\text{adp}}(A_{\text{MC}}) = O\left(\log(\epsilon_T)^4 \epsilon_T^{-2}\right) \quad \text{and} \quad \text{Cost}_{\text{unf}}(A_{\text{MC}}) = O\left(\epsilon_T^{-\frac{1}{p}}\right),$$

by using the estimate (6.47) and Theorem 6.1.1 respectively. These predictions fit well with the observed computational runtime for the respective MLMC algorithms, (cf. Figure 6.11). Lastly, we observe that the numerical results are consistent with both algorithms fulfilling the goal (6.39) in Figure 6.12.
Figure 6.9: (Top) Variances \( \text{Var}(\Delta g) \) for Example 6.4.3 solved with the MSE adaptive MLMC algorithm for three singularity exponent values. (Bottom) Corresponding variances for the uniform MLMC algorithm. The more noisy data on the highest levels is due to the low number used for the initial samples, \( \hat{M} = 20 \), and only a subset of the generated 100 multilevel estimator realizations reached the last levels.

### 6.4.1 Computer Implementation

The computer code for all algorithms was written in Java and used the “Stochastic Simulation in Java” library to sample the random variables in parallel from thread-independent MRG32k3a pseudo random number generators, [24]. The experiments were run on multiple threads on Intel Xeon(R) CPU X5650, 2.67GHz processors and the computer graphics were made using the open source plotting library Matplotlib, [18].
Figure 6.10: (Top) Average number of samples $M_\ell$ for for Example 6.4.3 solved with the MSE adaptive MLMC algorithm for three singularity exponent values. (Bottom) Corresponding average number of samples for the uniform MLMC algorithm. The plotted decay rate reference lines, $c2^{-(\beta(p)+1)/2}\ell$, for $M_\ell$ follow implicitly from equation (6.45) (assuming that $\beta(p) = 2(1 - p)$ is the correct variance decay rate).

6.5 Conclusion

We have developed an $a$ posteriori, MSE adaptive Euler–Maruyama time-stepping algorithm and incorporated it into an MSE adaptive MLMC algorithm. The MSE error expansion presented in Theorem 6.2.1 is fundamental to the adaptive algorithm. Numerical tests have shown that MSE adaptive time-stepping may outperform uniform time-stepping, both in the single-level MC setting and in the MLMC setting, (Examples 6.2.9 and 6.4.3). Due to the complexities of implementing adaptive time-stepping, the numerical examples in this work were restricted to quite simple, low-regularity SDE problems with singularities in the temporal coordinate. In the future, we aim to study SDE problems with low-regularity in the state coordinate (prelim-
Figure 6.11: Average runtime vs. $\epsilon_T^{-1}$ for the two MLMC algorithms for three singularity exponent values in Example 6.4.3.

In summary, tests and analysis do however indicate that then some ad hoc molding of the adaptive algorithm is required.

Although a posteriori adaptivity has proven to be a very effective method for deterministic differential equations, the use of information from the future of the numerical solution of the dual problem makes it a somewhat unnatural method to extend to Itô SDE: It can result in numerical solutions that are not $\mathcal{F}_t$-adapted, which consequently may introduce a bias in the numerical solutions. [7] provides an example of a failing adaptive algorithm for SDE. A rigorous analysis of the convergence properties of our developed MSE adaptive algorithm would strengthen the theoretical basis of the algorithm further. We leave this for future work.
Figure 6.12: Approximation errors for both of the MLMC algorithms solving Example 6.4.3. At every $\epsilon_T$ value, circles and triangles represent the errors from 100 independent multilevel estimator realizations of the respective algorithms.
Bibliography


6.A Theoretical Results

6.A.1 Error Expansion for the MSE in 1D

In this section, we derive a leading-order error expansion for the MSE (6.11) in the 1D setting when the drift and diffusion coefficients are respectively mappings of the form $a : [0,T] \times \mathbb{R}$ and $b : [0,T] \times \mathbb{R} \to \mathbb{R}$. We begin by deriving a representation of the MSE in terms of products of local errors and weights.

Recalling the definition of the flow map, $\varphi(x,t) := g(X_T^x)$, and the first variation of the flow map and the path itself given in Section 6.2.1.1, we use the Mean Value Theorem to deduce that

$$g(X_T) - g(X_T^x) = \varphi(0,x_0) - \varphi(0,X_T^x)$$
$$= \sum_{n=0}^{N-1} \varphi(t_n,X_{t_n}) - \varphi(t_{n+1},X_{t_{n+1}})$$
$$= \sum_{n=0}^{N-1} \varphi\left(t_{n+1},X_{t_{n+1}}^{t_{n+1}}\right) - \varphi(t_{n+1},X_{t_{n+1}})$$
$$= \sum_{n=0}^{N-1} \varphi_x\left(t_{n+1},X_{t_{n+1}} + s_n \Delta e_n\right) \Delta e_n,$$

where the local error is given by $\Delta e_n := X_{t_{n+1}}^{t_{n+1}} - X_{t_{n+1}}$ and $s_n \in [0,1]$. Itô expansion of the local error gives the following representation:

$$\Delta e_n = \int_{t_n}^{t_{n+1}} a(t, X_t^x_{t_n,t_n}) - a(t_n, X_{t_n}) \, dt + \int_{t_n}^{t_{n+1}} b(t, X_t^x_{t_n,t_n}) - b(t_n, X_{t_n}) \, dW_t$$

$$= \int_{t_n}^{t_{n+1}} \int_{t_n}^{t} \left( a(t_n + ax + \frac{axx}{2} b^2)(s, X_{t_n}^x_{t_n,t_n}) \right) ds \, dt$$
$$+ \int_{t_n}^{t_{n+1}} \int_{t_n}^{t} \left( b(t_n + bx + \frac{bx}{2} b^2)(s, X_{t_n}^x_{t_n,t_n}) \right) ds \, dW_t$$

$$= \int_{t_n}^{t_{n+1}} \int_{t_n}^{t} \left( a(t_n + ax + \frac{axx}{2} b^2)(s, X_{t_n}^x_{t_n,t_n}) \right) ds \, dt$$
$$+ \int_{t_n}^{t_{n+1}} \int_{t_n}^{t} \left( b(t_n + bx + \frac{bx}{2} b^2)(s, X_{t_n}^x_{t_n,t_n}) \right) ds \, dW_t$$

(6.51)
By equation (6.50) we may express the MSE by the following squared sum

\[
E \left[ (g(X_T) - g(\hat{X}_T))^2 \right] = E \left[ \left( \sum_{n=0}^{N-1} \varphi_x(t_{n+1}, \hat{X}_{t_{n+1}} + s_n \Delta e_n \Delta c_n) \right)^2 \right]
\]

\[
= \sum_{n,k=0}^{N-1} E \left[ \varphi_x(t_{k+1}, \hat{X}_{t_{k+1}} + s_k \Delta e_k) \varphi_x(t_{n+1}, \hat{X}_{t_{n+1}} + s_n \Delta e_n) \Delta e_k \Delta e_n \right].
\]

This is the first step in deriving the error expansion in Theorem 6.2.1. The remaining steps follow in the proof below.

**Proof of Theorem 6.2.1.** The main tools used in proving this theorem are Taylor and Itô–Taylor expansions, Itô isometry, and truncation of higher order terms. For errors attributed to the leading-order local error term, \( \hat{\Delta} b_n \), (cf. equation (6.51)), we do detailed calculations, and the remainder is bounded by stated higher order terms.

We begin by noting that under the assumptions in Theorem 6.2.1 Lemmas 6.2.4 and 6.2.5 respectively verify then the existence and uniqueness of the solution of the SDE \( X \) and the numerical solution \( \hat{X} \), and provide higher order moment bounds for both. Furthermore, due to the assumption of the mesh points being stopping times for which \( t_n \in \mathcal{F}_{t_{n-1}} \) for all \( n \), it follows also that the numerical solution is adapted to the filtration, i.e., \( \hat{X}_{t_n} \in \mathcal{F}_{t_n} \) for all \( n \).

We further need to extend the flow map and the first variation notation from Section 6.2.1.1. Let \( \hat{X}^{x,t_k}_{t_n} \) for \( n \geq k \) denote the numerical solution of the Euler–Maruyama scheme

\[
\hat{X}^{x,t_k}_{t_j+1} = \hat{X}^{x,t_k}_{t_j} + a(t_j, \hat{X}^{x,t_k}_{t_j}) \Delta t_j + b(t_j, \hat{X}^{x,t_k}_{t_j}) \Delta W_j, \quad j \geq k, \quad (6.52)
\]

with initial condition \( X_{t_k} = x \). The first variation of \( \hat{X}^{x,t_k}_{t_n} \) is defined by \( \partial_x \hat{X}^{x,t_k}_{t_n} \). Provided that \( E[|x|^{2p}] < \infty \) for all \( p \in \mathbb{N} \), \( x \in \mathcal{F}_{t_k} \) and provided the assumptions of Lemma 6.2.5 hold, it is straightforward to extend the proof of the lemma to verify
that \((X^{x,t_k}, \partial_x X^{x,t_k})\) converges strongly to \((X^{x,t_k}, \partial_x X^{x,t_k})\) for \(t \in [t_k, T]\),

\[
\max_{k \leq n \leq \tilde{N}} \left( \left( \mathbb{E} \left[ \left| X^{x,t_k}_{t_n} - X^{x,t_k}_{t_n} \right|^{2p} \right] \right)^{1/2p} \right) \leq C \tilde{N}^{-1/2}, \quad \forall p \in \mathbb{N}
\]

and

\[
\max_{k \leq n \leq \tilde{N}} \left( \left( \mathbb{E} \left[ \left| \partial_x X^{x,t_k}_{t_n} - \partial_x X^{x,t_k}_{t_n} \right|^{2p} \right] \right)^{1/2p} \right) \leq C \tilde{N}^{-1/2}, \quad \forall p \in \mathbb{N}
\]

and

\[
\max_{k \leq n \leq \tilde{N}} \left( \max \left( \mathbb{E} \left[ \left| X^{x,t_k}_{t_n} \right|^{2p} \right], \mathbb{E} \left[ \left| \partial_x X^{x,t_k}_{t_n} \right|^{2p} \right] \right) \right) < \infty, \quad \forall p \in \mathbb{N}. \quad (6.53)
\]

In addition to this, we will also make use of moment bounds for the second and third variation of the flow map in the proof, i.e., \(\varphi_{xx}(t, x)\) and \(\varphi_{xxx}(t, x)\). The second variation is described in Section 6.A.2, where it is shown in Lemma 6.A.1 that provided that \(x \in F_t\) and \(\mathbb{E}[|x|^{2p}] < \infty\) for all \(p \in \mathbb{N}\), then

\[
\max \left( \mathbb{E}[|\varphi_{xx}(t, x)|^{2p}], \mathbb{E}[|\varphi_{xxx}(t, x)|^{2p}], \mathbb{E}[|\varphi_{xxxx}(t, x)|^{2p}] \right) < \infty, \quad \forall p \in \mathbb{N}.
\]

Considering the MSE error contribution from the leading order local error terms \(\tilde{\Delta} b_n\), i.e.,

\[
\mathbb{E} \left[ \varphi_x(t_{k+1}, X_{t_{k+1}} + s_k \Delta e_k) \varphi_x(t_{n+1}, X_{t_{n+1}} + s_n \Delta e_n) \tilde{\Delta} b_k \tilde{\Delta} b_n \right], \quad (6.54)
\]

we have for \(k = n\),

\[
\mathbb{E} \left[ \left( \varphi_x(t_{n+1}, X_{t_{n+1}}) + \varphi_{xx}(t_{n+1}, X_{t_{n+1}} + \hat{s}_n \Delta e_n) s_n \Delta e_n \right)^2 \tilde{\Delta} b_n^2 \right] = \mathbb{E} \left[ \varphi_x(t_{n+1}, X_{t_{n+1}})^2 \tilde{\Delta} b_n^2 + o(\Delta t_n^2) \right].
\]

The above \(o(\Delta t_n^2)\) follows from Young’s and Hölder’s inequalities, where the last inequality is derived by applying the moment bounds for multiple Itô integrals described
in [22, Lemma 5.7.5] and under the assumptions (R.1), (R.2), (M.1), (M.2) and (M.3).

This yields

\[
E[\Delta a_n^4|\mathcal{F}_t] \leq CE \left[ \sup_{s \in [t_n, t_{n+1})} \left| a_t + a_x a + \frac{a_{xx} b^2}{2} \right|^4 (s, X_s) | \mathcal{F}_{t_n} \right] \Delta t_n^8, \\
E[\Delta a_n^4|\mathcal{F}_t] \leq CE \left[ \sup_{s \in [t_n, t_{n+1})} \left| a_x b \right|^4 (s, X_s) | \mathcal{F}_{t_n} \right] \Delta t_n^6, \\
E[\Delta b_n^4|\mathcal{F}_t] \leq CE \left[ \sup_{s \in [t_n, t_{n+1})} \left| b_t + b_x a + \frac{b_{xx} b^2}{2} \right|^4 (s, X_s) | \mathcal{F}_{t_n} \right] \Delta t_n^6, \\
E[\Delta b_n^8|\mathcal{F}_t] \leq CE \left[ \sup_{s \in [t_n, t_{n+1})} \left| b_x b \right|^8 (s, X_s) | \mathcal{F}_{t_n} \right] \Delta t_n^8. \\
\]  

(6.55)

And by similar reasoning,

\[
E[\varphi_{xx}(X_{t_{n+1}} + \hat{s}_n \Delta e_n, t_{n+1})^2 s_n^2 \Delta e_n^2 \hat{\Delta} b_n^2] \leq CE[\Delta t_n^4].
\]

For achieving independence between forward paths and dual solutions in the expectations, an Itô–Taylor expansion of \( \varphi_x \) leads to the equality

\[
E[\varphi_x(t_{n+1}, X_{t_{n+1}})^2 \hat{\Delta} b_n^2] = E\left[ \varphi_x(t_{n+1}, X_{t_n})^2 \hat{\Delta} b_n^2 + o(\Delta t_n^2) \right].
\]

Introducing the null set completed \( \sigma \)-algebra

\[
\hat{\mathcal{F}}^n = \sigma(\{W_s \}_{0 \leq s \leq t_n} \lor \sigma(\{W_s - W_{t_{n+1}} \}_{t_{n+1} \leq s \leq T}) \lor \sigma(X_0),
\]

we observe that \( \varphi_x(t_{n+1}, X_{t_n})^2 \) is \( \hat{\mathcal{F}}^n \) measurable by construction, (cf. [27, App. B]).
Moreover, by conditional expectation,

\[
\begin{align*}
E\left[\varphi_x(t_{n+1}, X_{t_n})^2 \Delta b_n^2\right] &= E\left[\varphi_x(t_{n+1}, X_{t_n})^2 E\left[\Delta b_n^2 | \mathcal{F}^n\right]\right] \\
&= E\left[\varphi_x(t_{n+1}, X_{t_n})^2 (b_x b)^2(t_n, X_{t_n}) \frac{\Delta t_n^2}{2} + o(\Delta t_n^2)\right],
\end{align*}
\]

where the last equality follows from using Itô's formula,

\[
(b_x b)^2(t, X^{X_{t_n}, t_n}) = (b_x b)^2(t_n, X_{t_n}) + \int_{t_n}^t \left( \partial_t + a \partial_x + \frac{b^2}{2} \partial_x^2 \right) (b_x b)^2(s, X_s^{X_{t_n}, t_n}) \, ds \\
+ \int_{t_n}^t (b \partial_x (b_x b)^2)(s, X_s^{X_{t_n}, t_n}) \, dW_s, \quad t \in [t_n, t_{n+1}),
\]

to derive that

\[
E\left[\Delta b_n^2 | \mathcal{F}^n\right] = E \left[ \left( \int_{t_n}^{t_{n+1}} \left( \int_{t_n}^t (b_x b)(s, X_s^{X_{t_n}, t_n}) \, dW_s \, dW_t \right)^2 \right) | X_{t_n} \right] \\
= \frac{(b_x b)^2(t_n, X_{t_n}) \Delta t_n^2}{2} + o(\Delta t_n^2).
\]

Here, the higher order \(o(\Delta t_n^2)\) terms are bounded in a similar fashion as the terms in inequality (??), by using [22, Lemma 5.7.5].

For the terms in (6.54) for which \(k < n\), we will show that

\[
\sum_{k,n=0}^{\hat{N}-1} E\left[\varphi_x(t_{k+1}, X_{t_k+1} + s_k \Delta e_k) \varphi_x(t_{n+1}, X_{t_n+1} + s_n \Delta e_n) \Delta b_k \Delta b_n\right] = \sum_{n=0}^{\hat{N}-1} E\left[o(\Delta t_n^2)\right],
\]

which means that the contribution to the MSE from these terms is negligible to leading order. For the use in later expansions, let us first observe by use of the chain
rule that for any \( y \in \mathcal{F}_{t_n} \) with bounded second moment,

\[
\varphi_x(t_{k+1}, y) = g'(X_T^{y_{t_{k+1}}}) \partial_x X_T^{y_{t_{k+1}}}
\]
\[
= g'(X_T^{X_{t_{k+1}} + s_k \Delta e_{k,t_{k+1}}}) \partial_x X_T^{X_{t_{k+1}} + s_k \Delta e_{k,t_{k+1}}}
\]
\[
= \varphi_x(t_{n+1}, X_{t_{n+1}}^{y_{t_{k+1}}}) \partial_x X_{t_{n+1}}^{y_{t_{k+1}}}
\]

and that

\[
\partial_x X_{t_{n+1}}^{X_{t_{k+1}} + s_k \Delta e_{k,t_{k+1}}} = \partial_x X_{t_{n}}^{X_{t_{k+1}} + s_k \Delta e_{k,t_{k+1}}}
\]
\[
+ \int_{t_{n}}^{t_{n+1}} a_x(s, X_s^{X_{t_{k+1}} + s_k \Delta e_{k,t_{k+1}}}) \partial_x X_s^{X_{t_{k+1}} + s_k \Delta e_{k,t_{k+1}}} ds
\]
\[
+ \int_{t_{n}}^{t_{n+1}} b_x(s, X_s^{X_{t_{k+1}} + s_k \Delta e_{k,t_{k+1}}}) \partial_x X_s^{X_{t_{k+1}} + s_k \Delta e_{k,t_{k+1}}} dW_s.
\]

We next introduce the \( \sigma \)-algebra

\[
\hat{\mathcal{F}}_{k,n} := \sigma(\{W_s\}_{0 \leq s \leq t_k}) \lor \sigma(\{W_s - W_{t_{k+1}}\}_{t_{k+1} \leq s \leq t_n}) \lor \sigma(\{W_s - W_{t_{n+1}}\}_{t_{n+1} \leq s \leq T}) \lor \sigma(X_0),
\]

and Itô–Taylor expand the \( \varphi_x \) functions in (6.56) about center points that are \( \hat{\mathcal{F}}_{k,n} \)-measurable.
where

\[
\varphi_x(t_{k+1}, \overline{X}_{t_{k+1}} + s_k \Delta e_k) = \varphi_x(t_{n+1}, X_{t_{n+1}}^{X_{t_{k+1}} + s_k \Delta e_k,t_{k+1}}) \partial_x X_{t_{n+1}}^{X_{t_{k+1}} + s_k \Delta e_k,t_{k+1}}
\]

\[
= \left[ \varphi_x(t_{n+1}, X_{t_{n}}^{X_{t_{k+1}} + s_k \Delta e_k,t_{k+1}}) + \varphi_{xx}(t_{n+1}, X_{t_{n}}^{X_{t_{k+1}} + s_k \Delta e_k,t_{k+1}}) \left( X_{t_{n+1}}^{X_{t_{k+1}} + s_k \Delta e_k,t_{k+1}} - X_{t_{n}}^{X_{t_{k+1}} + s_k \Delta e_k,t_{k+1}} \right) \right]
\]

\[
+ \varphi_{xxx}(t_{n+1}, X_{t_{n}}^{X_{t_{k+1}} + s_k \Delta e_k,t_{k+1}}) \left( \frac{X_{t_{n+1}}^{X_{t_{k+1}} + s_k \Delta e_k,t_{k+1}} - X_{t_{n}}^{X_{t_{k+1}} + s_k \Delta e_k,t_{k+1}}}{2} \right)^2
\]

\[
\times \partial_x X_{t_{n}}^{X_{t_{k+1}} + s_k \Delta e_k,t_{k+1}} + \partial_{xx} X_{t_{n}}^{X_{t_{k+1}} + s_k \Delta e_k,t_{k+1}} (a(t_k, \overline{X}_{t_k}) \Delta t_k + b(t_k, \overline{X}_{t_k}) \Delta W_k + s_k \Delta e_k)
\]

\[
+ \partial_{xxx} X_{t_{n}}^{X_{t_{k+1}} + s_k \Delta e_k,t_{k+1}} (a(t_k, \overline{X}_{t_k}) \Delta t_k + b(t_k, \overline{X}_{t_k}) \Delta W_k + s_k \Delta e_k,t_{k+1})
\]

\[
\times \left( \frac{a(t_k, \overline{X}_{t_k}) \Delta t_k + b(t_k, \overline{X}_{t_k}) \Delta W_k + s_k \Delta e_k}{2} \right)^2
\]

\[
+ \int_{t_n}^{t_{n+1}} a(s, X_{s}^{X_{t_{k+1}} + s_k \Delta e_k,t_{k+1}}) \partial_x X_{s}^{X_{t_{k+1}} + s_k \Delta e_k,t_{k+1}} ds
\]

\[
+ \int_{t_n}^{t_{n+1}} b(s, X_{s}^{X_{t_{k+1}} + s_k \Delta e_k,t_{k+1}}) \partial_x X_{s}^{X_{t_{k+1}} + s_k \Delta e_k,t_{k+1}} dW_s \right],
\]

(6.57)

where

\[
X_{t_{n+1}}^{X_{t_{k+1}} + s_k \Delta e_k,t_{k+1}} - X_{t_{n}}^{X_{t_{k+1}} + s_k \Delta e_k,t_{k+1}}
\]

\[
= \int_{t_n}^{t_{n+1}} a(s, X_{s}^{X_{t_{k+1}} + s_k \Delta e_k,t_{k+1}}) ds + \int_{t_n}^{t_{n+1}} b(s, X_{s}^{X_{t_{k+1}} + s_k \Delta e_k,t_{k+1}}) dW_s
\]

\[
+ \partial_x X_{t_{n}}^{X_{t_{k+1}} + s_k \Delta e_k,t_{k+1}} (a(t_k, \overline{X}_{t_k}) \Delta t_k + b(t_k, \overline{X}_{t_k}) \Delta W_k + s_k \Delta e_k,t_{k+1}) (a(t_k, \overline{X}_{t_k}) \Delta t_k + b(t_k, \overline{X}_{t_k}) \Delta W_k + s_k \Delta e_k),
\]
and

\[
\varphi_x(t_{n+1}, X_{t_{n+1}} + s_n \Delta e_n) = \varphi_x(t_{n+1}, X_{t_n}^{t_k,t_{k+1}}) + \varphi_{xx}(t_{n+1}, X_{t_n}^{t_k,t_{k+1}}) \Delta \nu_{k,n} + \varphi_{xxx}(t_{n+1}, X_{t_n}^{t_k,t_{k+1}}) \frac{\Delta \nu_{k,n}^2}{2} + \varphi_{xxxx}(t_{n+1}, X_{t_n}^{t_k,t_{k+1}}) \Delta \nu_{k,n}^3 + \varphi_{xxxxx}(t_{n+1}, X_{t_n}^{t_k,t_{k+1}}) \Delta \nu_{k,n}^4,
\]

with

\[
\Delta \nu_{k,n} := a(t_n, X_{t_n}) \Delta t_n + b(t_n, X_{t_n}) \Delta W_n + s_n \Delta e_n + \partial_x X_{t_n}^{t_k,t_{k+1}} \dot{s}_k(a(t_k, X_{t_k}) \Delta t_k + b(t_k, X_{t_k}) \Delta W_k + s_k \Delta e_k).
\]

Plugging the expansions (6.57) and (6.58) into the expectation

\[
E \left[ \varphi_x(t_{k+1}, X_{k+1} + s_k \Delta e_k) \varphi_x(t_{n+1}, X_{n+1} + s_n \Delta e_n) \tilde{\Delta} b_k \tilde{\Delta} b_n \right],
\]

the summands in the resulting expression that only contains products of the first variations vanishes,

\[
E \left[ \varphi_x(t_{n+1}, X_{t_n}^{t_k,t_{k+1}}) \partial_x X_{t_n}^{t_k,t_{k+1}} \varphi_x(t_{n+1}, X_{t_n}^{t_k,t_{k+1}}) \tilde{\Delta} b_k \tilde{\Delta} b_n \right] = E \left[ \tilde{\Delta} b_k \tilde{\Delta} b_k | \mathcal{F}^k,n \right] \varphi_x(t_{n+1}, X_{t_n}^{t_k,t_{k+1}}) \partial_x X_{t_n}^{t_k,t_{k+1}} \varphi_x(t_{n+1}, X_{t_n}^{t_k,t_{k+1}}) = 0.
\]

One can further deduce that all of the the summands in which the product of multiple Itô integrals \( \tilde{\Delta} b_k \) and \( \tilde{\Delta} b_n \) are multiplied only with one additional Itô integral of first-order vanish by using the fact that the inner product of the resulting multiple Itô integrals is zero, cf. [22, Lemma 5.7.2], and by separating the first and second variations from the Itô integrals by taking a conditional expectation with respect to
the suitable filtration. We illustrate this with a couple of examples,

\[
\begin{align*}
E \left[ \varphi_x \left( \bar{X}_{t_{n+1}, X_{t_n}}^{t_k, t_{k+1}} \right) \partial_{XX} X_{t_n}^{t_{k}, t_{k+1}} b(t_k, X_{t_n}) \Delta W_k \varphi_x \left( t_{n+1}, X_{t_n}^{t_k, t_{k+1}} \right) \tilde{\Delta} b_k \tilde{\Delta} b_n \right] \\
= E \left[ \varphi_x \left( t_{n+1}, X_{t_n}^{t_{k}, t_{k+1}} \right) \partial_{XX} X_{t_n}^{t_{k}, t_{k+1}} b(t_k, X_{t_n}) \Delta W_k \varphi_x \left( t_{n+1}, X_{t_n}^{t_k, t_{k+1}} \right) \tilde{\Delta} b_k \\
\times E \left[ \tilde{\Delta} b_n | \tilde{F}^n \right] \right] = 0,
\end{align*}
\]

and

\[
\begin{align*}
E \left[ \varphi_x \left( t_{n+1}, X_{t_n}^{t_{k}, t_{k+1}} \right) \partial_{XX} X_{t_n}^{t_{k}, t_{k+1}} b(t_n, X_{t_n}) \Delta W_n \varphi_x \left( t_{n+1}, X_{t_n}^{t_k, t_{k+1}} \right) \tilde{\Delta} b_k \tilde{\Delta} b_n \right] \\
= E \left[ \varphi_x \left( t_{n+1}, X_{t_n}^{t_{k}, t_{k+1}} \right) \varphi_x \left( t_{n+1}, X_{t_n}^{t_{k}, t_{k+1}} \right) \tilde{\Delta} b_k b(t_n, X_{t_n}) \tilde{\Delta} b_n \right] = 0.
\end{align*}
\]

From these observations, assumption (M.3), inequality (6.55), and, when necessary, additional expansions of integrands to render the leading order integrand either \( \tilde{F}^k \)- or \( \tilde{F}^n \)-measurable and thereby sharpen the bounds (an example of such an expansion is

\[
\tilde{\Delta} b_n = \int_{t_{n}}^{t_{n+1}} \int_{t_{n}}^{t} (b_x b)(s, X_{t_n}^{t_{k}, t_{k+1}}) dW_s dW_t
\]

\[
= \int_{t_{n}}^{t_{n+1}} \int_{t_{n}}^{t} (b_x b)(s, X_{t_n}^{t_{k}, t_{k+1}}) dW_s dW_t + \text{h.o.t.}.
\]

We derive after a laborious computation which we will not include here that

\[
\left| E \left[ \varphi_x \left( t_{k+1}, X_{t_k+1} + s_k \Delta e_k \right) \varphi_x \left( t_{n+1}, X_{t_{n+1}} + s_n \Delta e_n \right) \tilde{\Delta} b_k \tilde{\Delta} b_n \right] \right| \leq C N^{-3/2} \sqrt{E[\Delta t_k^2] E[\Delta t_n^2]}.
\]
This further implies that

\[
\sum_{k,n=0,k\neq n}^{\tilde{N}-1} E\left[ \varphi_x(t_{k+1}, X_{t_{k+1}} + s_k \Delta e_k) \varphi_x(t_{n+1}, X_{t_{n+1}} + s_n \Delta e_n) \tilde{\Delta} b_k \tilde{\Delta} b_n \right] \\
\leq C \tilde{N}^{-3/2} \sum_{k,n=0,k\neq n}^{\tilde{N}-1} \sqrt{E[\Delta t_k^2]E[\Delta t_n^2]} \\
\leq C \tilde{N}^{-3/2} \left( \sum_{n=0}^{\tilde{N}-1} \sqrt{E[\Delta t_n^2]} \right)^2 \\
\leq C \tilde{N}^{-1/2} \sum_{n=0}^{\tilde{N}-1} E[\Delta t_n^2],
\]

such that inequality (6.56) holds.

So far, we have shown that

\[
E\left[ \left( \sum_{n=0}^{N-1} \varphi_x(t_{n+1}, X_{t_{n+1}} + s_n \Delta e_n) \tilde{\Delta} b_n \right)^2 \right] \\
= E\left[ \sum_{n=0}^{N-1} \varphi_x(t_{n+1}, X_{t_n})^2 \frac{(b_x b)^2}{2}(t_n, X_{t_n}) \Delta t_n^2 + o(\Delta t_n^2) \right]. \tag{6.59}
\]

The MSE contribution from the other local error terms, \(\tilde{\Delta} a_n, \tilde{\Delta} a_n\) and \(\tilde{\Delta} b_n\), can also be bounded using the above approach with Itô–Taylor expansions, \(\tilde{\mathcal{F}}^{m,n}\)-conditioning and Itô isometries. This yields that

\[
E\left[ \varphi_x(t_{k+1}, X_{t_{k+1}} + s_k \Delta e_k) \varphi_x(t_{n+1}, X_{t_{n+1}} + s_n \Delta e_n) \tilde{\Delta} a_k \tilde{\Delta} a_n \right] \\
= E\left[ \varphi_x(X_{t_k}, t_k) \varphi_x(t_n, X_{t_n}) \left( \frac{a_t + a_x a + a_x x b^2/2}{2} \right)(t_k, X_{t_k}) \times \\
\left( \frac{a_t + a_x a + a_x x b^2/2}{2} \right)(t_n, X_{t_n}) \Delta t_k^2 \Delta t_n^2 + o(\Delta t_k^2 \Delta t_n^2) \right]. \tag{6.60}
\]
\[ E\left[\varphi_x(t_{k+1}, \overline{X}_{t_{k+1}} + s_k \Delta e_k) \varphi_x(t_{n+1}, \overline{X}_{t_{n+1}} + s_n \Delta e_n) \overline{\Delta a_k \Delta a_n}\right] \]

\[
= \begin{cases} 
E\left[\varphi_x(t_n, \overline{X}_{t_n})^2 \frac{(a_n b_n)^2}{2} (t_n, \overline{X}_{t_n}) \Delta t_n^3 + o(\Delta t_n^3)\right], & \text{if } k = n, \\
\mathcal{O}\left(\tilde{N}^{-3/2} (E[\Delta t_k^3]E[\Delta t_n^3])^{1/2}\right), & \text{if } k \neq n,
\end{cases}
\]

and

\[
E\left[\varphi_x(t_{k+1}, \overline{X}_{t_{k+1}} + s_k \Delta e_k) \varphi_x(t_{n+1}, \overline{X}_{t_{n+1}} + s_n \Delta e_n) \overline{\Delta b_k \Delta b_n}\right] \]

\[
= \begin{cases} 
E\left[\varphi_x(t_n, \overline{X}_{t_n})^2 \frac{(b_n + b_{n+1} + b_{n+2})^2}{3} (t_n, \overline{X}_{t_n}) \Delta t_n^3 + o(\Delta t_n^3)\right], & \text{if } k = n, \\
\mathcal{O}\left(\tilde{N}^{-3/2} (E[\Delta t_k^3]E[\Delta t_n^3])^{1/2}\right), & \text{if } k \neq n.
\end{cases}
\]

Moreover, conservative bounds for error contributions involving products of different local error terms, e.g., \(\overline{\Delta a_k \Delta b_n}\), can be induced from the above bounds and Hölder’s inequality. For example,

\[
\left| \sum_{k,n=0}^{\tilde{N} - 1} \varphi_x(t_{k+1}, \overline{X}_{t_{k+1}} + s_k \Delta e_k) \overline{\Delta a_k \varphi_x(t_{n+1}, \overline{X}_{t_{n+1}} + s_n \Delta e_n) \overline{\Delta b_n}} \right| 
\]

\[
= \left| \sum_{k=0}^{\tilde{N} - 1} \varphi_x(t_{k+1}, \overline{X}_{t_{k+1}} + s_k \Delta e_k) \overline{\Delta a_k} \left(\sum_{k=0}^{\tilde{N} - 1} \varphi_x(t_{n+1}, \overline{X}_{t_{n+1}} + s_n \Delta e_n) \overline{\Delta b_n}\right) \right| 
\]

\[
\leq \sqrt{\sum_{k=0}^{\tilde{N} - 1} \varphi_x(t_{k+1}, \overline{X}_{t_{k+1}} + s_k \Delta e_k) \overline{\Delta a_k}^2} \sqrt{\sum_{n=0}^{\tilde{N} - 1} \varphi_x(t_{n+1}, \overline{X}_{t_{n+1}} + s_n \Delta e_n) \overline{\Delta b_n}^2} 
\]

\[
= \mathcal{O}\left(\tilde{N}^{-1/2} \left(\sum_{n=0}^{\tilde{N} - 1} E[\Delta t_n^2]\right)^{1/2}\right). 
\]

The proof is completed in two replacement steps applied to \(\varphi_x\) on the right-hand side of equality (6.59). First, we replace \(\varphi_x(t_{n+1}, \overline{X}_{t_n})\) by \(\varphi_x(t_n, \overline{X}_{t_n})\). Under the regularity assumed in this theorem, the replacement is possible without introducing
additional leading order error terms as

\[
E[|\varphi_x(t_{n+1}, \overline{X}_{t_n}) - \varphi_x(t_n, \overline{X}_{t_n})|] = E\left[|g'(X_T^{\overline{X}_{t_n}, t_{n+1}})\partial_x X_T^{\overline{X}_{t_n}, t_{n+1}} - g'(X_T^{\overline{X}_{t_n}, t_n})\partial_x X_T^{\overline{X}_{t_n}, t_n}|\right]
\leq E\left[|g'(X_T^{\overline{X}_{t_n}, t_{n+1}}) - g'(X_T^{\overline{X}_{t_n}, t_n})|\partial_x X_T^{\overline{X}_{t_n}, t_{n+1}} | + E\left[|g'(X_T^{\overline{X}_{t_n}, t_n})(\partial_x X_T^{\overline{X}_{t_n}, t_{n+1}} - \partial_x X_T^{\overline{X}_{t_n}, t_n})|\right]
\leq O(\tilde{N}^{-1/2}).
\]

Here, the last equality follows from the assumptions (M.2), (M.3), (R.2), and (R.3), and Lemmas 6.2.4 and 6.2.5.

\[
E\left[|g'(X_T^{\overline{X}_{t_n}, t_{n+1}}) - g'(X_T^{\overline{X}_{t_n}, t_n})|\partial_x X_T^{\overline{X}_{t_n}, t_{n+1}} | \right] 
\leq C \sqrt{E\left[|X_T^{\overline{X}_{t_n}, t_{n+1}} - X_T^{\overline{X}_{t_n+1}, t_{n+1}}|^2\right] E\left[|\partial_x X_T^{\overline{X}_{t_n+1}, t_{n+1}}|^2\right]}
\leq C \left(E\left[|\partial_x X_T^{(1-s)\overline{X}_{t_n+1} + s\overline{X}_{t_n}, t_{n+1}}|^4\right]\right)^{1/4}

\times \left(E\left[\left|\int_{t_n}^{t_{n+1}} a(s, X_s^{\overline{X}_{t_n}, t_n})ds + \int_{t_n}^{t_{n+1}} b(s, X_s^{\overline{X}_{t_n}, t_n})dW_s\right|^4\right]\right)^{1/4}
\leq C \left(E\left[\sup_{t_n \leq s \leq t_{n+1}} |a(s, X_s^{\overline{X}_{t_n}, t_n})|^4 \Delta t_n^4 + \sup_{t_n \leq s \leq t_{n+1}} |b(s, X_s^{\overline{X}_{t_n}, t_n})|^4 \Delta t_n^2\right]\right)^{1/4}
= O(\tilde{N}^{-1/2}),
\]
and that

\[
E\left[ |g'(X_T^{X_{tn}, tn})(\partial_x X_T^{X_{tn}, tn+1} - \partial_x X_T^{X_{tn+1}, tn+1})| \right] \leq C \sqrt{E\left[ \left| \partial_x X_T^{X_{tn}, tn+1} - \partial_x X_T^{X_{tn+1}, tn+1} \right|^2 \right]}
\]

\[
= C \sqrt{E\left[ \left| \partial_x X_T^{X_{tn}, tn+1} - \partial_x X_T^{X_{tn+1}, tn+1} \right|^2 \right]}
\]

\[
\leq C \left( \sqrt{E\left[ \left| \partial_x X_T^{X_{tn+1}, tn+1} \right|^2 \right]} \right)
\]

\[
+ \sqrt{E\left[ \left| \int_{tn}^{tn+1} a_x(s, X_s^{X_{tn+1}, tn})ds + \int_{tn}^{tn+1} b_x(s, X_s^{X_{tn+1}, tn})dW_s \right|^2 \right]}
\]

\[
\leq C \sqrt{E\left[ \left| \partial_x X_T^{X_{tn+1}, tn+1} \right|^2 \right]} + \mathcal{O}(\tilde{N}^{-1/2})
\]

\[
= \mathcal{O}(\tilde{N}^{-1/2})
\]

The last step is to replace the first variation of the exact path \( \varphi_x(t_n, X_{tn}) \) with the first variation of the numerical solution \( \varphi_{x,n} = g'(X_T)\partial_x X_T^{X_{tn}, tn} \). This is also possible without introducing additional leading order error terms by the same assumptions and similar bounding arguments as in the two preceding bounds as

\[
E[\varphi_{x,n} - \varphi_x(t_n, X_{tn})] = E\left[ |g'(X_T)\partial_x X_T^{X_{tn}, tn} - g'(X_T^{X_{tn}, tn})\partial_x X_T^{X_{tn}, tn}| \right]
\]

\[
\leq E\left| g'(X_T)\right| |\partial_x X_T^{X_{tn}, tn} - \partial_x X_T^{X_{tn+1}, tn}| + E\left[ |g'(X_T) - g'(X_T^{X_{tn}, tn})| |\partial_x X_T^{X_{tn}, tn}| \right]
\]

\[
= \mathcal{O}(\tilde{N}^{-1/2})
\]

\[\square\]
6.A.2 Second, Third and Fourth Variations

The proof of Theorem 6.2.1 relies on bounded moments of higher order variations of the flow map $\varphi$. In this section, we will verify that these higher order variations are indeed well defined random variables with all required moments bounded.

To this end, we define the following set of coupled SDE

\begin{align*}
    dY_u^{(1)} &= a(u, Y_u^{(1)}) du + b(u, Y_u^{(1)}) dW_u, \\
    dY_u^{(2)} &= a_x(u, Y_u^{(1)}) Y_u^{(2)} du + b_x(u, Y_u^{(1)}) Y_u^{(2)} dW_u, \\
    dY_u^{(3)} &= \left( a_{xx}(u, Y_u^{(1)}) \left( Y_u^{(2)} \right)^2 + a_x(u, Y_u^{(1)}) Y_u^{(3)} \right) du \\
    &\quad + \left( b_{xx}(u, Y_u^{(1)}) \left( Y_u^{(2)} \right)^2 + b_x(u, Y_u^{(1)}) Y_u^{(3)} \right) dW_u, \\
    dY_u^{(4)} &= \left( a_{xxx}(u, Y_u^{(1)}) \left( Y_u^{(2)} \right)^3 + 3a_{xx}(u, Y_u^{(1)}) Y_u^{(2)} Y_u^{(3)} + a_x(u, Y_u^{(1)}) Y_u^{(4)} \right) du \\
    &\quad + \left( b_{xxx}(u, Y_u^{(1)}) \left( Y_u^{(2)} \right)^3 + 3b_{xx}(u, Y_u^{(1)}) Y_u^{(2)} Y_u^{(3)} + b_x(u, Y_u^{(1)}) Y_u^{(4)} \right) dW_u, \\
    dY_u^{(5)} &= \left( a_{xxxx}(u, Y_u^{(1)}) \left( Y_u^{(2)} \right)^4 + 6a_{xxx}(u, Y_u^{(1)}) \left( Y_u^{(2)} \right)^2 Y_u^{(3)} \right) du \\
    &\quad + \left( a_{xx}(u, Y_u^{(1)}) \left( 3 \left( Y_u^{(3)} \right)^2 + 4Y_u^{(2)} Y_u^{(4)} \right) + a_x(u, Y_u^{(1)}) Y_u^{(5)} \right) du \\
    &\quad + \left( b_{xxxx}(u, Y_u^{(1)}) \left( Y_u^{(2)} \right)^4 + 6b_{xxx}(u, Y_u^{(1)}) \left( Y_u^{(2)} \right)^2 Y_u^{(3)} \right) dW_u \\
    &\quad + \left( b_{xx}(u, Y_u^{(1)}) \left( 3 \left( Y_u^{(3)} \right)^2 + 4Y_u^{(2)} Y_u^{(4)} \right) + b_x(u, Y_u^{(1)}) Y_u^{(5)} \right) dW_u, \\
\end{align*}

(6.61)

defined for $u \in [t, T]$ with the initial condition $Y_t = (x, 1, 0, 0, 0)$. The first component of the vector coincides with equation (6.12), whereas the second one is the first variation of the path from equation (6.16). The last three components can be understood as the second, third and fourth variations of the path, respectively.

Making use of the solution of SDE (6.61), we also define the second, third and
fourth variations as

\[ \varphi_{xx}(t, x) = g'(X^{x,t}_T) \partial_{xx} X^{x,t}_T + g''(X^{x,t}_T)(\partial_x X^{x,t}_T)^2, \]

\[ \varphi_{xxx}(t, x) = g'(X^{x,t}_T) \partial_{xxx} X^{x,t}_T + \cdots + g''''(X^{x,t}_T)(\partial_x X^{x,t}_T)^3, \tag{6.62} \]

\[ \varphi_{xxxx}(t, x) = g'(X^{x,t}_T) \partial_{xxxx} X^{x,t}_T + \cdots + g''''(X^{x,t}_T)(\partial_x X^{x,t}_T)^4. \]

In the sequel, we prove that the solution to equation (6.61) when understood in the integral sense that extends (6.12) is a well defined random variable with bounded moments. Given sufficient differentiability of the payoff \( g \), this results in the boundedness of the higher order variations as required in Theorem 6.2.1.

**Lemma 6.A.1.** Assume that (R.1), (R.2), and (R.3) in Theorem 6.2.1 hold and that for any fixed \( t \in [0, T] \) and \( x \in \mathcal{F}_t \) such that \( E[|X(t)|^{2p}] < \infty \) for all \( p \in \mathbb{N} \). Then, equation (6.61) has pathwise unique solutions with finite moments. That is,

\[ \max_{i \in \{1, 2, \ldots, 5\}} \left( \sup_{u \in [t, T]} E\left[ |Y_u^{(i)}|^{2p} \right] \right) < \infty, \quad \forall p \in \mathbb{N}. \]

Furthermore, the higher variations as defined by equation (6.62) satisfy

\[ \varphi_x(t, x), \varphi_{xx}(t, x), \varphi_{xxx}(t, x), \varphi_{xxxx}(t, x) \in \mathcal{F}_T \]

and for all \( p \in \mathbb{N} \),

\[ \max \{ E[|\varphi_x(t, x)|^{2p}], E[|\varphi_{xx}(t, x)|^{2p}], E[|\varphi_{xxx}(t, x)|^{2p}], E[|\varphi_{xxxx}(t, x)|^{2p}] \} < \infty. \]

**Proof.** We note that the system of SDE (6.61) can be trivially truncated to its first \( d_1 \leq 5 \) elements. That is, the truncated SDE for \( \{Y_u^{(j)}\}_{j=1}^{d_1} \) for \( d_1 < 5 \) has drift and diffusion functions \( \hat{a} : [0, T] \times \mathbb{R}^{d_1} \to \mathbb{R}^{d_1} \) and \( \hat{b} : [0, T] \times \mathbb{R}^{d_1} \to \mathbb{R}^{d_1 \times d_2} \) that do not depend on \( Y_u^{(j)} \) for \( j \geq d_1 \).
This enables verifying existence of solutions for the SDE in stages: first for \((Y^{(1)}, Y^{(2)})\), thereafter for \((Y^{(1)}, Y^{(2)}, Y^{(3)})\), and so forth, proceeding iteratively to add the next component \(Y^{(d_1+1)}\) to the SDE. We shall also exploit this structure for proving the result of bounded moments for each component. The starting point for our proof is Lemma 6.2.4 which guarantees existence, uniqueness and the needed moment bounds for the first two components \(Y^{(1)}\), and \(Y^{(2)}\). It will turn out in the sequel that, thanks to the regularity in the drift and diffusion functions of the SDE \(6.61\), this regularity will cascade further to \(Y^{(j)}\) for \(j \in \{3, 4, 5\}\).

Kloeden and Platen [22, Theorems 4.5.3 and 4.5.4] note that their existence and uniqueness theorems for SDE cannot be modified in order to account for looser regularity conditions, and the proof below is a case in point. Our approach here follows closely the presentation of Kloeden and Platen, with slight modifications on the inequalities that are used to achieve bounds at various intermediate stages of the proof.

As a beginning stage for the proof, let us note that Theorem [20, Theorem 5.2.5] guarantees that the solutions of \(6.61\) are pathwise unique and focus on verifying the claimed results for \(Y^{(3)}_u\).

We define a successive set of approximations \(Y^{(3)}_{u,n}, n \in \mathbb{N}\) by

\[
Y^{(3)}_{u,n+1} = \int_t^u a_{xx}(s, Y^{(1)}_s) \left(Y^{(2)}_s\right)^2 + a_x(s, Y^{(2)}_s) Y^{(3)}_{s,n} ds
+ \int_t^u b_{xx}(s, Y^{(1)}_s) \left(Y^{(2)}_s\right)^2 + b_x(s, Y^{(2)}_s) Y^{(3)}_{s,n} dW_s,
\]

with the initial approximation defined by \(Y^{(3)}_{u,1} = 0\), for all \(u \in [t, T]\). Let us denote by

\[
Q = \int_t^u a_{xx}(s, Y^{(1)}_s) \left(Y^{(1)}_s\right)^2 ds + \int_t^u b_{xx}(s, Y^{(1)}_s) \left(Y^{(2)}_s\right)^2 dW_s \tag{6.63}
\]

the terms that do not depend on the, for the time being, highest order variation \(Y^{(3)}_{u,n}\).
We then have, using Young’s inequality, that
\[
\mathbb{E}\left[|Y_{u,n+1}^{(3)}|^2\right] \leq 3\mathbb{E}[|Q|^2] + 3\mathbb{E}\left[\left(\int_t^u a_x(s, Y_s^{(1)})Y_s^{(3)} ds\right)^2\right] + 3\mathbb{E}\left[\left(\int_t^u b_x(s, Y_s^{(1)})Y_s^{(3)} dW_s\right)^2\right] \\
\leq 3\mathbb{E}[|Q|^2] + 3(u-t)\mathbb{E}\left[\left(\int_t^u |a_x(s, Y_s^{(1)})Y_s^{(3)}|^2 ds\right)^2\right] + 3\mathbb{E}\left[\left(\int_t^u |b_x(s, Y_s^{(1)})Y_s^{(3)}|^2 ds\right)^2\right].
\]

The boundedness of the partial derivatives of the drift and diffusion terms in (6.12) gives us
\[
\mathbb{E}\left[|Y_{u,n+1}^{(3)}|^2\right] \leq 3\mathbb{E}[|Q|^2] + C(u-t+1)\mathbb{E}\left[\left(1 + |Y_{s,n}^{(3)}|^2\right)^2 ds\right].
\]

By induction, we consequently obtain that
\[
\sup_{t \leq u \leq T} \mathbb{E}\left[|Y_{u,n}^{(3)}|^2\right] < \infty, \quad \forall n \in \mathbb{N}.
\]

Now, we set \(\Delta Y_{u,n}^{(3)} = Y_{u,n+1}^{(3)} - Y_{u,n}^{(3)}\). Then
\[
\mathbb{E}[|\Delta Y_{u,n}^{(3)}|^2] \leq 2\mathbb{E}\left[\left(\int_t^u a_x(s, Y_s^{(1)})\Delta Y_s^{(3)} ds\right)^2\right] + 2\mathbb{E}\left[\left(\int_t^u b_x(s, Y_s^{(1)})\Delta Y_s^{(3)} dW_s\right)^2\right] \\
\leq 2(u-t)\int_t^u \mathbb{E}\left[|a_x(s, Y_s^{(1)})\Delta Y_s^{(3)}|^2\right] ds + 2\int_t^u \mathbb{E}\left[|b_x(s, Y_s^{(1)})\Delta Y_s^{(3)}|^2\right] ds \\
\leq C_1 \int_t^u \mathbb{E}\left[|\Delta Y_{s,n-1}^{(3)}|^2\right] ds.
\]

Thus, by Grönwall’s inequality,
\[
\mathbb{E}[|\Delta Y_{u,n}^{(3)}|^2] \leq \frac{C_1^{n-1}}{(n-1)!} \int_t^u (u-s)^{n-1} \mathbb{E}\left[|\Delta Y_{s,1}^{(3)}|^2\right] ds.
\]
Let us next show that $E\left|\Delta Y_{s,1}\right|^2$ is bounded. First,

$$E\left[\Delta Y_{s,1}^{(3)}\right]^2 = E\left[\left(\int_t^u a_x(s, Y_s^{(1)})Y_{s,2}^{(3)} ds + \int_t^u b_x(s, Y_s^{(1)})Y_{u,2}^{(3)} dW_s\right)^2\right]$$

$$\leq C(u - t + 1) \sup_{s \in [t,u]} E\left[\left|Y_{s,2}^{(3)}\right|^2\right].$$

Consequently, there exists a $C \in \mathbb{R}$ such that

$$E\left[\left|\Delta Y_{u,n}^{(3)}\right|^2\right] \leq \frac{C^n(u - t)^n}{n!}, \quad \sup_{u \in [t,T]} E\left[\left|\Delta Y_{u,n}^{(3)}\right|^2\right] \leq \frac{C^n(T - t)^n}{n!}.$$

We define

$$Z_n = \sup_{t \leq u \leq T} \left|\Delta Y_{u,n}^{(3)}\right|,$$

and note that

$$Z_n \leq \int_t^T \left|a_x(s, Y_s^{(1)})Y_{s,n+1}^{(3)} - a_x(s, Y_s^{(1)})Y_{s,n}^{(3)}\right| ds$$

$$+ \sup_{t \leq u \leq T} \left|\int_t^u b_x(s, Y_s^{(1)})Y_{s,n+1}^{(3)} - b_x(s, Y_s^{(1)})Y_{s,n}^{(3)} dW_s\right|.$$

Using Doob’s and Schwartz’s inequalities, as well as the boundedness of $a_x$ and $b_x$,

$$E\left[|Z_n|^2\right] \leq 2(T - t) \int_t^T E\left[\left|a_x(s, Y_s^{(1)})Y_{s,n+1}^{(3)} - a_x(s, Y_s^{(1)})Y_{s,n}^{(3)}\right|^2\right] ds$$

$$+ 8 \int_t^T E\left[\left|b_x(s, Y_s^{(1)})Y_{s,n+1}^{(3)} - b_x(s, Y_s^{(1)})Y_{s,n}^{(3)}\right|^2\right] ds$$

$$\leq \frac{C^n(T - t)^n}{n!},$$
for some $C \in \mathbb{R}$. Using the Markov inequality, we get
\[
\sum_{n=1}^{\infty} P\left(Z_n > n^{-2}\right) \leq \sum_{n=1}^{\infty} \frac{n^4 C^n (T-t)^n}{n!}.
\]

The right-hand side of the equation above converges by the ratio test, whereas the Borel-Cantelli Lemma guarantees the (almost sure) existence of $K^* \in \mathbb{N}$, such that $Z_k < k^2, \forall k > K^*$. We conclude that $Y^{(3)}_{u,n}$ converges uniformly in $L^2(P)$ to the limit $Y^{(3)}_u = \sum_{n=1}^{\infty} \Delta Y^{(3)}_{u,n}$ and that since $\{Y^{(3)}_{u,n}\}_n$ is a sequence of continuous and $\mathcal{F}_u$-adapted processes, $Y^{(3)}_u$ is also continuous and $\mathcal{F}_u$-adapted. Furthermore, as $n \to \infty$,
\[
\left| \int_t^u a_x(s, Y^{(1)}_s) Y^{(3)}_s ds - \int_t^u a_x(s, Y^{(1)}_s) Y^{(3)}_s ds \right| \leq C \int_t^u |Y^{(3)}_{s,n} - Y^{(3)}_s| ds \to 0, \quad \text{a.s.,}
\]
and, similarly,
\[
\left| \int_t^u b_x(s, Y^{(1)}_s) Y^{(3)}_s dW_s - \int_t^u b_x(s, Y^{(1)}_s) Y^{(3)}_s dW_s \right| \to 0, \quad \text{a.s.}
\]
This implies that $Y^{(3)}_u$ is a solution to the SDE (6.61).

Having established that $Y^{(3)}_u$ solves the relevant SDE and that it has a finite second moment, we may follow the principles laid out in [22, Theorem 4.5.4] and show that all even moments of
\[
Y^{(3)}_u = Q + \int_t^u a_x(t, Y^{(1)}_s) Y^{(3)}_s ds + \int_t^u b_x(t, Y^{(1)}_s) Y^{(3)}_s dW_s
\]
are finite. By Itô’s Lemma, we get that for any even integer $l$,
\[
|Y^{(3)}_u|_l^l = \int_t^u |Y^{(3)}_s|_l^{l-2} Y^{(3)}_s \left( a_{xx}(s, Y^{(1)}_s) \left( Y^{(2)}_s \right)^2 + a_x(s, Y^{(1)}_s) Y^{(3)}_s \right) ds \\
+ \int_t^u \frac{l(l-1)}{2} |Y^{(3)}_s|_l^{l-2} \left( b_{xx}(s, Y^{(1)}_s) \left( Y^{(2)}_s \right)^2 + b_x(s, Y^{(1)}_s) Y^{(3)}_s \right)^2 ds \\
+ \int_t^u |Y^{(3)}_s|_l^{l-2} Y^{(3)}_s \left( b_{xx}(s, Y^{(1)}_s) \left( Y^{(2)}_s \right)^2 + b_x(s, Y^{(1)}_s) Y^{(3)}_s \right) dW_s.
\]
Taking expectations, the Itô integral vanishes,

\[
E\left[|Y_s^{(3)}|^l\right] = E\left[\int_t^u \left|Y_s^{(3)}\right|^{l-2} Y_s^{(3)} \left(a_{xx}(s, Y_s^{(1)}) (Y_s^{(2)})^2 + a_x(s, Y_s^{(1)}) Y_s^{(3)}\right) ds\right] \\
+ E\left[\int_t^u \frac{l(l-1)}{2} \left|Y_s^{(3)}\right|^{l-2} \left(b_{xx}(s, Y_s^{(1)}) (Y_s^{(2)})^2 + b_x(s, Y_s^{(1)}) Y_s^{(3)}\right)^2 ds\right].
\]

Using Young’s inequality, denoting the term that does not depend on \(Y_s^{(3)}\) in the first integral by \(Q\), and exploiting the boundedness of \(a_x\), we have that

\[
E\left[|Y_u^{(3)}|^l\right] - Q \leq C \int_t^u E\left[|\tilde{Y}_{3,u}|^l\right] ds \\
+ E\left[\int_t^u \frac{l(l-1)}{2} \left|Y_s^{(3)}\right|^{l-2} \left(b_{xx}(s, Y_s^{(1)}) (Y_s^{(2)})^2 + b_x(s, Y_s^{(1)}) Y_s^{(3)}\right)^2 ds\right].
\]

By the same treatment for the latter integral, lumping together all terms independent of \(Y_s^{(3)}\) and using that \(b_x\) is bounded,

\[
E\left[|Y_u^{(3)}|^l\right] - \tilde{Q} \leq C \int_t^u E\left[|Y_u^{(3)}|^l\right] ds.
\]

Thus, by Grönwall’s inequality, \(E\left[|Y_u^{(3)}|^l\right] < \infty\).

Having established the existence and pathwise uniqueness of \(Y_s^{(3)}\), as well as the finiteness of its moments, verifying the same properties for \(Y_u^{(4)}\) and \(Y_u^{(5)}\) can be done using similar arguments relying, most importantly, on the boundedness of the relevant derivatives of the drift and diffusion functions. Finally, the \(\mathcal{F}_T\)-measurability and moment bounds for the variations of the flow map \(\phi\) up to order four (cf. equation (6.62)) can be verified by a straightforward extension of the argument in the proof of Lemma 6.2.4.
6.A.3 Error Expansion for the MSE in Multiple Dimensions

In this section, we extend the 1D MSE error expansion presented in Theorem 6.2.1 to the multi-dimensional setting.

Consider the SDE

\[ dX_t = a(t, X_t) \, dt + b(t, X_t) \, dW_t, \quad t \in (0, T] \]

\[ X_0 = x_0, \]

(6.64)

where \( X : [0, T] \to \mathbb{R}^{d_1} \), \( W : [0, T] \to \mathbb{R}^{d_2} \), \( a : [0, T] \times \mathbb{R}^{d_1} \to \mathbb{R}^{d_1} \) and \( b : [0, T] \times \mathbb{R}^{d_1} \to \mathbb{R}^{d_1 \times d_2} \). Let further \( x_i \) denote the \( i \)-th component of \( x \in \mathbb{R}^{d_1} \), \( a^{(i)} \), the \( i \)-th component of a drift coefficient and \( b^{(i,j)} \) and \( b^T \) denote the \((i, j)\)-th element and the transpose of the diffusion matrix \( b \), respectively. (To avoid confusion, this derivation does not make use of any MLMC notation, particularly not the multilevel superscript \( \ell \).

Using the Einstein summation convention to sum over repeated indices, but not over the time index \( n \), the 1D local error terms in equation (6.51) generalize into

\[ \Delta a^{(i)}_n = \int_{t_n}^{t_{n+1}} \int_{t_n}^{t} \left( a^{(i)}_t + a^{(i)}_{x_j} a^{(j)}_s + \frac{1}{2} a^{(i)}_{x_j x_k} (b b^T)^{(j,k)}_s \right) \, ds \, dt, \]

\[ \overline{\Delta} a^{(i)}_n = \int_{t_n}^{t_{n+1}} \int_{t_n}^{t} a^{(i)}_{x_j} b^{(j,k)}_s \, dW^{(k)}_s \, dt, \]

\[ \Delta b^{(i)}_n = \int_{t_n}^{t_{n+1}} \int_{t_n}^{t} b^{(i,j)}_t + b^{(i,j)}_{x_k} a^{(k)}_s + \frac{1}{2} b^{(i,j)}_{x_k x_\ell} (b b^T)^{(k,\ell)}_s \, ds \, dW^{(j)}_t, \]

\[ \overline{\Delta} b^{(i)}_n = \int_{t_n}^{t_{n+1}} \int_{t_n}^{t} b^{(i,j)}_{x_k} b^{(k,\ell)}_s \, dW^{(\ell)}_s \, dW^{(j)}_t, \]

where all the above integrand functions in all equations implicitly depend on the state argument \( X^{X_{t_n}, t_n}_s \). In flow notation, \( a^{(i)}_t \) is shorthand for \( a^{(i)}(s, X^{X_{t_n}, t_n}_s) \).

Under sufficient regularity, a tedious calculation similar to the proof of Theo-
rem 6.2.1 verifies that, for a given smooth payoff, \( g : \mathbb{R}^{d_1} \to \mathbb{R} \),

\[
E \left[ (g(X_T) - g(\overline{X}_T))^2 \right] \leq E \left[ \sum_{n=0}^{N-1} \overline{p}_n \Delta t_n^2 + o(\Delta t_n^2) \right],
\]

where

\[
\overline{p}_n := \frac{1}{2} \varphi_{x_i,n} \left( (bb^T)^{k,\ell} \left( b_{x_k} b_{x_{\ell}}^T \right)^{(i,j)}(t_n, X_{t_n}) \varphi_{x_{j,n}} \right). \tag{6.65}
\]

In the multi-dimensional setting, the \( i \)-th component of first variation of the flow map, \( \varphi_x = (\varphi_{x_1}, \varphi_{x_2}, \ldots, \varphi_{x_{d_1}}) \), is given by

\[
\varphi_{x_i}(t, y) = g_{x_j}(X_T^y) \partial_{x_i} \left( X_T^y \right)^{(j)}.
\]

The first variation is defined as the second component to the solution of the SDE,

\[
dY_{s(1,i)}^{(1)} = a^{(i)}(s, Y_{s(1)}^{(1)}) \, ds + b^{(i,j)}(s, Y_{s(1)}^{(1)}) \, dW_s^{(j)},
\]

\[
dY_{s(2,i,j)}^{(2,i,j)} = a_{x_k}^{(i)}(s, Y_{s(1)}^{(1)}) \, Y_{s(2,k,j)}^{(2,k,j)} \, ds + b_{x_k}^{(i,\ell)}(s, Y_{s(1)}^{(1)}) \, Y_{s(2,k,j)}^{(2,k,j)} \, dW_s^{(\ell)},
\]

where \( s \in (t, T) \) and the initial conditions are given by \( Y_t^{(1)} = x \in \mathbb{R}^{d_1}, \ Y_t^{(2)} = I_{d_1} \), with \( I_{d_1} \) denoting the \( d_1 \times d_1 \) identity matrix. Moreover, the extension of the numerical method for solving the first variation of the 1D flow map (6.25) reads

\[
\varphi_{x_i,n} = \varphi_{x_{i,n+1}}(t_n, X_{t_n}) \varphi_{x_{j,n+1}}, \quad n = N - 1, N - 2, \ldots, 0. \tag{6.66}
\]

with the \( j \)-th component of \( c : [0, T] \times \mathbb{R}^{d_1} \to \mathbb{R}^{d_1} \) defined by

\[
c^{(j)}(t_n, X_{t_n}) = X_{t_n}^{(j)} + a^{(j)}(t_n, X_{t_n}) \Delta t_n + b^{(j,k)}(t_n, X_{t_n}) \Delta W_n^{(k)}.
\]

Let \( U \) and \( V \) denote subsets of Euclidean spaces and let us introduce the multi-
index \( \nu = (\nu_1, \nu_2, \ldots, \nu_d) \) to represent spatial partial derivatives of order \(|\nu| := \sum_{j=1}^{d} \nu_j \) on the following short form \( \partial_{x_{\nu}} := \prod_{j=1}^{d} \partial_{x_j}^{\nu_j} \). We further introduce the following function spaces.

\[
C(U; V) := \{ f : U \to V \mid f \text{ is continuous} \},
\]
\[
C_b(U; V) := \{ f : U \to V \mid f \text{ is continuous and bounded} \},
\]
\[
C_b^k(U; V) := \left\{ f : U \to V \mid f \in C(U; V) \text{ and } \frac{d^j}{dx^j} f \in C_b(U; V) \right\}
\]

for all integers \( 1 \leq j \leq k \),
\[
C_b^{k_1, k_2}([0, T] \times U; V) := \left\{ f : [0, T] \times U \to V \mid f \in C([0, T] \times U; V), \text{ and } \partial^j_t \partial^\nu f \in C_b([0, T] \times U; V) \text{ for all integers s.t. } j \leq k_1 \text{ and } 1 \leq j + |\nu| \leq k_2 \right\}.
\]

**Theorem 6.A.2** (MSE leading order error expansion in the multi-dimensional setting). Assume that drift and diffusion coefficients and input data of the SDE (6.64) fulfill

(R.1) \( a \in C_b^{2, 4}([0, T] \times \mathbb{R}^{d_1}; \mathbb{R}^{d_1}) \) and \( b \in C_b^{2, 4}([0, T] \times \mathbb{R}^{d_1}; \mathbb{R}^{d_1 \times d_2}) \),

(R.2) there exists a constant \( C > 0 \) such that

\[
|a(t, x)|^2 + |b(t, x)|^2 \leq C(1 + |x|^2), \quad \forall x \in \mathbb{R}^{d_1} \text{ and } \forall t \in [0, T],
\]

(R.3) \( g' \in C_b^3(\mathbb{R}^{d_1}) \) and there exists a \( k \in \mathbb{N} \) such

\[
|g(x)| + |g'(x)| \leq C(1 + |x|^k), \quad \forall x \in \mathbb{R}^{d_1},
\]

(R.4) for the initial data, \( X_0 \in \mathcal{F}_0 \) and \( E[|X_0|^p] < \infty \) for all \( p \geq 1 \).

Assume further the mesh points \( 0 = t_0 < t_1 < \ldots < t_N = T \).
(M.1) are stopping times for which \( t_n \in \mathcal{F}_{t_{n-1}} \) for \( n = 1, 2, \ldots, N \),

(M.2) for all mesh realizations, there exists a deterministic integer, \( \bar{N} \), and a \( c_1 > 0 \) such that \( c_1 \bar{N} \leq N \leq \bar{N} \) and a \( c_2 > 0 \) such that \( \max_{n \in \{0, 1, \ldots, N-1\}} \Delta t_n < c_2 \bar{N}^{-1} \),

(M.3) and there exists a \( c_3 > 0 \) such that for all \( p \in [1, 8] \) and \( n \in \{0, 1, \ldots, \bar{N} - 1\} \),

\[
\mathbb{E}[\Delta t_n^{2p}] \leq c_3 (\mathbb{E}[\Delta t_n^2])^p.
\]

Then, as \( \bar{N} \) increases,

\[
\mathbb{E}
\left[
  \left(g(X_T) - g(\overline{X}_T)\right)^2
\right]
\approx
\mathbb{E}
\left[
  \sum_{n=0}^{N-1}
  \frac{(\varphi_{x_i}(t_n, X_n))}{2}
  \left(b_{x_i}^T b_{x_j} + b_{x_i}^T b_{x_k} b_{x_j}^T b_{x_k}^T\right)
  \Delta t_n^2 + o(\Delta t_n^2)
\right],
\]

where we have dropped the arguments of the first variation as well as the diffusion matrices for clarity.

Replacing the first variation \( \varphi_{x_i}(t_n, X_n) \) by the numerical approximation \( \overline{\varphi}_{x_i,n} \), as defined in (6.66) and using the error density notation \( \overline{\rho} \) from (6.65), we obtain the following to leading order all-terms-computable error expansion:

\[
\mathbb{E}
\left[
  \left(g(X_T) - g(\overline{X}_T)\right)^2
\right]
\approx
\mathbb{E}
\left[
  \sum_{n=0}^{N-1}
  \overline{\rho}_n \Delta t_n^2 + o(\Delta t_n^2)
\right].
\]  

(6.67)

6.B A Uniform Time Step MLMC Algorithm

The uniform time step MLMC algorithm for MSE approximations of SDE was proposed in [8]. Below, we present the version of that method that we use in the numerical tests in this work for reaching the approximation goal (6.2).

Acknowledgments.

The authors would like to thank Arturo Kohatsu-Higa for his helpful suggestions for improvements in the proof of Theorem 6.2.1.
Algorithm 5 mlmcEstimator

**Input:** \( TOL_T, TOL_S, \text{ confidence } \delta, \text{ input mesh } \Delta t^{(-1)}, \text{ input mesh intervals } N_{-1}, \text{ initial number of samples } \hat{M}, \text{ weak convergence rate } \alpha, \text{ SDE problem.} \)

**Output:** Multilevel estimator \( A_{\mathcal{ML}}. \)

Compute the confidence parameter \( C_C(\delta) \) by (6.44).

Set \( L = -1. \)

**while** \( L < 3 \) or \( (6.46), \) using the input rate \( \alpha, \) is violated **do**

Set \( M_L = \hat{M}, \) generate a set of (Euler–Maruyama) realizations \( \{ \Delta_t g(\omega_{i,t}) \}_{i=1}^{M_L} \) on mesh and Wiener path pairs \( (\Delta_t^{(L-1)}, \Delta_t^{(L)}) \) and \( (W^{(L-1)}, W^{(L)}) \), where the uniform mesh pairs have step sizes \( \Delta t^{(L-1)} = T/N_{L-1} \) and \( \Delta t^{(L)} = T/N_L \), respectively.

**for** \( \ell = 0 \) to \( L \)

Compute the sample variance \( V(\Delta_t g; M_\ell). \)

**end for**

**for** \( \ell = 0 \) to \( L \) do

Determine the number of samples by

\[
M_\ell = \left[ \frac{C_C^2}{TOL_S^2} \sqrt{\frac{\text{Var}(\Delta_t g)}{N_\ell}} \sum_{\ell=0}^{L} \sqrt{N_\ell \text{Var}(\Delta_t g)} \right].
\]

(The equation for \( M_\ell \) is derived by Lagrangian optimization, cf. Section 6.3.2.1)

**if** New value of \( M_\ell \) is larger than the old value **then**

Compute additional (Euler–Maruyama) realizations \( \{ \Delta_t g(\omega_{i,t}) \}_{i=M_\ell+1}^{M_\ell^{\text{new}}} \) on mesh and Wiener path pairs \( (\Delta_t^{(\ell-1)}, \Delta_t^{(\ell)}) \) and \( (W^{(\ell-1)}, W^{(\ell)}) \), where the uniform mesh pairs have step sizes \( \Delta t^{(\ell-1)} = T/(2^\ell N_{-1}) \) and \( \Delta t^{(\ell)} = T/(2^{\ell+1} N_{-1}) \), respectively.

**end if**

**end for**

**end while**

Compute \( A_{\mathcal{ML}} \) using the generated samples by the formula [6.6].

---

**Support** This work was supported by King Abdullah University of Science and Technology (KAUST); by Norges Forskningsråd, research project 214495 LIQCRY; and by the University of Texas, Austin Subcontract (Project Number 024550, Center for Predictive Computational Science). The first author was and the third author is a member of the Strategic Research Initiative on Uncertainty Quantification in Computational Science and Engineering at KAUST (SRI-UQ).
Chapter 7

Article II

Abstract

We provide a bound for the error committed when using a Fourier method to price European options when the underlying follows an exponential Lévy dynamic. The price of the option is described by a partial integro-differential equation (PIDE). Applying a Fourier transformation to the PIDE yields an ordinary differential equation that can be solved analytically in terms of the characteristic exponent of the Lévy process. Then, a numerical inverse Fourier transform allows us to obtain the option price. We present a novel bound for the error and use this bound to set the parameters for the numerical method. We analyse the properties of the bound for a dissipative and pure-jump example. The bound presented is independent of the asymptotic behaviour of option prices at extreme asset prices. The error bound can be decomposed into a product of terms resulting from the dynamics and the option payoff, respectively. The analysis is supplemented by numerical examples that demonstrate results comparable to and superior to the existing literature.

7.1 Introduction

Lévy processes form a rich field within mathematical finance. They allow modelling of asset prices with possibly discontinuous dynamics. An early and probably the best known model involving a Lévy process is the \cite{29} model, which generalises the \cite{5} model. More recently, we have seen more complex models allowing for more general...
dynamics of the asset price. Examples of such models include the [22] model (see also [14]), the Normal Inverse Gaussian model ([4,33]), the Variance Gamma model ([26,27]), and the Carr-Geman-Madan-Yor (CGMY) model ([9,10]). For a good exposition on jump processes in finance we refer to [11] (also see [31] and [17]).

Prices of European options whose underlying asset is driven by the Lévy process are solutions to partial integro-differential Equations (PIDEs) ([1,8,21,30]) that generalise the Black-Scholes equation by incorporating a non-local integral term to account for the discontinuities in the asset price. This approach has also been extended to cases where the option price features path dependence, for instance in [7,16] and [25].

The Lévy-Khintchine formula provides an explicit representation of the characteristic function of a Lévy process (cf, [37]). As a consequence, one can derive an exact expression for the Fourier transform of the solution of the relevant PIDE. Using the inverse fast Fourier transform (iFFT) method, one may efficiently compute the option price for a range of asset prices simultaneously. Furthermore, in the case of European call options, one may use the duality property presented by [15] and iFFT to efficiently compute option prices for a wide range of strike prices.

Despite the popularity of Fourier methods for option pricing, few works can be found on the error analysis and related parameter selection for these methods. A bound for the error not only provides an interval for the precise value of the option, but also suggests a method to select the parameters of the numerical method. An important work in this direction is the one by [24] in which several payoff functions are considered for a rather general set of models, whose characteristic function is assumed to be known. [18] presents the framework and theoretical approach for the error analysis, and establishes polynomial convergence rates for approximations of the option prices. For a more contemporary review on the error committed in various FT-related methods we refer the reader to [6], that extends the classical flat Fourier
methods by deforming the integration contours on the complex plane, studying discretely monitored barrier options studied in [12].

In this work, we present a methodology for studying and bounding the error committed when using FT methods to compute option prices. We also provide a systematic way of choosing the parameters of the numerical method, in a way that minimises the strict error bound, thus guaranteeing adherence to a pre-described error tolerance. We focus on exponential Lévy processes that may be of either diffusive or pure jump. Our contribution is to derive a strict error bound for a Fourier transform method when pricing options under risk-neutral Lévy dynamics. We derive a simplified bound that separates the contributions of the payoff and of the process in an easily processed and extensible product form that is independent of the asymptotic behaviour of the option price at extreme prices and at strike parameters. We also provide a proof for the existence of optimal parameters of the numerical computation that minimise the presented error bound. When comparing our work with Lee’s work we find that Lee’s work is more general than ours in that he studies a wider range of processes, on the other hand, our results apply to a larger class of payoffs. On test examples of practical relevance, we also find that the bound presented produces comparable or better results than the ones previously presented in the literature, with acceptable computational cost.

The paper is organised in the following sections: In Section 7.2 we introduce the PIDE setting in the context of risk-neutral asset pricing; we show the Fourier representation of the relevant PIDE for asset pricing with Lévy processes and use that representation for derivative pricing. In Section 7.3 we derive a representation for the numerical error and divide it into quadrature and cutoff contributions. We also describe the methodology for choosing numerical parameters to obtain minimal error bounds for the FT method. The derivation is supported by numerical examples using relevant test cases with both diffusive and pure-jump Lévy processes in Section
7.2 Fourier method for option pricing

Consider an asset whose price at time $t$ is modelled by the stochastic process $S = (S_t)$ defined by $S_t = S_0 e^{X_t}$, where $X = (X_t) \in \mathbb{R}$ is assumed to be a Lévy process whose jump measure $\nu$ satisfies

$$\int_{\mathbb{R} - \{0\}} \min\{y^2, 1\} \nu(dy) < \infty\quad (7.1)$$

Assuming the risk-neutral dynamic for $S_t$, the price at time $t = T - \tau$ of a European option with payoff $G$ and maturity time $T$ is given by

$$\Pi(\tau, s) = e^{-r\tau} \mathbb{E}[G(S_T) | S_{T-\tau} = s]$$

where $r$ is the short rate that we assume to be constant and $\tau: 0 \leq \tau \leq T$ is the time to maturity. Extensions to non-constant deterministic short rates are straightforward.

The infinitesimal generator of a Lévy process $X$ is given by

$$\mathcal{L}^X f(x) \equiv \lim_{h \to 0} \frac{\mathbb{E}[f(X_{t+h}) | X_t = x] - f(x)}{h} = \gamma f'(x) + \frac{1}{2} \sigma^2 f''(x) + \int_{\mathbb{R} - \{0\}} \left( f(x + y) - f(x) - y 1_{|y| \leq 1} f'(x) \right) \nu(dy) \quad (7.2)$$

where $(\gamma, \sigma^2, \nu)$ is the characteristic triple of the Lévy process. The risk-neutral assumption on $(S_t)$ implies

$$\int_{|y| > 1} e^y \nu(dy) < \infty\quad (7.3)$$

and fixes the drift term (see [21]) $\gamma$ of the Lévy process to

$$\gamma = r - \frac{1}{2} \sigma^2 - \int_{\mathbb{R} - \{0\}} \left( e^y - 1 - y 1_{|y| \leq 1} \right) \nu(dy) \quad (7.4)$$
Thus, the infinitesimal generator of $X$ may be written under the risk-neutral assumption as

$$
\mathcal{L}^X f(x) = \left( r - \frac{\sigma^2}{2} \right) f'(x) + \frac{\sigma^2}{2} f''(x) + \int_{\mathbb{R} - \{0\}} (f(x + y) - f(x) - (e^y - 1)f'(x)) \nu(dy)
$$

(7.5)

Consider $g$ as the reward function in log prices (ie, defined by $g(x) = G(S_0 e^x)$). Now, take $f$ to be defined as

$$
f(\tau, x) \equiv \mathbb{E}[g(X_T) | X_{T-\tau} = x]
$$

Then $f$ solves the following PIDE:

$$
\begin{cases}
\partial_\tau f(\tau, x) = \mathcal{L}^X f(\tau, x) \\
f(0, x) = g(x), \quad (\tau, x) \in [0, T] \times \mathbb{R}
\end{cases}
$$

Observe that $f$ and $\Pi$ are related by

$$
\Pi(\tau, S_0 e^x) = e^{-\tau r} f(\tau, x)
$$

(7.6)

Consider a damped version of $f$ defined by $f_\alpha(\tau, x) = e^{-\alpha \tau} f(\tau, x)$; we see that $\partial_\tau f_\alpha = e^{-\alpha \tau} \mathcal{L}^X f(\tau, x)$.

There are different conventions for the Fourier transform. Here we consider the operator $\mathcal{F}$ such that

$$
\mathcal{F}[f](\omega) \equiv \int_{\mathbb{R}} e^{i\omega x} f(x) \, dx
$$

(7.7)

defined for functions $f$ for which the previous integral is convergent. We also use $\hat{f}(\omega)$ as a shorthand notation of $\mathcal{F}[f](\omega)$. To recover the original function $f$, we
define the inverse Fourier transform as

\[ \mathcal{F}^{-1} [f] (x) = \frac{1}{2\pi} \int_{\mathbb{R}} e^{-i\omega x} f(\omega) d\omega \]

We have that \( \mathcal{F}^{-1} [\hat{f}] (x) = f(x) \).

Applying \( \mathcal{F} \) to \( f_\alpha \) we get \( \hat{f}_\alpha (\omega) = \hat{f} (\omega + i\alpha) \). Observe also that the Fourier transform applied to \( L^X f(\tau, x) \) gives \( \Psi (-i\omega) \hat{f} (\tau, \omega) \), where \( \Psi (\cdot) \) is the characteristic exponent of the process \( X \), which satisfies \( \mathbb{E} [e^{zX_\tau}] = e^{\psi(z)} \). The explicit expression for \( \Psi (\cdot) \) is

\[ \Psi (z) = \left( r - \frac{\sigma^2}{2} \right) z + \frac{\sigma^2}{2} z^2 + \int_{\mathbb{R}} (e^{zy} - 1 - (e^y - 1)z) \nu(dy) \]  

(7.8)

From the previous considerations it can be concluded that

\[ \partial_\tau \hat{f}_\alpha = \Psi (\alpha - i\omega) \hat{f} (\omega - i\alpha) \]  

(7.9)

Now \( \hat{f} (\omega - i\alpha) = \hat{f}_\alpha (\omega) \) so \( \hat{f}_\alpha \) satisfies the following ODE

\[
\begin{cases}
\partial_\tau \frac{f_\alpha (\tau, \omega)}{f_\alpha (\tau, \omega)} = \Psi (\alpha - i\omega) \\
\hat{f}_\alpha (0, \omega) = \hat{g}_\alpha (\omega)
\end{cases}
\]

(7.10)

Solving the previous ODE explicitly, we obtain

\[ \hat{f}_\alpha (\tau, \omega) = e^{\tau \psi(\alpha - i\omega)} \hat{g}_\alpha (\omega) \]  

(7.11)

Observe that the first factor in the right-hand side in the above equation is \( \mathbb{E} [e^{(\alpha - i\omega)X_\tau}] \), (ie, \( \varphi_1 (-i\alpha - \omega) \)), where \( \varphi_\tau (\cdot) \) denotes the characteristic function of the random vari-
able $X_{\tau}$

\[ \varphi_{\tau}(\omega) \equiv \mathbb{E}[\tau \Psi(i\omega)] \quad (7.12) \]

Now, to obtain the value function we employ the inverse Fourier transformation, to obtain

\[ f_{\alpha}(\tau, x) = \mathcal{F}^{-1}\left[ \hat{f}_{\alpha} \right](\tau, x) = \frac{1}{2\pi} \int_{\mathbb{R}} e^{-i\omega x} \hat{f}_{\alpha}(\tau, \omega) d\omega \quad (7.13) \]

or

\[ f_{\alpha}(\tau, x) = \frac{\Delta \omega}{\pi} \sum_{k=-\infty}^{+\infty} \text{Re} \left[ e^{-i(k+\frac{1}{2})\Delta \omega x} \hat{f}_{\alpha}(\tau, \left(k + \frac{1}{2}\right) \Delta \omega) \right] d\omega \quad (7.14) \]

As it is typically not possible to compute the inverse Fourier transform analytically, we approximate it by discretising and truncating the integration domain using trapezoidal quadrature \((7.13)\). Consider the following approximation:

\[ f_{\alpha,\Delta \omega,n}(\tau, x) = \frac{\Delta \omega}{2\pi} \sum_{k=-n}^{n-1} e^{-i(k+\frac{1}{2})\Delta \omega x} \hat{f}_{\alpha}(\tau, \left(k + \frac{1}{2}\right) \Delta \omega) \quad (7.15) \]

\[ = \frac{\Delta \omega}{\pi} \sum_{k=0}^{n-1} \text{Re} \left[ e^{-i(k+\frac{1}{2})\Delta \omega x} \hat{f}_{\alpha}(\tau, \left(k + \frac{1}{2}\right) \Delta \omega) \right] \quad (7.16) \]

Bounding and consequently minimising the error in the approximation of $f(\tau, x)$ by

\[ f_{\Delta \omega,n}(\tau, x) \equiv e^{\alpha x} f_{\alpha,\Delta \omega,n}(\tau, x) \]

is the main focus of this paper and will be addressed in the following section.

**Remark 7.2.1.** Although we are mainly concerned with option pricing when the payoff function can be damped in order to guarantee regularity in the $L^1$ sense, we note here that our main results are naturally extendable to include the Greeks of the option.
Indeed, we have by (7.11) that

\[ f(t, x) = \frac{1}{2\pi} \int_{\mathbb{R}} e^{(\alpha - i\omega)x} \hat{f}_\alpha(\tau, \omega) \, d\omega \]  

(7.17)

so the Delta and Gamma of the option equal

\[ \Delta (t, x) \equiv \frac{\partial f(t, x)}{\partial x} = \frac{1}{2\pi} \int_{\mathbb{R}} (\alpha - i\omega) e^{(\alpha - i\omega)x} \hat{f}_\alpha(\tau, \omega) \, d\omega \]  

(7.18)

\[ \Gamma (t, x) \equiv \frac{\partial^2 f(t, x)}{\partial x^2} = \frac{1}{2\pi} \int_{\mathbb{R}} (\alpha - i\omega)^2 e^{(\alpha - i\omega)x} \hat{f}_\alpha(\tau, \omega) \, d\omega \]  

(7.19)

Because the expressions involve partial derivatives with respect to only \( x \), the results in this work are applicable for the computation of \( \Delta \) and \( \Gamma \) through a modification of the payoff function:

\[ \hat{g}_{\alpha, \Delta}(\omega) = \hat{g}_\alpha(\omega) (\alpha - i\omega) \]  

(7.20)

\[ \hat{g}_{\alpha, \Gamma}(\omega) = \hat{g}_\alpha(\omega) (\alpha - i\omega)^2 \]  

(7.21)

When the Fourier space payoff function manifests exponential decay, the introduction of a coefficient that is polynomial in \( \omega \) does not change the regularity of \( \hat{g} \) in a way that would significantly change the following analysis. Last, we note that since we do our analysis for PIDEs on a mesh of \( x \)'s, one may also compute the option values in one go and obtain the Greeks with little additional effort using a finite difference approach for the derivatives.

7.2.1 Evaluation of the method for multiple values of \( x \) simultaneously

The Fast Fourier Transform (FFT) algorithm provides an efficient way of computing (7.15) for an equidistantly spaced mesh of values for \( x \) simultaneously. Examples of works that consider this widely extended tool are [19, 20, 25] and 34.
Similarly, one may define the Fourier frequency $\omega$ as the conjugate variable of some external parameter on which the payoff depends. Especially, for the practically relevant case of call options, we can denote the log-strike as $k$ and treat $x$ as a constant and write:

$$\tilde{f}_{k,\alpha}(\omega) \equiv \int_{\mathbb{R}} e^{(\alpha + i\omega)k} f_k(x) \, dk$$

(7.22)

Using this convention, the time dependence is given by

$$\tilde{f}_{k,\alpha}(\tau, x) = e^{(i\omega + \alpha + 1)x} \varphi_{\tau}(\omega - i(\alpha + 1)) \left( i\omega + \alpha \right) \left( i\omega + \alpha + 1 \right)$$

(7.23)

contrasted with the $x$-space solution

$$\hat{f}_{k,\alpha}(\tau, x) = e^{(i\omega - \alpha + 1)k} \varphi_{x}(\omega + i\alpha) \left( i\omega + \alpha \right) \left( i\omega + \alpha + 1 \right)$$

(7.24)

We note that for call option payoff to be in $L^1$, we demand that $\alpha$ in (7.23) is positive. Omitting the exponential factors that contain the $x$ and $k$ dependence in (7.23) and (7.24) respectively, we have that one can arrive from (7.23) to (7.24) using the mapping $\alpha \mapsto -\alpha - 1$. Thanks to this, much of the analysis regarding the $x$-space transformation generalises in a straightforward manner to the $k$-space transform.

### 7.3 Error bound

The aim of this section is to compute a bound of the error when approximating the option price $f(\tau, x)$ by $f_{\alpha, \Delta\omega, n}(\tau, x)$, defined in (7.15). Considering

$$f_{\alpha, \Delta\omega}(\tau, x) = \frac{\Delta\omega}{2\pi} \sum_{k \in \mathbb{Z}} e^{-i(k + \frac{1}{2})\Delta\omega x} \hat{f}_{\alpha} \left( \tau, \left( k + \frac{1}{2} \right) \Delta\omega \right)$$

(7.25)
the total error $\mathcal{E}$ can be split into a sum of two terms: the quadrature and truncation errors. The former is the error from the approximation of the integral in (7.13) by the infinite sum in (7.25), while the latter is due to the truncation of the infinite sum. Using triangle inequality, we have

$$\mathcal{E} := |f(\tau, x) - f_{\Delta \omega, n}(\tau, x)| \leq \mathcal{E}_Q + \mathcal{E}_F$$

(7.26)

with

$$\mathcal{E}_Q = e^{\alpha x} |f_{\alpha}(\tau, x) - f_{\alpha, \Delta \omega}(\tau, x)|$$

$$\mathcal{E}_F = e^{\alpha x} |f_{\alpha, \Delta \omega}(\tau, x) - f_{\alpha, \Delta \omega, n}(\tau, x)|$$

Observe that each $\mathcal{E}$, $\mathcal{E}_Q$ and $\mathcal{E}_F$ depend on three kinds of parameters:

- Parameters underlying the model and payoff such as volatility and strike price. We call these physical parameters.
- Parameters relating to the numerical scheme such as $\alpha$ and $n$.
- Auxiliary parameters that will be introduced in the process of deriving the error bound. These parameters do not enter the computation of the option price, but they need to be chosen appropriately to have as tight a bound as possible.

We start by analysing the quadrature error.

### 7.3.1 Quadrature error

Denote by $A_a$, with $a > 0$, the strip of width $2a$ around the real line:

$$A_a \equiv \{ z \in \mathbb{C} : |\text{Im}[z]| < a \}$$
The following theorem presents conditions under which the quadrature error goes to zero at a spectral rate as $\Delta \omega$ goes to zero. Later in this section, we discuss simpler conditions to verify the hypotheses and analyse in more detail the case when the process $X$ is a diffusive process or there are “enough small jumps.”

**Theorem 7.3.1.** Assume that for $a > 0$:

1. the characteristic function of the random variable $X_1$ has an analytic extension to the set
   
   $$A_a - \alpha i \equiv \{z \in \mathbb{C}: |\text{Im } z + \alpha| < a\}$$

2. the Fourier transform of $g_\alpha(x)$ is analytic in the strip $A_a$ and

3. there exists a continuous function $\gamma \in L^1(\mathbb{R})$ such that
   
   $$|\hat{f}_\alpha(\tau, \omega + i\beta)| < \gamma(\omega)$$
   
   for all $\omega \in \mathbb{R}$ and for all $\beta \in [-a, a]$

Then the quadrature error is bounded by

$$\mathcal{E}_Q \leq e^{\alpha x} \frac{M_{\alpha,a}(\tau, x)}{2\pi (e^{2\pi a/\Delta \omega} - 1)}$$

where $M_{\alpha,a}(\tau, x)$ is given by

$$M_{\alpha,a}(\tau, x) := \sum_{\beta \in \{-a, a\}} \int_{\mathbb{R}} |e^{-i(\omega + i\beta)x} \hat{f}_\alpha(\tau, \omega + i\beta)| \, d\omega \quad (7.27)$$

$M_{\alpha,a}(\tau, x)$ equals the Hardy norm (defined in (7.28)) of the function $\omega \mapsto e^{-i(\omega + i\beta)x} \hat{f}_\alpha(\tau, \omega + i\beta)$, which is finite.

The proof of Theorem 7.3.1 is an application of Theorem 3.2.1 in [35], whose relevant parts we include for ease of reading. Using the notation in [35], $H^1_{A_a}$ is the family of functions $w$ that are analytic in $A_a$ and such that

$$||w||_{H^1_{A_a}} := \lim_{\varepsilon \to 0} \int_{\partial A_a(\varepsilon)} |w(z)| \, |dz| < \infty \quad (7.28)$$
where
\[ A_a(\varepsilon) = \left\{ z \in \mathbb{C} : |\text{Re}[z]| < \frac{1}{\varepsilon}, |\text{Im}[z]| < a(1 - \varepsilon) \right\} \]

**Lemma 7.3.2** (Theorem 3.2.1 in [35]). Let \( w \in H^1_{A_a} \), then define:

\[
I(w) = \int_{\mathbb{R}} w(x) \, dx \tag{7.29}
\]

\[
J(w, h) = h \sum_{j=-N}^{N} w(jh) \tag{7.30}
\]

\[
\zeta(w, h) = I(w) - J(w, h) \tag{7.31}
\]

then

\[
|\zeta(w, h)| \leq \frac{e^{-\frac{\pi}{2h}} ||w||_{H^1_{A_a}}}{2\sinh\left(\frac{\pi a}{h}\right)} \tag{7.32}
\]

**Proof of Theorem 7.3.1.** First observe that H1 and H2 imply that the function \( w(z) = e^{-ixz+\Delta\omega/2} \hat{f}_\alpha(\tau, z + \Delta\omega/2) \) is analytic in \( A_a \). H3 allows us to use dominated convergence theorem to prove that \( ||w||_{H^1_{A_a}} \) is finite and coincides with \( M_{\alpha,a}(\tau, x) \). Applying Lemma 7.3.2 the proof is completed. \( \square \)

Regarding the hypotheses of Theorem 7.3.1, the next propositions provide simpler conditions that imply H1 and H2 respectively.

**Proposition 7.3.3.** If \( \alpha, a \) and \( \nu \) are such that

\[
\int_{y>1} e^{(a-a)y} \nu(dy) < \infty \quad \text{and} \quad \int_{y<-1} e^{(a-a)y} \nu(dy) < \infty \tag{7.33}
\]

then H1 in Theorem 7.3.1 is fulfilled.

**Proof.** Denoting by \( \varphi_1(\cdot) \) the characteristic function of \( X_1 \), we want to prove that \( z \mapsto \varphi_1(z + \alpha i) \) is analytic in \( A_a \). Considering that \( \varphi_1(z + \alpha i) = e^{\psi(iz - \alpha)} \), the only
non-trivial part of the proof is to verify that

\[ z \mapsto \int p(z, y) \nu(dy) \]  \hspace{1cm} (7.34)

is analytic in \( A_a \), where \( p: A_a \times \mathbb{R} \to \mathbb{C} \) is given by

\[ p(z, y) = e^{yz - \alpha} - 1 - (e^y - 1)(iz - \alpha) \]

To prove this fact, we show that we can apply the main result and the only theorem in [28], which, given a measure space \( (\Omega, \mathcal{A}, \mu) \) and an open subset \( G \subseteq \mathbb{C} \), ensures the analyticity of \( \int f(\cdot, \omega)d\mu(\omega) \), provided that \( f: G \times \Omega \to \mathbb{C} \) satisfies: \( f(z, \cdot) \) is \( \mathcal{A} \)-measurable for all \( z \in G \); \( f(\cdot, \omega) \) is holomorphic for all \( \omega \in \Omega \); and \( \int |f(\cdot, \omega)|d\mu(\omega) \) is locally bounded. In our case we consider the measure space to be \( \mathbb{R} \) with the Borel \( \sigma \)-algebra and the Lebesgue measure, \( G = A_a \) and \( f = p \). It is clear that \( p(x, \cdot) \) is Borel measurable and \( p(\cdot, y) \) is holomorphic. It remains to verify that

\[ z \mapsto \int_{\mathbb{R}^+} |p(z, y)| \nu(dy) \]

is locally bounded. To this end, we assume that \( \text{Re}[z] < b \) (and, since \( z \in A_a \), \( \text{Im}[z] < a \)) and split the integration domain in \( |y| > 1 \) and \( 0 < |y| \leq 1 \) to prove that both integrals are uniformly bounded.

Regarding the integral in \( |y| > 1 \), we observe that

\[ |p(z, y)| \leq e^{y(\alpha + \text{Im}[z])} + 1 + (e^y + 1)(\alpha + a + b) \]  \hspace{1cm} (7.35)

for \( y < -1 \) we have \( e^{y(\alpha + \text{Im}[z])} < e^{y(\alpha - a)} \) while for \( y > 1 \) we have \( e^{y(\alpha + \text{Im}[z])} < e^{y(\alpha + a)} \). Using the previous bounds and the hypotheses together with (7.1) and (7.3), we obtain the needed bound.
For the integral in $0 < |y| \leq 1$, observe that, denoting $f(z, y) = |p(z, y)|$, we have, $f(z, 0) = 0$ for every $z$, $\partial_y f(z, 0) = 0$ for every $z$, and $|\partial_{yy} f(z, y)| < c$ for $z \in A_a, \Re[z] < b, |y| < 1$. From these observations we get that the McLaurin polynomial of degree one of $y \mapsto f(z, y)$ is null for every $z$, and we can bound $f(z, y)$ by the remainder term, which, in our region of interest, is bounded by $\frac{c}{2} y^2$, obtaining

$$\int_{0 < |y| \leq 1} |p(z, y)| \nu(dy) \leq \frac{c}{2} \int_{0 < |y| \leq 1} y^2 \nu(dy)$$

(7.36)

which is finite by hypothesis on $\nu$, which finishes the proof.

Proposition 7.3.4. If for all $b < a$, the function $x \mapsto e^{b|x|} g_a(x)$ is in $L^2(\mathbb{R})$ then H2 in Theorem 7.3.1 is fulfilled.

Proof. The proof is a direct application of Theorem IX.13 in [32]

We now turn our attention to a more restricted class of Levy processes. Namely, processes such that either $\sigma^2 > 0$ or there exists $\lambda \in (0, 2)$ such that $C(\lambda)$ defined in (7.37) is strictly positive. For this class of processes, we can state our main result explicitly in terms of the characteristic triplet.

Given $\lambda \in (0, 2)$, define $C(\lambda)$ as

$$C(\lambda) = \inf_{\kappa > 1} \left\{ \kappa^\lambda \int_{0 < |y| < \frac{\kappa}{2}} y^2 \nu(dy) \right\} \quad (7.37)$$

Observe that $C(\lambda) \geq 0$ and, by our assumptions on the jump measure $\nu$, $C(\lambda)$ is finite. Furthermore, if $\lambda \in (0, 2)$ is such that

$$\liminf_{\epsilon \downarrow 0} \frac{1}{\epsilon^\lambda} \int_{0 < |y| < \epsilon} y^2 \nu(dy) > 0$$

(7.38)
then $C(\lambda) > 0$. To see this, observe that (7.38) implies the existence of $\epsilon_0$ such that

$$\inf_{\epsilon \leq \epsilon_0} \left\{ \frac{1}{\epsilon^\lambda} \int_{0<|y|<\epsilon} y^2 \nu(dy) \right\} > 0$$

If $\epsilon_0 < 1$ observe that

$$\inf_{\epsilon_0 \leq \epsilon \leq 1} \left\{ \frac{1}{\epsilon^\lambda} \int_{0<|y|<\epsilon} y^2 \nu(dy) \right\} \geq \int_{0<|y|<\epsilon_0} y^2 \nu(dy) > 0$$

where for the first inequality it was taken into account that $\frac{1}{\epsilon^\lambda} \geq 1$ and that the integral is increasing with $\epsilon$. Combining the two previous infima and considering $|\kappa| = \frac{1}{\epsilon}$ we get that $C(\lambda) > 0$.

Furthermore, we note that for a Lévy model with finite jump intensity, such as the Black-Scholes and Merton models that satisfy the first of our assumption, $C(\lambda) = 0$ for all $\lambda \in (0, 2)$.

**Theorem 7.3.5.** Assume that: $\alpha$ and $a$ are such that (7.33) holds; $\hat{g}_\alpha \in L^\infty_{\mathcal{A}_a}$; and either $\sigma^2 > 0$ or $C(\lambda) > 0$ for some $\lambda \in (0, 2)$. Then the quadrature error is bounded by

$$\mathcal{E}_Q \leq e^{\alpha x} \frac{\tilde{M}_{\alpha,a}(\tau, x)}{2\pi (e^{2\pi a/\Delta \omega} - 1)}$$

where

$$\tilde{M}_{\alpha,a}(\tau, x) = \sum_{c \in \{-1, 1\}} e^{\alpha x} e^{\tau \Psi(\alpha)} |\hat{g}_{\alpha}(ca)| \int_{\mathbb{R}} e^{-\tau\left( \frac{c^2}{2} \omega^2 + \frac{|\omega|^{2-\lambda}}{4} C(\lambda) 1_{|\omega|>1} \right)} d\omega \quad (7.39)$$

Furthermore, if $\sigma^2 > 0$ we have

$$\tilde{M}_{\alpha,a}(\tau, x) \leq \frac{\sqrt{2\pi}}{\sigma \sqrt{\tau}} \sum_{c \in \{-1, 1\}} e^{\alpha x} e^{\tau \Psi(\alpha)} |\hat{g}_{\alpha}(ca)| \quad (7.40)$$
Proof. Considering \( h_{\alpha,a}(\tau, x, \omega) \) defined by

\[
h_{\alpha,a}(\tau, x, \omega) = \sum_{c \in \{-1,1\}} |e^{-i(\omega+i\alpha)x} \hat{f}_\alpha(\tau, \omega + i\alpha)|
\]

we have that

\[
M_{\alpha,a}(\tau, x) = \int_{\mathbb{R}} h_{\alpha,a}(\tau, x, \omega) \, d\omega
\]

On the other hand, for \( \beta \in (-a, a) \):

\[
|e^{-i(\omega+i\beta)x} \hat{f}_\alpha(\tau, \omega + i\beta)| = e^{\beta x} |\hat{f}_\alpha(\tau, \omega + i\beta)| = e^{\beta x} \left| e^{\tau \Psi(\alpha+\beta-i\omega)} \right| |\hat{g}_\alpha(\omega + i\beta)|
\]

For the factor involving the characteristic exponent we have

\[
|e^{\tau \Psi(\alpha+\beta-i\omega)}| = e^{\tau \text{Re}[\Psi(\alpha+\beta-i\omega)]}
\]

Now, observe that

\[
\text{Re} [\Psi (\alpha + \beta - i\omega)] = (\alpha + \beta) \left( r - \frac{\sigma^2}{2} \right) + \frac{\sigma^2}{2} \left( (\alpha + \beta)^2 - \omega^2 \right) + \int_{\mathbb{R} \setminus \{0\}} (e^{(\alpha+\beta)y} \cos(-y\omega) - 1 - (\alpha + \beta) (e^y - 1)) \nu(dy)
\]

If \(|\omega| \leq 1\) we bound \(\cos(-y\omega)\) by 1, getting

\[
\text{Re} [\Psi (\alpha + \beta - i\omega)] \leq (\alpha + \beta) \left( r - \frac{\sigma^2}{2} \right) + \frac{\sigma^2}{2} \left( (\alpha + \beta)^2 - \omega^2 \right) + \int_{\mathbb{R} \setminus \{0\}} (e^{(\alpha+\beta)y} - 1 - (\alpha + \beta) (e^y - 1)) \nu(dy)
\]

Assume \(|\omega| > 1\). Using that for \(|x| < 1\) it holds that \(\cos(x) < 1 - x^2/4\), we can
bound the first term of the integral in the following manner:

\[
\int_{\mathbb{R} - \{0\}} e^{(\alpha + \beta)y} \cos(y\omega) \nu(dy) \leq \int_{0<|y|<1/|\omega|} e^{(\alpha + \beta)y} \left(1 - \omega^2 y^2/4\right) \nu(dy) \\
+ \int_{|y|\geq1/|\omega|} e^{(\alpha + \beta)y} \nu(dy) \\
\leq \int_{\mathbb{R} - \{0\}} e^{(\alpha + \beta)y} \nu(dy) - \frac{|\omega|^{2-\lambda} \nu(dy)}{4} \\
\leq \int_{\mathbb{R} - \{0\}} e^{(\alpha + \beta)y} \nu(dy) - \frac{|\omega|^{2-\lambda}}{4} C(\lambda) 
\]

(7.46)

Inserting (7.46) back into (7.44) we get

\[
\text{Re} \left[ \Psi (\alpha + \beta - i\omega) \right] \leq (\alpha + \beta) \left( r - \frac{\sigma^2}{2} \right) + \frac{\sigma^2}{2} \left( (\alpha + \beta)^2 - \omega^2 \right) \\
+ \int_{\mathbb{R} - \{0\}} (e^{(\alpha + \beta)y} - 1 - (\alpha + \beta) (e^y - 1)) \nu(dy) \\
- \frac{|\omega|^{2-\lambda} C(\lambda)}{4} \\
= \Psi (\alpha + \beta) - \frac{\sigma^2}{2} \omega^2 - \frac{|\omega|^{2-\lambda} C(\lambda)}{4} 
\]

Taking the previous considerations and integrating in \( \mathbb{R} \) with respect to \( \omega \), we obtain (7.39).

Finally, observing that \( C(\lambda) \geq 0 \) and bounding it by 0, the bound (7.40) is obtained by evaluating the integral.

\[\square\]

**Remark 7.3.6.** In the case of call options, hypothesis H2 implies a dependence between the strip-width parameter \( a \) and damping parameter \( \alpha \). We have that the damped payoff of the call option is in \( L^1(\mathbb{R}) \) if and only if \( \alpha > 1 \) and hence the appropriate choice of strip-width parameter is given by \( 0 < a < \alpha - 1 \). A similar argument holds for the case of put options, for which the Fourier-transformed damped payoff is identical to the calls with the distinction that \( \alpha < 0 \). In such case, we require \( a < -\alpha \).

The case of binary options whose payoff has finite support \( (G(x) = 1_{[x_-,x_+]}(x)) \)
we can set any $a \in \mathbb{R}$ (ie, no damping is needed at all and even if such damping is chosen, it has no effect on the appropriate choice of $a$).

Remark 7.3.7. The bound we provide for the quadrature error is naturally positive and increasing in $\Delta \omega$. It decays to zero at a spectral rate as $\Delta \omega$ decreases to 0.

7.3.2 Frequency truncation error

The frequency truncation error is given by

$$
\mathcal{E}_F = \frac{e^{\alpha x} \Delta \omega}{\pi} \sum_{k=n}^{\infty} \left| \text{Re} \left[ e^{-i(k+\frac{1}{2})\Delta \omega} \hat{f}_\alpha \left( \tau, \left( k + \frac{1}{2} \right) \Delta \omega \right) \right] \right|
$$

If a function $c: (\omega_0, \infty) \to (0, \infty)$ satisfies

$$
\left| \text{Re} \left[ e^{-i(k+\frac{1}{2})\Delta \omega} \hat{f}_\alpha \left( \tau, \left( k + \frac{1}{2} \right) \Delta \omega \right) \right] \right| \leq c \left( \left( k + \frac{1}{2} \right) \Delta \omega \right)
$$

for every natural number $k$, then we have that

$$
\mathcal{E}_F \leq \frac{e^{\alpha x} \Delta \omega}{\pi} \sum_{k=n}^{\infty} \left| \text{Re} \left[ e^{-i(k+\frac{1}{2})\Delta \omega} \hat{f}_\alpha \left( \tau, \left( k + \frac{1}{2} \right) \Delta \omega \right) \right] \right| \\
\leq \frac{e^{\alpha x} \Delta \omega}{\pi} \sum_{k=n}^{\infty} c \left( \left( k + \frac{1}{2} \right) \Delta \omega \right)
$$

Furthermore, if $c$ is a non-increasing concave integrable function, we get

$$
\mathcal{E}_F \leq \frac{e^{\alpha x}}{\pi} \int_{n\Delta \omega}^{\infty} c(\omega) d\omega
$$

(7.48)

When $\hat{g}_\alpha \in L^\infty_{[\omega_0, \infty)}$ and either $\sigma^2 > 0$ or $C(\lambda) > 0$, then the function $c$ in (7.47) can be chosen as

$$
c(\omega) = \| \hat{g}_\alpha \|_{L^\infty_{[\omega_0, \infty)}} e^{\tau \Psi(\alpha)} e^{-\tau \left( \frac{\omega^2 + |\omega|^2 - \lambda}{4} C(\lambda) \right) 1_{|\omega| > 1}}
$$

(7.49)

To prove that this function satisfies (7.47) we can use the same bound we found
in the proof of Theorem \[ \text{7.3.5} \] with \( \beta = 0 \), to obtain

\[
\text{Re} \left[ \Psi (\alpha - i\omega) \right] \leq \Psi (\alpha) - \frac{\sigma^2}{2} \omega^2 - \frac{|\omega|^{2-\lambda}}{4} C (\lambda) \mathbf{1}_{|\omega|>1}
\]

from where the result is straightforward.

### 7.3.3 Bound for the full error

In this section we summarize the bounds obtained for the error under different assumptions and analyse their central properties.

In general the bound provided in this paper are of the form

\[
\bar{E} = \frac{e^{\alpha x}}{\pi} \left( \frac{\bar{M}}{e^{2\pi a/\Delta \omega} - 1} + \int_{n\Delta \omega}^{\infty} c (\omega) \, d\omega \right) \tag{7.50}
\]

where \( \bar{M} \) is an upper bound of \( M_{\alpha,a} (\tau,x) \) defined in \( \text{7.27} \) and \( c \) is non-increasing, integrable and satisfies \( \text{7.47} \). Both \( \bar{M} \) and \( c \) may depend on the parameters of the model and the artificial parameters, but they are independent of \( \Delta \omega \) and \( n \). Typically one can remove the dependence of some of the parameters, simplifying the expressions but obtaining less tight bounds.

When analysing the behaviour of the bound one can observe that the term corresponding with the quadrature error decreases to zero spectrally when \( \Delta \omega \) goes to 0. The second term goes to zero if \( n\Delta \omega \) diverges, but we are unable to determine the rate of convergence without further assumptions.

Once an expression for the error bound is obtained, the problem of how to choose the parameters of the numerical method to minimise the bound arises, assuming a constraint on the computational effort one is willing to use. The computational effort of the numerical method depends only on \( n \). For this reason we aim at finding the parameters that minimise the bound for a fixed \( n \). The following result shows that the bound obtained, as a function of \( \Delta \omega \), has a unique local minimum, which is the
global minimum.

**Proposition 7.3.8.** Fix $\alpha$, $a$, $n$, and $\lambda$ and consider the bound $\bar{E}$ as a function of $\Delta \omega$. There exists an optimal $\Delta \omega^* \in \left[ \frac{\omega_0}{n}, \infty \right)$ such that $\bar{E}$ is decreasing in $\left( \frac{\omega_0}{n}, \Delta \omega^* \right)$ and increasing in $\left( \Delta \omega^*, \infty \right)$; thus, a global minimum of $\bar{E}$ is attained at $\Delta \omega^*$.

Furthermore, the optimal $\Delta \omega$ is either the only point in which $\Delta \omega \mapsto p(n \Delta \omega, b) - c(n \Delta \omega)$, with $p$ defined in (7.51), changes sign, or $\Delta \omega = \frac{\omega_0}{n}$ if $p(\omega_0, b) - c(\omega_0) > 0$.

**Proof.** Let us simplify the notation by calling $y = n \Delta \omega$, $b = 2\pi an$ and $\tilde{E} = \pi e^{-\alpha x} E$.

We want to prove the existence of $y^*$: $y^* \geq \omega_0$ such that $\tilde{E}(y)$ is decreasing for $\omega_0 < y < y^*$ and increasing for $y > y^*$. We have

$$\tilde{E}(y) = \frac{\bar{M}}{e^{b/y} - 1} + \int_y^\infty c(\omega) d\omega.$$  

The first term is differentiable with respect to $y$ and goes to 0 if $y \to 0^+$. This allows us to express it as an integral of its derivative. We can then express $\tilde{E}(y)$ as

$$\tilde{E}(y) = \tilde{E}(\omega_0) + \int_{\omega_0}^y \left( \frac{b \bar{M} e^{b/\omega}}{(e^{b/\omega} - 1)^2 \omega^2} - c(\omega) \right) d(\omega)$$

The first term on the right-hand side of the previous equation is constant. Now we move on to proving that the integrand is increasing with $y$ and it is positive if $y$ is large enough. Denote by

$$p(y, b) = \frac{b \bar{M} e^{b/y}}{(e^{b/y} - 1)^2 y^2}$$

(7.51)

Taking into account that $c$ is integrable, we can compute the limit of the integrand in $\infty$, obtaining

$$\lim_{y \to +\infty} p(y, b) - c(y) = \frac{\bar{M}}{b} > 0$$

Let us prove that $p(y, b)$ is increasing with $y$ for all $b > 0$, which renders $p(y, b) - c(y)$
also increasing with \(y\). The derivative of \(p\) with respect to \(y\) is given by
\[
\frac{\partial_y p(y, b)}{e^{b/y} - 1} = \frac{b \bar{M} e^{b/y} ((b/y) e^{b/y} - 2 e^{b/y} + b/y + 2)}{y^3 (e^{b/y} - 1)^3}
\]
in which the denominator and the first factor in the numerator are clearly positive. To prove that the remainder factor is also positive, observe that \(x e^x - 2 e^x + x + 2 > 0\) if \(x > 0\).

\[
\text{(7.52)}
\]

\[\bar{E} = \bar{E}_Q + \bar{E}_F\]

where
\[
\bar{E}_Q = \sum_{c \in \{-1, 1\}} e^{\alpha x} e^{\alpha x} e^{\tau \Psi(\alpha)} \left| \hat{g}_\alpha (ca) \right| \int_{\mathbb{R}} e^{-\tau \left( \frac{e^{2 \omega^2} + |\omega|^{2-\lambda}}{4} \right) C(\lambda) 1_{|\omega| > 1}} d\omega
\]

\[\text{(7.53)}\]

\[
\bar{E}_F = \frac{e^{\alpha x}}{\pi} \left\| \hat{g}_\alpha \right\|_{L^\infty_{\mathbb{R}}} e^{\tau \Psi(\alpha)} \int_{n\Delta \omega}^{\infty} e^{-\tau \left( \frac{e^{2 \omega^2} + |\omega|^{2-\lambda}}{4} \right) C(\lambda) 1_{|\omega| > 1}} d\omega
\]

\[\text{(7.54)}\]

This reproduces the essential features of Theorem 6.6 in [18]. the bound (7.54) can be further improved by substituting \(\left\| \hat{g}_\alpha \right\|_{L^\infty_{\mathbb{R}}}\) by \(\left\| \hat{g}_\alpha \right\|_{L^\infty_{[n\Delta \omega, \infty)}}\).

Remark 7.3.9. Observe that the bound of both the quadrature and the cutoff error is given by a product of one factor that depends exclusively on the payoff and another factor that depends on the asset dynamic. This property makes it easy to evaluate the bound for a specific option under different dynamics of the asset price. In Subsection
7.4.4 we analyse the terms that depend on the payoff function for the particular case of call options.

Remark 7.3.10. From (7.53) it is evident that the speed of the exponential convergence of the trapezoidal rule for analytic functions is dictated by the width of the strip in which the function being transformed is analytic. Thus, in the limit of small error tolerances, it is desirable to set \( \alpha \) as large as possible to obtain optimal rates. However, non-asymptotic error tolerances are often practically relevant and in these cases the tradeoff between optimal rates and the constant term \( |\hat{g}_\alpha| \) becomes non-trivial. As an example, for the particular case of the Merton model, we have that any finite value of \( \alpha \) will do. However, this improvement of the rate of spectral convergence is more than compensated for by the divergence in the constant term.

The integrals in (7.53) and (7.54) can, in some cases, be computed analytically, or bounded from above by a closed form expression. Consider for instance dissipative models with finite jump intensity. These models are characterised by \( \sigma^2 > 0 \) and \( C(\lambda) = 0 \). Thus the integrals can be expressed in terms of the cumulative normal distribution \( \Phi \):

\[
\int_{\mathbb{R}} e^{-\gamma \frac{\omega^2}{2}} d\omega = \sqrt{\frac{2\pi}{\tau \sigma^2}},
\]

(7.55)

\[
\int_{-\infty}^{\infty} e^{-\gamma \frac{\omega^2}{2}} d\omega = \sqrt{\frac{2\pi}{\tau \sigma^2}} \left( 1 - \Phi \left( \sqrt{\tau \sigma^2} \right) \right)
\]

(7.56)

Now we consider the case of pure-jump processes (ie, \( \sigma^2 = 0 \)) that satisfy the condition \( C(\lambda) > 0 \) for some \( \lambda \in (0, 2) \). In this case the integrals are expressible in terms of the incomplete gamma function \( \gamma \). First, let us define the auxiliary integral:

\[
I(a, b) \equiv e^{-a} + a^{-\frac{1}{2}} \gamma \left( \frac{1}{b}, a \right)
\]
for $a, b > 0$. Using this, the integrals become:

\[
\int_{\mathbb{R}} e^{-\tau \frac{|\omega|^{2-\lambda}}{4} C(\lambda) 1_{|\omega| > 1}} = 2 \left( 1 + I \left( \frac{\tau C(\lambda)}{4}, 2 - \lambda \right) \right)
\]

(7.57)

\[
\int_{\varsigma}^{\infty} e^{-\tau \frac{|\omega|^{2-\lambda}}{4} C(\lambda) 1_{|\omega| > 1}} \begin{cases} 
I \left( \frac{\tau C(\lambda)}{4}, 2 - \lambda \right) + 1 - \varsigma & \varsigma < 1 \\
\varsigma I \left( \frac{\tau^{2-\lambda} C(\lambda)}{4}, 2 - \lambda \right) & \varsigma \geq 1
\end{cases}
\]

(7.58)

An example of a process for which the previous analysis works is the CGMY model presented in \[9,10\], for the regime $Y > 0$.

Lastly, when both $C(\lambda)$ and $\sigma^2$ are positive, the integrals in (7.53) and (7.54) can be bounded by a simpler expression. Consider the two following auxiliary bounds for the same integral, in which $\varsigma \geq 1$:

\[
\int_{\varsigma}^{\infty} e^{-\tau \frac{\sigma^2}{\tau} \omega^2 + \frac{|\omega|^{2-\lambda}}{4} C(\lambda)} d\omega \leq e^{-\tau \frac{\sigma^2}{\tau} \varsigma^2} \int_{\varsigma}^{\infty} e^{-\tau \frac{|\omega|^{2-\lambda}}{4} C(\lambda)} d\omega
\]

(7.59)

\[
= \varsigma e^{-\tau \frac{\sigma^2}{\tau} \varsigma^2} I \left( \frac{\tau \varsigma^{2-\lambda} C(\lambda)}{4}, 2 - \lambda \right)
\]

\[
\int_{\varsigma}^{\infty} e^{-\tau \frac{\sigma^2}{\tau} \omega^2} d\omega \leq e^{-\tau \frac{2-\lambda}{4} C(\lambda)} \int_{\varsigma}^{\infty} e^{-\tau \frac{\sigma^2}{\tau} \omega^2} d\omega
\]

(7.60)

\[
= \sqrt{\frac{2\pi}{\tau \sigma^2}} e^{-\tau \frac{\varsigma^2-\lambda}{4} C(\lambda)} \left( 1 - \Phi \left( \varsigma \sqrt{\tau \sigma^2} \right) \right)
\]

We have that $b(\varsigma)$, defined as the minimum of the right hand sides of the two previous equations,

\[
b(\varsigma) = \min \left\{ \varsigma e^{-\tau \frac{\sigma^2}{\tau} \varsigma^2} I \left( \frac{\tau \varsigma^{2-\lambda} C(\lambda)}{4}, 2 - \lambda \right), \sqrt{\frac{2\pi}{\tau \sigma^2}} e^{-\tau \frac{\varsigma^2-\lambda}{4} C(\lambda)} \left( 1 - \Phi \left( \varsigma \sqrt{\tau \sigma^2} \right) \right) \right\}
\]
is a bound for the integral. Bearing this in mind we have

\[ \int_{\mathbb{R}} e^{-\tau \left( \frac{\sigma^2}{2} \omega^2 + \frac{|\omega|^{2-\lambda}}{4} C(\lambda) 1_{|\omega|>1} \right)} d\omega \leq 2\Phi \left( \sqrt{\tau \sigma^2} \right) - 1 + 2b(1) \] (7.61)

and

\[ \int_{\varsigma}^{\infty} e^{-\tau \left( \frac{\sigma^2}{2} \omega^2 + \frac{|\omega|^{2-\lambda}}{4} C(\lambda) 1_{|\omega|>1} \right)} d\omega \leq b(\varsigma) \] (7.62)

provided that \( \varsigma \geq 1 \).

### 7.4 Computation and minimization of the bound

In this section, we present numerical examples on the bound presented in the previous section using practical models known from the literature. We gauge the tightness of the bound compared to the true error using both dissipative and pure-jump processes. We also demonstrate the feasibility of using the expression of the bound as a tool for choosing numerical parameters for the Fourier inversion.

#### 7.4.1 Call option in variance gamma model

The variance gamma model provides a test case to evaluate the bound in the pure-jump setting. We note that of the two numerical examples presented, it is the less regular one in the sense that \( \sigma^2 = 0 \) and \( C(\lambda) = 0 \) for \( 0 < \lambda < 2 \), indicating that Theorem 7.3.5 in particular is not applicable.

The Lévy measure of the VG model is given by:

\[ \nu_{VG} (dy) = dy \left( 1_{y>0} \frac{Ke^{\eta+y}}{y} - 1_{y<0} \frac{Ke^{\eta-y}}{y} \right) \]
and the corresponding characteristic function is given by eq. (7) of [26]:

\[
\varphi_r(\omega) = \left(1 - i\theta\chi\omega + \frac{\sigma^2\chi}{2}\right)^{-\frac{\xi}{\chi}}
\]

\[
K = \chi^{-1}
\]

\[
\eta_- = \left(\sqrt{\frac{\sigma^2\chi^2}{4} + \frac{\sigma^2\nu}{2} - \frac{\theta\chi}{2}}\right)^{-1}
\]

\[
\eta_+ = \left(\sqrt{\frac{\sigma^2\chi^2}{4} + \frac{\sigma^2\nu}{2} + \frac{\theta\chi}{2}}\right)^{-1}
\]

By Proposition [7.3.3] we get that

\[
a < \min\{\eta_- - \alpha, \eta_+ + \alpha\} \quad (7.63)
\]

which, combined with the requirement that \(g_\alpha \in L^1(\mathbb{R})\) (cf, Remark [7.3.6]), implies:

\[
a < \min\{\eta_+ - \alpha, \eta_- + \alpha, \alpha - 1\} \quad (7.64)
\]

\[
a < \min\{\eta_+ - \alpha, \eta_- + \alpha, -\alpha\}
\]

for calls and puts, respectively. We note that evaluation of the integral in (7.13) is possible also for \(\alpha \in (0, 1)\) and for \(\alpha < 0\). In fact, there is a correspondence between shifts in the integration countour and put-call parity. Integrals with \(\alpha < 0\) give rise to put option prices instead of calls. For an extended discussion of this, we refer to [24] or [6], in which conformal deformation of the integration contour is exploited in order to achieve improved numerical accuracy.

In [24] and in our calculations the parameters equal \(\eta_+ = 39.7840\), \(\eta_- = 20.2648\) and \(K = 5.9311\).

Table [7.1] presents the specific parameters and compares the bound for the VG model with the results obtained by [24]. Based on the table, we note that for the
VG model presented in [26] we can achieve comparable or better error bounds when compared to the study by Lee.

To evaluate the bound, we perform the integration of (7.27) and (7.48) by relying on the Clenshaw-Curtis quadrature method provided in the SciPy package. To supplement Table 7.1 for a wide range of \( n \), we present the magnitude of the bound compared to the true error in Figure 7.1.

In Figure 7.1 we see that the choice of numerical parameters for the Fourier inversion has a strong influence on the error of the numerical method. One does not in general have access to the true solution. Thus, the parameters need to be optimised with respect to the bound. Recall that \( \mathcal{E} = \mathcal{E}(\alpha, \Delta \omega, a, n) \) and \( \mathcal{E} = \bar{\mathcal{E}}(\alpha, \Delta \omega, n) \) denote the true and estimated errors, respectively. Keeping the number of quadrature points \( n \) fixed, we let \( (\alpha_1, \Delta \omega_1, a_1) \) and \( (\alpha_2, \Delta \omega_2) \) denote the minimisers of the estimated and true errors, respectively

\[
(\alpha_1, \Delta \omega_1, a_1) = \arg \inf \bar{\mathcal{E}} \quad (7.65)
\]
\[
(\alpha_2, \Delta \omega_2) = \arg \inf \mathcal{E} \quad (7.66)
\]
We further let $E_1$ and $E_2$ denote the true error as a function of the parameters minimising the estimated and the true error, respectively

$$E_1 = E(\alpha_1, \Delta\omega_1) \quad (7.67)$$
$$E_2 = E(\alpha_2, \Delta\omega_2) \quad (7.68)$$

In Figure 7.1 we see that the true error increases by approximately an order of magnitude when optimising to the bound instead of to the true error, translating into a two-fold difference in the number of quadrature points needed for a given tolerance. The difference between $E_1$ and the bound is approximately another order of magnitude and necessitates another two-fold number of quadrature points compared to the theoretical minimum.

In Figure 7.2, we present the true error $E$ for the Fourier method for the two test cases in Table 7.1. We note that while minimising error bounds will produce sub-optimal results, the numerical parameters that minimise the bound are a good approximation of the true optimal parameters. This, of course, is a consequence of

\[\text{The reference value to compute the true error was obtained by the numerical methods with } n \text{ and } \Delta\omega \text{ such that the level of accuracy is of the order } 10^{-10}.\]
Table 7.1: The error bound for European call/put options in the VG model for select examples. Reference result $\mathcal{E}^*$ from [24]

<table>
<thead>
<tr>
<th>$K$</th>
<th>80</th>
<th>90</th>
<th>100</th>
<th>110</th>
<th>120</th>
</tr>
</thead>
<tbody>
<tr>
<td>$12\tau = 1$</td>
<td>$\alpha$</td>
<td>$-16.9$</td>
<td>$-13.8$</td>
<td>$21.6$</td>
<td>$29.10$</td>
</tr>
<tr>
<td></td>
<td>$a$</td>
<td>$3.33$</td>
<td>$6.45$</td>
<td>$18.1$</td>
<td>$9.77$</td>
</tr>
<tr>
<td>$N = 32$</td>
<td>$\omega_{\text{max}}$</td>
<td>$229$</td>
<td>$229$</td>
<td>$363$</td>
<td>$363$</td>
</tr>
<tr>
<td></td>
<td>$\overline{\mathcal{E}}$</td>
<td>$3.35 \times 10^{-4}$</td>
<td>$0.00334$</td>
<td>$0.00562$</td>
<td>$3.97 \times 10^{-4}$</td>
</tr>
<tr>
<td></td>
<td>$\overline{\mathcal{E}}^*$</td>
<td>$6 \times 10^{-4}$</td>
<td>$0.0032$</td>
<td>$0.0058$</td>
<td>$6 \times 10^{-4}$</td>
</tr>
<tr>
<td>$12\tau = 4$</td>
<td>$\alpha$</td>
<td>$-13.8$</td>
<td>$-13.8$</td>
<td>$22.1$</td>
<td>$23.7$</td>
</tr>
<tr>
<td></td>
<td>$a$</td>
<td>$6.11$</td>
<td>$6.11$</td>
<td>$17.9$</td>
<td>$15.2$</td>
</tr>
<tr>
<td>$N = 8$</td>
<td>$\omega_{\text{max}}$</td>
<td>$62.4$</td>
<td>$42.4$</td>
<td>$84.9$</td>
<td>$126$</td>
</tr>
<tr>
<td></td>
<td>$\overline{\mathcal{E}}$</td>
<td>$3.99 \times 10^{-4}$</td>
<td>$0.00312$</td>
<td>$0.00398$</td>
<td>$3.57 \times 10^{-4}$</td>
</tr>
<tr>
<td></td>
<td>$\overline{\mathcal{E}}^*$</td>
<td>$1.3 \times 10^{-3}$</td>
<td>$0.0057$</td>
<td>$0.0055$</td>
<td>$9 \times 10^{-4}$</td>
</tr>
</tbody>
</table>

Remark 7.4.1. In practice, the Hardy norm in coefficient $M$ reduces to evaluating an $L^1$ norm along the two boundaries of the strip of width $2a$. We find that, for practical purposes, the performance of the Clenshaw-Curtis quadrature of the QUADPACK library provided by SciPy library is more than adequate, enabling the evaluation of the bound in a fraction of a second.

For example, the evaluations of the bounds in Table 7.1 take only around 0.3 seconds to evaluate on Mid 2014 Macbook Pro equipped with a 2.6 GHz Intel Core i5 processor, this without attempting to optimize or parallelize the implementation and while checking for input sanity factors such as the evaluation of the characteristic function in a domain that is a subset in the permitted strip.

We believe that through optimizing routines, skipping sanity checks for inputs and using a lower-level computation routines this can be optimized even further, guaranteeing a fast performance even when numerous evaluations are needed.

Remark 7.4.2. Like many other authors, we note the exceptional guaranteed accuracy of the FT-method, with only dozens of quadrature points. This is partially a
Table 7.2: Numerical performance of the bound for the Kou model, with the test case in [38] (see also [13]) with the number of quadrature points set to $n = 32$. The point of comparison $E^*$ refers to the corresponding bound computed with the method described in Chapters 6.1 to 6.4 of [24]. In the $E^\dagger$, the cutoff error has been evaluated using a computationally more intensive Clenshaw-Curtis quadrature instead of an asymptotic argument with an exponentially decaying upper bound for the option price.

result of the regularity of the European option price. Numerous Fourier-based methods have been developed for pricing path-dependent options. One might, for the sake of generality of implementation, be tempted to use these methods for European options as well, correcting for the lack of early exercise opportunities. This certainly can be done, but due to the weakened regularity, the required numbers of quadrature points are easily in the thousands, even when no rigorous bound for the error is required.

We raise one point of comparison, the the European option pricing example in Table 2 of [20], which indicates a number of quadrature points for pricing the option in the range of thousands. With the method introduced, to guarantee $\bar{E} \approx 10^{-3}$, even with no optimisation, $n = 64$ turns out to be sufficient.

### 7.4.2 Call options under Kou dynamics

For contrast with the pure-jump process presented above, we also test the performance of the bound for Kou model and present relevant results in 7.2. This model differs from the first example not only by being dissipative but also in regularity, in the sense that the maximal width of the domain $A_a$ is, for the case at hand, considerably...
The Lévy measure in the Kou model is given by

$$\nu_{\text{Kou}}(dy) = \lambda \left( pe^{-\eta_1 y} 1_{y>0} + q e^{\eta_2 y} 1_{y<0} \right)$$

with $p + q = 1$. For the characterisation given in [38] the values are set as

$$\lambda = 0.1, \quad r = 0.05, \quad \tau = 0.25, \quad S_0 = 100$$

$$p = 0.3445, \quad \eta_1 = 3.0465, \quad \eta_2 = 3.0775$$

from the expression of the characteristic exponent (see [23])

$$\Psi(z) = z \left( r - \frac{\sigma^2}{2} - \lambda \zeta \right) + \frac{z^2 \sigma^2}{2} + \lambda \left( \frac{p \eta_1}{\eta_1 - z} + \frac{q \eta_2}{\eta_2 + z} - 1 \right)$$

it is straightforward to see

$$A_a \subset \{ z \in \mathbb{C} : \text{Im} z \in (-3.0465, 3.0775) \}$$

This range is considerably narrower than that considered earlier. In the case of transforming the option prices in strike space, the relevant expressions for option prices as well as the error bounds contain a factor exponential in $k$. The practical implication of this is that for deep out of the money calls, it is often beneficial to exploit the put-call-parity and to compute deep in the money calls. However, in the case at hand, the strip width does not permit such luxury. As a consequence, the parameters that minimize the bound are near-identical through a wide range of moneyness, suggesting use of FFT algorithm to evaluate the option prices at once for a range of strikes.
7.4.3 Binary option in the Merton model

For the particular case of Merton model, the Lévy measure is given by

\[ \nu_{\text{Merton}}(dy) = \frac{\lambda}{\sqrt{2\pi\sigma^2}} \exp\left(\frac{-(y - r_j)^2}{2\sigma_j^2}\right) \]

and the characteristic exponent correspondingly by

\[ \Psi(z)_{\text{Merton}} = z\left(\frac{r - \sigma^2z}{2}\right) + \frac{\sigma^2z^2}{2} + \lambda \left(e^{zr_j + \frac{\sigma_j^2z^2}{2}} - 1 - z\left(e^{zr_j + \frac{\sigma_j^2}{2}} - 1\right)\right) \]

we may employ a fast semi-closed form evaluation of the relevant integrals instead of resorting to quadrature methods. We choose the Merton model as an example of bounding the error of the numerical method for such a model. The parameters are adopted from the estimated parameters for S&P 500 Index from [2]:

\[ S_0 = 100, \quad \lambda = 0.089, \quad \sigma = 0.1765, \quad r = 0.05, \quad r_j = -0.8898, \quad \sigma_j = 0.4505 \]

In Figure 7.3, we present the bound and true error for the Merton model to demonstrate the bound on another dissipative model. The option presented is a binary option with finite support on [95, 105]; no damping was needed or used. We note that like in the case of the pure-jump module presented above, our bound reproduces the qualitative behaviour of the true error. The configuration resulting from optimising the bound is a good approximation of the true error. Such behaviour is consistent through the range of \( n \) of the most practical relevance.

7.4.4 Call options

In Subsection 7.3.4 explicit expressions to bound \( \mathcal{E} \) are provided. To evaluate these bounds it is necessary to compute \( \|\hat{g}_\alpha\|_{L^\alpha} \) and \( \|\hat{g}_\alpha\|_{L^\infty_\alpha} \). According to Remark 7.3.9, once we compute these values we could use them for any model, provided that they
Figure 7.3: The true error $E_1$ and the bound $\overline{E}$ for the dissipative Merton model, for a range of quadrature points $n$, along with the bound-minimising configurations contrasted to the true error.

satisfy the conditions considered there.

The payoff of perhaps the most practical relevance is that of a call option. Consider $g$ defined by:

$$g(x) = (S_0 e^x - K)^+ = S_0 \left( e^x - e^k \right)^+$$

for which a selection of a damping parameter $\alpha > 0$ is necessary to have the damped payoff in $L^1(\mathbb{R})$ and to ensure the existence of a Fourier transformation. In this case we have

$$\hat{g}_\alpha(\omega) = S_0 \int_\mathbb{R} \exp \left( (1 - \alpha + i\omega) x \right) - \exp \left( k + (i\omega - \alpha) x \right) \, dx$$

$$= S_0 \exp \left( (1 - \alpha + i\omega) k \right) \frac{1}{(1 + i\omega - \alpha)(i\omega - \alpha)}$$

(7.69)

(7.70)

and

$$|\hat{g}_\alpha(\omega)|^2 = \frac{S_0^2 e^{2(1-\alpha)k}}{(\alpha^2 + \omega^2)(1 - \alpha)^2 + \omega^2}$$

(7.71)
It is easy to see that the previous expression decreases as $|\omega|$ increase. This yields

$$
\|\hat{g}_\alpha\|_{L^\infty_R} = |\hat{g}_\alpha(0)| = \frac{S_0e^{(1-\alpha)k}}{\alpha^2 - \alpha}
$$

(7.72)

and

$$
\|\hat{g}_\alpha\|_{L^\infty_{[\kappa, \infty)}} = |\hat{g}_\alpha(\kappa)|
$$

(7.73)

The maximisation of $|\hat{g}_\alpha|$ in the strip $A_a$ of the complex plane is more subtle. Denoting $\hat{g}_\alpha(\eta, \rho) = \hat{g}_\alpha(\eta + i\rho)$, we look for critical points that satisfy $\partial_\eta |\hat{g}_\alpha| = 0$. This gives

$$
4\eta^3 + 2\eta \left( 4\rho_\alpha + 2\alpha^2 - 2\rho - 2\alpha + \rho^2 + 1 \right) = 0.
$$

(7.74)

For $\rho$ fixed, $|\hat{g}_\alpha|$ has a vanishing derivative with respect to $\eta$ at a maximum of three points. Of the three roots of the derivative, only the one characterised by $\eta = 0$ is a local maximum, giving us that for call options

$$
\|\hat{g}_\alpha\|_{L^\infty_{A_a}} = \max_{\rho \in [-a, a]} |\hat{g}_\alpha(0, \rho)|
$$

(7.75)

Now, observe that $|\hat{g}_\alpha(0, \rho)|$ is a differentiable real function of $\rho$, whose derivative is given by the following polynomial of second degree:

$$
p(\rho) \equiv k \left( \rho + \alpha - 2\rho_\alpha - \alpha^2 - \rho^2 \right) - 2\alpha - 2\rho + 1
$$

(7.76)

We conclude that

$$
\|\hat{g}_\alpha\|_{L^\infty_{A_a}} = \max_{\rho \in B} \{ |\hat{g}_\alpha(0, \rho)| \}
$$

(7.77)

where $B$ is the set of no more than four elements consisting of $a; -a; and the real
roots of $p$ that fall in $(-a,a)$.

**Remark 7.4.3.** So far, we have assumed the number of quadrature points $n$ to be constant. In real life applications, however, this is often not the case. Typically the user might want to choose a minimal $n$ that is sufficient to guarantee error that lies within a pre-defined error tolerance.

In such a case, we propose the following, very simplistic scheme for optimising numerical parameters and choosing the appropriate $n$ to satisfy error smaller than $\epsilon$:

1. Select $n = n_0$ and optimize to find the relevant configuration

2. See if $\mathcal{E}_Q + \mathcal{E}_F < \epsilon$, if not, increase $n$ by choosing it from a pre-determined increasing sequence $n = n_j$ and repeat procedure.

Especially in using FFT algorithms to evaluate the Fourier transforms, we propose $n_j = 2^j n_0$. We further note that typically the optimal configuration for the optimizing configuration for $n_{j+1}$ quadrature points does not differ too dramatically from the configuration that optimizes bounds for $n_j$.

### 7.5 Conclusion

We have presented a decomposition of the error committed in the numerical evaluation of the inverse Fourier transform needed in asset pricing for exponential Lévy models into truncation and quadrature errors. For a wide class of exponential Lévy models, we have presented an $L^\infty$-bound for the error.

The error bound differs from the earlier work presented in [24] in the sense that it does not rely on the asymptotic behaviour of the option payoff at extreme strikes or option prices, allowing pricing a wide variety of non-standard payoff functions such as the ones in [36]. The bound, however, does not take into account path-dependent options. We argue that the error for the methods that allow evaluating american, bermudan, or knockoff options are considerably more cumbersome and
produce significantly larger errors so that in implementations where performance is important, such as calibration, using American option pricing methods for European options is not justified.

The bound also provides a general framework in which the truncation error is evaluated using a quadrature method that remains invariant regardless of the asymptotic behaviour of the option price function. The structure of the bound allows for a modular implementation that decomposes the error components arising from the dynamics of the system and the payoff into a product form for a large class of models, including all dissipative models. On select examples, we also demonstrate the performance that is comparable or superior to the relevant points of comparison.

We have focused on the minimization of the bound as a proxy for minimizing numerical error. Doing this, one obtains, for a given parametrization of a model, a rigorous $L^\infty$ bound for the error committed in solving the European option price. We have shown that the bound reproduces the qualitative behaviour of the actual error. This supports the argument for selecting numerical parameters in a way that minimizes the bound, giving evidence that this selection will, besides guaranteeing numerical precision, be close to the actual minimizing configuration that is not often achievable at an acceptable computational cost.

The bound can be used in the primitive setting of establishing a strict error bound for the numerical estimation of option prices for a given set of physical and numerical parameters or as a part of a numerical scheme, whereby the end user wishes to estimate an option price either on a single point or in a domain up to a predetermined error tolerance.

In the future, the error bounds presented can be used in efforts requiring multiple evaluations of Fourier transformations. Examples of such applications include multi-dimensional Fourier transformations, possibly in sparse tensor grids, as well as time-stepping algorithms for American and Bermudan options. Such applications
are sensitive towards the error bound being used, as any numerical scheme will be required to run multiple times, either in high dimension or for multiple time steps (or both).

### 7.6 Acknowledgements

Häppölä, Crocce are members and Tempone is the director KAUST Strategic Research Initiative for Uncertainty quantification.

We wish to thank an anonymous referee for their critique that significantly improved the quality of the manuscript.

### 7.7 Declarations of Interest

The authors report no conflicts of interest. The authors alone are responsible for the content and writing of the paper.


**Bibliography**


7.A Truncation bound in Lee’s scheme

For the strike-space transformed option price we have

\[ \tilde{f}_\alpha(\omega) = \frac{\varphi_1(\omega - i(\alpha + 1))}{(i\omega + \alpha)(i\omega + \alpha + 1)} \]

In the particular case of the Kou model, we have

\[ \varphi_1(\omega) = e^{\tau(i\omega(r - \frac{\sigma^2}{2} - \lambda\zeta) - \frac{\sigma^2\sigma^2}{2} + \lambda\left(\frac{p\eta_1}{\eta_1 - i\omega} - \frac{q\eta_2}{\eta_2 + 1 + i\omega - 1}\right))} \]

Thus,

\[ |\varphi_1(\omega)| = e^{\tau\left((\alpha + 1)(r - \frac{\sigma^2}{2} - \lambda\zeta) + \frac{(\alpha + 1)^2 - \sigma^2}{2}\right)} e^{\tau\lambda\Re\left(\frac{p\eta_1}{\eta_1 - \alpha - 1 - i\omega} - \frac{q\eta_2}{\eta_2 + 1 + i\omega - 1}\right)} \]

We bound the first jump term resulting from upward jumps as

\[ \frac{p\eta_1}{\eta_1 - \alpha - 1 - i\omega} = \frac{p\eta_1 (\eta_1 - \alpha - 1 + i\omega)}{(\eta_1 - \alpha - 1)^2 + \omega^2} \]

from which it follows

\[ \Re\left(\frac{p\eta_1}{\eta_1 - \alpha - 1 - i\omega}\right) \leq \frac{p\eta_1 (\eta_1 - \alpha - 1)}{(\eta_1 - \alpha - 1)^2} \]

similar inequality holds for the term resulting from downward jumps giving us that

\[ |\varphi_1(\omega)| \leq e^{\tau\left((\alpha + 1)(r - \frac{\sigma^2}{2} - \lambda\zeta) + \frac{(\alpha + 1)^2 - \sigma^2}{2}\right)} e^{\tau\lambda\left(\frac{p\eta_1 (\eta_1 - \alpha - 1)}{(\eta_1 - \alpha - 1)^2} + \frac{q\eta_2 (\eta_2 + \alpha + 1)^2 + 1}{(\eta_2 + \alpha + 1)^2 + 1}\right)} \]
The term \( \exp \left( -\frac{\omega^2}{2} \right) \) can be bounded from above by an exponential term \( \exp (\gamma \omega) \):

\[
\exp \left( -\frac{\sigma^2 \omega^2}{2} \right) \leq \exp (\gamma \omega + K) \quad (7.78)
\]

\[-\sigma^2 \omega^2 + 2\gamma \omega - 2K \leq 0\]

the condition for the quadratic to have no solutions is given by

\[
\gamma^2 - 2K\sigma^2 \leq 0,
\]

setting \( K = \frac{1}{2} \), \( \gamma = \sigma \) giving us the bound in eq. (7.78).

This puts us in place to apply theorem 6.1 of [24], with

\[
\Phi(u) = \frac{\exp \left( \tau \left( \frac{\alpha + 1}{\alpha + 1} \left( r - \frac{\sigma^2}{2} - \lambda \zeta \right) + \frac{\sigma^2(\alpha+1)^2}{2} \right) \right) \cdot \tau^\lambda \left( \frac{q\omega_1(q_1 - \alpha - 1)}{\eta_1^2 - \alpha - 1} + \frac{q\omega_2(q_2 + \alpha + 1)}{\eta_2^2 + \alpha + 1} \right) \cdot \eta_1 \cdot \eta_2}{\alpha^2 (\alpha + 1)^2}
\]

\( \gamma = \sigma \)
Chapter 8

Article III

Abstract

This work addresses the problem of pricing American basket options in a multivariate setting, which includes among others, the Bachelier and the Black-Scholes models. In high dimensions, nonlinear partial differential equation methods for solving the problem become prohibitively costly due to the curse of dimensionality. Instead, this work proposes to use a stopping rule that depends on the dynamics of a low-dimensional Markovian projection of the given basket of assets. It is shown that the ability to approximate the original value function by a lower-dimensional approximation is a feature of the dynamics of the system and is unaffected by the path-dependent nature of the American basket option. Assuming that we know the density of the forward process and using the Laplace approximation, we first efficiently evaluate the diffusion coefficient corresponding to the low-dimensional Markovian projection of the basket. Then, we approximate the optimal early-exercise boundary of the option by solving a Hamilton-Jacobi-Bellman partial differential equation in the projected, low-dimensional space. The resulting near-optimal early-exercise boundary is used to produce an exercise strategy for the high-dimensional option, thereby providing a lower bound for the price of the American basket option. A corresponding upper bound is also provided. These bounds allow to assess the accuracy of the proposed pricing method. Indeed, our approximate early-exercise strategy provides a straightforward lower bound for the American basket option price. Following a duality argument due
to Rogers, we derive a corresponding upper bound solving only the low-dimensional optimal control problem. Numerically, we show the feasibility of the method using baskets with dimensions up to fifty. In these examples, the resulting option price relative errors are only of the order of few percent.

8.1 Introduction

This work addresses the problem of pricing American basket options in a multivariate setting. Our approach relies on a stopping rule that depends on the dynamics of a low-dimensional Markovian projection of the given basket of assets.

Pricing path-dependent options is a notoriously difficult problem. Even for relatively simple cases, such as the Black-Scholes model or the Bachelier model, in which an analytic expression of the risk-neutral expected payoff at a terminal time, $T$, can be found, prices of path-dependent options, such as American options, must typically be solved for numerically. This difficulty is aggravated in high dimensions, where convergence rates of well-known numerical methods deteriorate exponentially as the number of dimensions increases. However, there is a plethora of American options being offered in the markets, in publicly traded markets or over-the-counter (OTC). Perhaps the best-known example is that of options written the S&P-100 index quoted on the Chicago Board Options Exchange (CBOE). In addition, the wide variety of exchange traded funds (ETF) tracking indices have American options written on them publicly quoted on CBOE. These funds include many prominent indices such as Euro Stoxx 50 and the Dow Jones Industrial average, as well as many regional indices. If one is interested in the index alone, then a low-dimensional model for the index is clearly sufficient. However, in many situations, consistent joint models of the index together with some or all the individual stocks may be required, which would lead to the moderate and high dimensional option pricing problems addressed in this paper.

The two most widely used approaches to pricing path-dependent options, binomial
tree methods and partial differential equation (PDE) methods, both suffer from the so-called curse of dimensionality. In the case of the probability trees or lattices, the size of the probabilistic trees, even in the case of recombining trees, already becomes prohibitively large in moderate dimensions. The other popular method requires solving the Black-Scholes equation using finite difference (FD) or finite element (FEM) methods. Both methods involve discrete differential operators whose size also scales exponentially in the number of dimensions.

In Monte Carlo simulation, the rate of convergence of weak approximations does not explicitly depend on the number of dimensions. With early-exercise options like American ones, however, Monte Carlo methods become more complicated. Although well suited for forward-propagation of uncertainties in a wide range of models, traditional Monte Carlo methods do not offer a straightforward way to construct an exercise strategy. Such a strategy typically needs to be obtained through backward induction. Because the price of an American option is based on assuming optimal execution of the option, any solution scheme needs to produce the optimal stopping strategy as a by-product of the pricing method. Many methods have been developed to produce a near-optimal execution strategy. [11] introduced a pair of schemes that evaluate upper and lower bounds of the prices of American options. [29] used least-squares regression in conjunction with Monte Carlo simulation to evaluate the price of American options. Their popular method has been widely implemented in various pricing engines, for example in the QuantLib library by [1].

In the least-squares Monte Carlo methodology, the value of holding an option is weighed against the cash flow captured by exercising the option. The intrinsic value of an option is, of course, known. However, the holding price is the discounted expectation of possible future outcomes. This expectation is estimated based on a Monte Carlo sample by regressing the holding price of the option to a few of decision variables or basis functions. Naturally, the choice of the appropriate basis functions
has a crucial effect on the quality of the outcome, and also the number of basis
functions should be much smaller than the size of the Monte Carlo sample to avoid
overfitting \[18,43,44\]. For work on the reduction of the computational complexity in
the regression methods, we refer the reader to \[8\].

Another method to approximate option prices in high dimensions is the optimal
quantizer approach of \[3\]. In this method the diffusion process is projected to a finite
mesh. This mesh is chosen optimally to minimize projection error, the conditional
expectation describing the holding price is then evaluated at each of the mesh points.
The quantization tree approach gives accurate approximations of the option price in
moderate dimension. Here, we present methods for selected-parametrization of the
Black-Scholes model over twice the dimension presented in \[3\] For work with rather
large number of dimensions, we refer the reader to the stratified state aggregation
along payoff (SSAP) method of \[4\]. In the SSAP method, one solves for an exercise
strategy through stratifying possible values of the intrinsic value of the option. \[2\]
used a similar approach for pricing Bermudan swaptions, characterizing the early
exercise boundary in terms of the intrinsic value.

Here, we propose and analyze a novel method for pricing American options written
on a basket of assets. Like the SSAP, the pricing method in this work relies on using
the intrinsic value, or the value of the underlying asset as a state variable. On the
other hand, our method is based on the Markovian projection of the underlying asset,
does not rely on the use of basis functions and provides upper and lower bounds for
the option price. These bounds are useful to assess the accuracy of our methodology.

In this exploratory work, we computationally study the feasibility of using stop-
ning rules based on a simplified surrogate process in pricing American options written
on a basket of assets. The method offers an efficient approximation to pricing and
hedging American options written on an index, or a security tracking such index.
Instead of the full-dimensional process, we use a lower-dimensional process obtained
through Markovian projection. Even though the evolution of the multiple assets involved in a given basket is usually assumed Markovian, the SDE describing the evolution of a linear combination or a basket of assets, is rarely Markovian in the basket value. We address this issue by means of Markovian projection, which provides a low-dimensional Markovian SDE that is suited to dynamic programming (DP) methods that solve the relevant Hamilton-Jacobi-Bellman (HJB) equation. Markovian projection techniques have been previously applied to a range of financial applications, see, for example, [16,34,36].

8.1.0.0.1 Outline The remainder of this work is organized as follows. In Section 8.2, we describe the Markovian projection in the context of projecting high-dimensional SDEs into lower dimensions. We show how the low-dimensional HJB equation gives rise to a stopping rule that in general is sub-optimal but provides a lower bound for the American option price. Using a duality approach from [37], we give an upper bound for the option price using the solution of the low-dimensional HJB equation. We show that in the Bachelier model, the lower and upper bounds coincide and provide an exact option valuation. We prove how the question of whether the cost-to-go function of an American option can be approximated using a low-dimensional approximation reduces to the corresponding question of European options, which are simpler to analyze. It is known that the Bachelier model is a close approximation to the Black-Scholes model in the realm of European option pricing [38]. We motivate that this approximation has a beneficial effect when pricing American basket options with our methodology since our method is exact for the Bachelier model. In Section 8.3, we detail the numerical implementation of the ideas developed in the preceding section and experiment with multivariate Bachelier and Black-Scholes models. Reporting results of numerical experiments, we verify the accuracy of our method with the Bachelier model and give supporting results to justify
the use of our method in cases where neither the European or American option prices can be precisely represented using a low-dimensional approximation. Using the Black-Scholes model as an example, we show that the approximation error of our method is few per cent, comparable to the bid-ask spread of even the more liquid openly traded options and well within the spread of more illiquid index options or options quoted on an ETF. Finally, we offer concluding remarks in Section 8.4.

8.2 Markovian projections and implied stopping times

In this section, we revisit the essential equations that describe risk-neutral option pricing of American options in a multivariate setting. We present in Section 8.2.1 how these equations have corresponding low-dimensional projections that can be obtained using the Markovian projection. In Section 8.2.2, we show how the projected PDEs give rise to lower and upper bounds for the solution of the original high-dimensional pricing problem.

Following the introduction of the relevant bounds, we discuss in Section 8.2.3 classes of models that are of particular interest in reduced-dimension evaluation. First, we recall in Lemma 8.2.7 how the Gaussian Bachelier model has the feature that the Markovian projection produces a one-dimensional SDE whose solution coincides in law with the underlying high-dimensional portfolio. We also show in Corollary 8.2.8 how this one-dimensional approximation property is preserved if the Bachelier model is generalized through the appropriate introduction of a stochastic clock. Secondly, we provide auxiliary results to characterize some Itô SDEs that have this exact reduced dimension structure that our proposed method exploits. Among these ancillary results, we have Lemma 8.2.12 which we use to reduce the discussion of dimension reduction of American options into the problem of analyzing low-dimensional approximations of the corresponding European option. Furthermore, we give a motivation for using the Markovian projection even for models that do not have the exact
reduced dimension property.

8.2.1 Markovian projections and approximate stopping times

Assume that the time evolution of the asset prices in the basket is given by a stochastic process in \( \mathbb{R}^d \), \( X(t, \omega) \), that is the unique strong solution to an Itô SDE,

\[
\begin{align*}
\frac{dX(t)}{dt} &= a(t, X(t)) dt + b(t, X(t)) dW(t), \quad 0 < t < T, \\
X(0) &= x_0,
\end{align*}
\] (8.1)

which is driven by a \( k \)-dimensional Wiener process with independent components, \( W \). We work under the risk-neutral measure and due to a no arbitrage assumption, the drift in (8.1) is a linear function,

\[
a(t, x) = rx,
\] (8.2)

where \( r \in \mathbb{R} \) is the short rate. Most of the discussion can also be generalized with minimal modifications to a time-dependent, stochastic, short rate when the short rate process is independent of the dynamics of the underlying assets, see Remark 8.2.11.

For \( 1 \leq i \leq d \) and \( 1 \leq j \leq k \) the diffusion coefficients, \( b_{ij} \), are at least second order differentiable functions and such that the pdf of \( X(t) \) exists for \( 0 < t \leq T \) and is a univariate, smooth function, cf. Assumption 8.3.1. Furthermore, we denote the canonical filtration generated by \( X(t) \) as \( \mathcal{F}_t = \sigma \{ X(q) : 0 \leq q \leq t \} \).

In the numerical examples in the subsequent section, we directly deal with the models of Bachelier [40] and Black-Scholes [10], acknowledging possible extensions to the constant elasticity of variance (CEV) model (see [15]) that can in a certain sense be understood as a compromise between the Bachelier and Black-Scholes models. Many other extensions are also possible, and we discuss some of them in Section 8.2.3. Note the time-homogeneous structure of the examined models and recognize
possible extensions to time-inhomogeneous models, for instance by using temporal reparametrization.

Furthermore, we assume for simplicity that the underlying pays no dividends. This work focuses extensively on models of Bachelier and Black-Scholes type. They are defined by their respective volatilities, namely

\[ b_{\text{Bachelier}}(t, x) = \Sigma, \]  
\[ b_{\text{Black-Scholes}, ij}(t, x) = x_i \Sigma_{ij}. \]

with \( \Sigma \in \mathbb{R}^{d \times k} \) in both models.

We focus on a portfolio of assets, \( S_1 \), given by weights \( P_1 \),

\[ S_1(t) = P_1 X(t), \]

as the underlying security, for \( P_1 \in \mathbb{R}^{1 \times d} \), with non-zero elements, possibly some but not all negative. We seek to price options with the payoff functional \( g : \mathbb{R} \to \mathbb{R} \). Arguably, the most interesting example is that of the put option, \( g(s) = (K - s)^+ \) for some \( K \in \mathbb{R} \).

The price of the European option written on the portfolio \( P_1 \) with expiry at \( T \) is given by

\[ u_E(t, x) = E \left[ \exp \left( -r (T - t) \right) g \left( P_1 X(T) \right) \mid X(t) = x \right]. \]

In contrast, when pricing American options, we seek to solve for

\[ u_A(t, x) = \sup_{\tau \in T_q} E \left[ \exp \left( -r (\tau - t) \right) g \left( P_1 X(\tau) \right) \mid X(t) = x \right], \]

\[ T_q = \{ \tau : \Omega \to [q, T] \mid \tau \leq t \} \in \mathcal{F}_t, \forall t \in [q, T] \].
The European option price $u_E$ given by (8.6) also satisfies the Black-Scholes equation in $(t, x) \in [0, T] \times D$,

$$-\partial_t u_E (t, x) = -ru_E (t, x) + \sum_i a_i (t, x) \partial_{x_i} u_E (t, x) + \frac{1}{2} \sum_{ij} \left( bb^T \right)_{ij} (t, x) \partial_{x_i x_j}^2 u_E (t, x)$$

$$u_E (T, \cdot) = g (P_1 \cdot),$$

(8.8)

with the appropriate domain $D \subset \mathbb{R}^d$. For example, in the Black-Scholes model, we have $D = D_{BS}^d = \mathbb{R}^d_+$ with the appropriate Dirichlet boundary condition at hyperplanes at which one or more components of $X (t)$ are zero. The boundary value is given by a lower-dimensional version of (8.8). Defining the second order linear differential operator

$$(L v) (t, x) = \left( -r + \sum_i a_i \partial_{x_i} + \frac{1}{2} \sum_{ij} \left( bb^T \right)_{ij} \partial_{x_i x_j}^2 \right) v (t, x),$$

we can write the corresponding non-linear HJB equation. Following the presentation of [?, Equation (6.2)], the American option price, $u_A$, satisfies

$$(L u_A + \partial_t u_A) (t, x) \leq 0, \quad (t, x) \in [0, T] \times D,$$

$$u_A (t, x) \geq g (P_1 x), \quad (t, x) \in [0, T] \times D,$$

$$((L u_A + \partial_t u_A) (t, x)) (u_A (t, x) - g (P_1 x)) = 0, \quad (t, x) \in [0, T] \times D.$$
we write the HJB equation for \( u_A \) shortly as

\[
-\partial_t u_A (t, x) = (H u_A) (t, x), \quad (t, x) \in [0, T] \times D,
\]

\[
u_A (t, \cdot) = g (P_1 \cdot).
\] (8.10)

For the Bachelier model, \( D \) is unbounded. For the Black-Scholes model, one or more components of \( X(t) \) vanish at the boundary \( \partial D \). Since both the drift (8.2) and the volatility (8.4) are linear in their arguments, the drift and the volatility vanish at the boundary. Resulting boundary value is thus given by a lower-dimensional variant of (8.10) where one or more of the components of \( X(t) \) are fixed to zero.

Instead of trying to solve (8.10) directly, we first turn our attention to a low-dimensional approximation of the portfolio process \( S_1 \) introduced in (8.5). This approximation is the Markovian projection of \( S_1 \) [22][35]. Indeed, we approximate the non-Markovian evolution of \( S_1 \) by the following surrogate process,

\[
d \overline{S} (t) = \overline{a}^{(x_0)} (t, \overline{S}(t)) \, dt + \overline{b}^{(x_0)} (t, \overline{S}(t)) \, dW (t), \quad t \in [0, T],
\]

\[
\overline{S} (0) = P_1 x_0,
\] (8.11)

The drift and volatility coefficients in (8.11) are evaluated through conditional expectations, namely

\[
\overline{a}^{(x_0)} (t, s) = \mathbb{E} [P_1 a (t, X(t)) | P_1 X (t) = s, X (0) = x_0],
\] (8.12)

\[
\left( \overline{b}^{(x_0)} \right) (t, s) = \mathbb{E} \left[ \left( P_1 b b^T P_1^T \right) (t, X(t)) | P_1 X (t) = s, X (0) = x_0 \right].
\] (8.13)

The Markovian projection (8.11) generates its canonical filtration, \( \mathcal{F}_t = \sigma \{ \overline{S} (q) : 0 \leq q \leq t \} \).

Observe that the surrogate process, \( \overline{S} (t) \) in (8.11), has, due to the proper selection of the drift and volatility functions and the appropriate initial value, the same marginal density as \( S(t) = P_1 X(t) \) for all \( t \in [0, T] \) [22]. For any given payoff func-
tion $g$ that yields a finite price in (8.6), this implies the identity
\[
E [\exp (-rT) g (P_1 X (T)) | X (0) = x_0] = E [\exp (-rT) g (\bar{S} (T)) | \bar{S} (0) = P_1 x_0],
\]
(8.14)

which means that we can price European options on the basket using only our knowledge of the Markovian process $\bar{S}$.

Assuming that we know the dynamics (8.11), we can evaluate the right-hand side of (8.14) using the Feynman-Kac Formula. By denoting
\[
\bar{u}_E (t, s) = E [\exp (-r (T - t)) g (\bar{S} (t)) | \bar{S} (t) = s],
\]
(8.15)
we have that $\bar{u}_E$ solves a corresponding linear backward PDE in one space dimension only,
\[
-\partial_t \bar{u}_E (t, s) = -r \bar{u}_E (t, s) + \bar{a}^{(x_0)} (t, s) \partial_s \bar{u}_E (t, s) + \frac{\left( \bar{b}^{(x_0)} (t, s) \right)^2}{2} \partial_{ss} \bar{u}_E (t, s), \quad t \in [0, T], \ s \in \bar{D},
\]
\[
\bar{u}_E (T, \cdot) = g (\cdot).
\]
(8.16)

**Remark 8.2.1** (Interpretation of projected PDEs). We have defined the projected PDE (8.16) that is of Black-Scholes type. Furthermore, the coefficients $\bar{a}(x_0)$ and $\bar{b}(x_0)$ of the equation are constructed through conditioning to the initial value of the SDE (8.1). Here, we use the the PDE (8.16) as a mathematical construct to evaluate the expectation (8.14). We do not interpret the solution of (8.16), or its extensions defined in the remainder of this work as tradeable option prices.

Note that the procedure above can be generalized to cases where the Markovian projection is carried out onto a space of dimension $\bar{d} > 1$. This is done simply by intro-
ducing additional portfolios and their weights, \( P^T = [P_1^T, P_2^T, P_3^T, \ldots, P_d^T] \), and defining the multidimensional dynamics for \( \overline{S} \) via the projected volatility coefficients as

\[
(bb^T)_{ij}^{(x_0)}(t, s) = \mathbb{E} \left[ (P_i^T b b^T P_j)(t, X(t)) \mid PX(t) = s, X(0) = x_0 \right], \quad 1 \leq i, j \leq d.
\] (8.17)

Summing up, as long as we can efficiently evaluate the coefficients in the SDE (8.11), it is possible to solve the low-dimensional Equation (8.16) instead of Equation (8.8) that suffers from the curse of dimensionality. Obviously, the efficient evaluation of the coefficients in the SDE of \( \overline{S} \) via conditional expectation as in (8.13) is in principle a daunting task. Section 8.3.1.1 proposes an efficient approximation to carry out this evaluation.

**Remark 8.2.2 (Computational domains and boundary conditions).** Instead of using the full unbounded domain of the PDE (8.16) in the numerical part of this work, we use a modified, computational domain, on which we impose an artificial boundary condition as follows.

First, note that the appropriate domain, \( D \), for (8.8) depends on the model of choice. For the \( d \)-dimensional Black-Scholes model, we have \( D = D_{\text{Black-Scholes}}^d = \mathbb{R}_+^d \) and correspondingly for the Bachelier model, \( D = D_{\text{Bachelier}}^d = \mathbb{R}_+^d \). When numerically solving the full, \( d \)-dimensional Equation (8.8), one often truncates the domain into a compact one and imposes artificial boundary conditions on the boundary of the localized computational domain. Here, we also truncate the projected domain, \( \overline{D} \), into a localized computational domain. At the boundary of the computational domain, we impose the artificial boundary condition \( \bar{u}(t, s) = g(s) \). In addition to the truncation, we note that the coefficients in (8.16) are defined only for regions where the density \( \phi \) of process \( P_1 X(t) \) has support. We extend artificially the domain of (8.16) to the rectangle \([0, T] \times [s_{\text{min}}, s_{\text{max}}]\) by extrapolating the relevant coefficients \( \phi^{(x_0)} \) and
\((b^{(x_0)})^2\). For \((\bar{b}^{(x_0)})^2\) we also set a lower bound to guarantee numerical stability and well-posedness.

In all our numerical examples, we make sure that our truncated and extrapolated computational domain is sufficiently large to make the corresponding domain truncation error negligible. For more in-depth discussions on this matter, we refer the reader to [13, 25, 27, 31].

Furthermore, to maintain brevity of notation, we will refrain from writing explicitly the artificial boundary conditions. All relevant PDEs in this work are understood to be numerically solved using Dirichlet boundary conditions implied by the intrinsic value of the option.

Just as the Black-Scholes equation, \((8.8)\) has a corresponding HJB equation \((8.10)\), we may use the corresponding HJB to the projected Black-Scholes equation \((8.16)\). The resulting HJB equation describes the cost-to-go function \(u_A\) of an American option written on the portfolio that has the projected dynamics of \((8.11)\):

\[
-\partial_t \bar{u}_A(t, s) = \left(\mathcal{L} \bar{u}_A\right)(t, s) \max(\pi_A(t, s), \pi_A(t, s) - g(s)) > 0 = \left(\mathcal{H} \bar{u}_A\right)(t, s) \quad (t, s) \in [0, T] \times \overline{D},
\]

\[
\bar{u}_A(T, \cdot) = g(\cdot).
\]

However, for American option prices, there is no identity corresponding to equality \((8.14)\). As a result, the magnitude of the difference \(|\bar{u}_A\left(0, P_1x_0\right) - u_A\left(0, x_0\right)|\) may not necessarily be small. Also, the boundary conditions in \((8.18)\) are subject to the same ambiguity as the ones of \((8.16)\) discussed in Remark 8.2.2. The main focus of this work is to address these issues and to estimate the difference between the computed value of \(\bar{u}_A\) and the sought \(u_A\), which is assumed beyond our reach being too costly to compute.

We note in passing that the processes \(X\) and \(\mathcal{S}\) live in different probability spaces.
Likewise, the stopping times corresponding to the full-dimensional and projected SDE are adapted to $\mathcal{F}_t$ and $\mathcal{F}_t$, respectively.

### 8.2.2 Implied stopping time and price bounds

Above we have laid out the question of the feasibility of using the projected dynamics $\overline{S}$ in pricing American options, we now show below in Section 8.2.2.1 how the solution of the projected problem $\pi_A$ gives rise to an exercise strategy that is sub-optimal. This sub-optimal exercise strategy gives a lower bound for the option price. We complement this lower bound with a corresponding upper bound in Section 8.2.2.2.

#### 8.2.2.1 Lower bound

In the full American option pricing problem (8.7), the optimal stopping time, $\tau^* \in \mathcal{T}_0$, such that

$$u_A (0, x_0) = E \left[ \exp \left( -r \tau^* \right) g (P_1 X (\tau^*)) \mid X (0) = x_0 \right],$$

is given by

$$\tau^* = \inf \left\{ t \in [0, T] : u_A (t, X (t)) = g (P_1 X (t)) \right\}. \quad (8.19)$$

Any stopping time $\tau \in \mathcal{T}_0$ gives a lower bound for the option price. We do not have access to the full cost-to-go function, $u_A$, and hence a natural replacement is given by the projected cost-to-go function $\pi_A$. Indeed, the projected cost-to-go function $\pi_A$ gives rise to two hitting times:

$$\overline{\tau}^* = \inf \left\{ t \in [0, T] : \pi_A (t, \overline{S} (t)) = g (\overline{S} (t)) \right\},$$
where the dynamics of \( S \) is given by (8.11) and

\[
\overline{\tau}^* \equiv \inf \{ t \in [0, T] : \overline{u}_A (t, \mathbf{P}_1 \mathbf{X} (t)) = g (\mathbf{P}_1 \mathbf{X} (t)) \}.
\] (8.20)

We note that due to the terminal condition on \( \overline{u}_A \) in (8.18) all hitting times are bounded by \( T \).

We conclude the discussion on the lower bound of the option value by stating the lower bound implied by the hitting time \( \overline{\tau}^* \in \mathcal{T}_0 \),

\[
u_A (0, \mathbf{x}_0) \geq \mathbb{E} [\exp (-r \overline{\tau}^*) g (\mathbf{P}_1 \mathbf{X} (\overline{\tau}^*)) | \mathbf{X} (0) = \mathbf{x}_0].
\] (8.21)

We emphasize that we have not made a comparison between \( u_A (0, \mathbf{x}_0) \) and \( \overline{u}_A (0, \mathbf{P}_1 \mathbf{x}_0) \).

**Remark 8.2.3** (On least-squares Monte Carlo). *The approach we have adopted shares some similarities with the least-squares Monte Carlo approach. However, there are key differences: In the least-squares Monte Carlo method, the stopping time can be understood as a hitting time into a region where the holding value of the option, as estimated through regression to a set of basis functions, is exceeded by the early exercise price. The hitting time (8.20) is likewise defined as a comparison between the estimated cost-to-go function, \( u_A \), and the early exercise price. However, the estimated cost-to-go function, \( \overline{u}_A \), does not depend on a choice of basis functions, only on the direction of the projection. On the other hand, \( \overline{u}_A \) is constructed using the Markovian projection \( \overline{S} \) instead of the true forward model \( \mathbf{X} \).*

### 8.2.2.2 Upper bound

To assess the accuracy of approximating the process with a low-dimensional Markovian projection, we want to devise a corresponding upper bound. For this, we use the dual representation due to [37].

The dual representation of the pricing problem is as follows. The price of the
American option is given by:

\[ u_A(0, x_0) = \inf_{R \in H_0^1} \mathbb{E} \left[ \sup_{0 \leq t \leq T} \left( \tilde{Z}(t) - R(t) \right) \right] |X(0) = x_0], \quad (8.22) \]

where \( H_0^1 \) denotes the space of all integrable martingales \( R, t \in [0, T] \) such that for \( R \in H_0^1 \)

\[ \sup_{0 \leq t \leq T} |R(t)| \in L^1, \]

\[ R(0) = 0. \]

Here \( \tilde{Z}(t) \) denotes the discounted payoff process

\[ \tilde{Z}(t) = \exp\left(-rt\right) g(X(t)), \quad t \in [0, T]. \quad (8.23) \]

Naturally, evaluating the statement within the infimum of Equation (8.22) with any martingale, \( R(t) \in H_0^1 \), will give an upper bound to the option price. A martingale, \( R^* (t) \), reaching the infimum (8.22) is called an optimizing martingale. In general, finding an optimizing martingale is as complex as finding the solution to the pricing problem. In fact, when the cost-to-go function, \( u_A(t, x) \), is known, the optimizing martingale can be written out following the approach in [24]:

\[ dR^* (t) = \exp\left(-rt\right) \left((\nabla u_A)^T b\right)(t, X(t)) dW(t), \quad t \in [0, T], \]

\[ R^* (0) = 0. \quad (8.24) \]

We construct a near-optimal martingale \( R^* \in H_0^1 \) by replacing in (8.24) the exact \( u_A \) with the approximate cost-to-go function, \( \pi_A \). This yields the explicit upper bound

\[ u_A(0, x_0) \leq \mathbb{E} \left[ \sup_{0 \leq t \leq T} \left( \tilde{Z}(q) - R^*(t) \right) \right] |X(0) = x_0], \quad (8.25) \]
where
\[
dR^*(t) = \exp(-rt) \left( (\nabla u_A)^T (t, P_1 X(t)) \right) P_1 b(t, X(t)) dW(t), \quad t \in [0, T],
\]
\[R^*(0) = 0.\] 

In other words, we evaluate the sensitivity, or delta, of the projected, approximate value function using the projected, non Markovian, version of the true stochastic process. We also note that the sensitivity of the projected, approximate value function can be used as an approximate sensitivity of the option value with regard to the value of the underlying portfolio. [?, chapter 3]

8.2.3 Dimension reduction for models relevant to quantitative finance

We have established a lower as well as an upper bound for the American basket option prices using Markovian projection. The question of which models feature tight bounds is naturally of interest for the applicability of our methodology. Thus, this section focuses on the domain of applicability of the Markovian projection. Below, we demonstrate that the procedure of Markovian projection produces exact results for the Bachelier Model. This is a consequence of the Gaussian returns in the model. In fact, it turns out that due to the constant volatility \((8.3)\) of the Bachelier model, the coefficients of the relevant low-dimensional PDEs can be evaluated without Laplace approximation.

Following our discussion about the Bachelier model, we then concentrate on the Black-Scholes model, which is known to produce option prices that are well approximated by the Bachelier model. Finally, we state conditions under which the Black-Scholes model also satisfies the property that the value function of the option depends only on a single state variable \(s\), namely the portfolio value \(P_1 x\).
8.2.3.1 Definitions

First, let us define some terminology. Let \( 1 \leq n < d \) and \( D \subset \mathbb{R}^d \) be a convex set with piecewise smooth boundary.

**Definition 8.2.4.** We call a function \( v : D \to \mathbb{R} \) essentially \( n \)-dimensional if there exist a function \( \zeta : \mathbb{R}^n \to \mathbb{R} \) and a matrix \( N \in \mathbb{R}^{n \times d} \) with orthogonal rows such that \( v : \mathbb{R}^d \to \mathbb{R} \) is given by

\[
v(x) = \zeta(Nx).
\]

**Definition 8.2.5.** By extension, we call a differential operator \( K \) essentially \( n \)-dimensional if the following backward PDE is well posed

\[
-\partial_t w(t, x) = K w(t, x), \quad (t, x) \in [0, T) \times D,
\]

\[
w(T, \cdot) = w_T(\cdot),
\]

and it has an unique essentially \( n \)-dimensional solution for any essentially \( n \)-dimensional terminal value, \( w_T \). Here we specifically mean that the function \( \zeta \) may depend on time, that is

\[
w(t, x) = \zeta(t, Nx),
\]

but the matrix \( N \) does not.

**Remark 8.2.6** (Time independence of lower dimensional subspaces). The definition above rules out solutions to (8.27) that are essentially lower-dimensional in each instant of time although the directions along which such functions have non-vanishing partial derivatives change over time. We also tacitly assume in this definition that the allowed terminal values make the problem (8.27) well posed. We later exploit this structure when proving Lemma 8.2.12 that allows us to reduce the analysis of
essentially low-dimensional models to the study of European value functions only, disregarding the possibility for early exercise.

8.2.3.2 Bachelier model

First, we prove that the Markovian projection gives exact results even for American options pricing when used on the Bachelier model. This arises from the fact that the Markovian-projected basket $\overline{S}$ coincides in law with the true basket $P_1X$. After discussing the one-dimensional nature of the Bachelier model, we propose possible extensions introducing a stochastic clock.

Lemma 8.2.7 (Dimension reduction in the Bachelier model). Let $X(t)$ solve \((8.1)\) and the drift and volatility be given by \((8.2)\) and \((8.3)\) respectively. Furthermore, let $\overline{S}(t)$ be the Markovian projection defined by eqs. \((8.11)\), \((8.12)\) and \((8.13)\) for $P_1X(t)$. Then $\overline{S}(t)$ and $P_1X(t)$ coincide in law.

Proof. The proof is direct.

We have established that the multivariate Bachelier model has an essentially one-dimensional generator.

However, we know that the model does not feature fat-tailed distribution for returns or clustering of volatility. Both features have been observed in the markets (see \cite{[33], [30], [14]}). In the following Corollary, we address these issues through the introduction of a stochastic clock. In this way, we introduce a larger class of arbitrage-free dynamics for which the price distribution conditioned to the value of the stochastic clock reduces to the one from the Bachelier model.
Corollary 8.2.8 (Stochastic time change in the Bachelier model). Let $X$ be given by the Bachelier model (8.2, 8.3) and let $U(t)$ be an almost surely increasing process, or a stochastic clock, independent of $X$ in $\mathbb{R}^+$, with $U(0) = 0$. Let the discounted process corresponding to $X$ be

$$R_X(t) = \exp(-rt) \ X(t), \quad (8.28)$$

then a related stock price process $Y$, given by

$$Y(t) = \exp(rt) \exp(-rU(t)) \ X(U(t)) = \exp(rt) \ R_X(U(t))$$

has an essentially one-dimensional generator.

Proof. The proof is divided into two steps.

8.2.3.2.1 Step 1 The combination of (8.28) and (8.1) yields that $R_X$ is a martingale with respect to its canonical filtration. We show that the same holds for $R_Y(t) = \exp(-rt)Y(t) = R_X(U(t))$.

We take $0 \leq s < t \leq T$ and consider the conditional expectation

$$E[R_Y(t) | R_Y(s)] = E[R_X(U(t)) | R_X(U(s))]$$

$$= E[E[R_X(U(t)) | R_X(U(s)), U(t), U(s)] | R_X(U(s))]$$

$$= E[R_X(U(s)) | R_X(U(s))]$$

$$= R_Y(s).$$

8.2.3.2.2 Step 2 Verify the claim of essentially one-dimensionality.

Our goal now is to represent the European option price on the basket $P_1 Y(T)$, $w$, in terms of a weighted average of European options, each of them written on the basket $P_1 X$. 
We have, recalling that $Y(t) = \exp(rt) \exp(-r \mathcal{U}(t)) X(\mathcal{U}(t))$,

$$w(t, y) = \exp(-r(T - t)) E[g(P_1 Y(T)) | Y(t) = y]$$

$$= \exp(-r(T - t)) E[E[g(P_1 Y(T)) | U(T), U(t), Y(t)] | Y(t) = y] \quad (8.29)$$

$$= \exp(-r(T - t)) E[\Pi | Y(t) = y]$$

with

$$\Pi = E[g(\exp(-r(U(T) - T)) P_1 X(U(T))) | U(T), U(t), X(U(t))]$$

being the price of a European option written on the basket $P_1 X$ with maturity time $U(T)$ and time to maturity $U(T) - U(t)$. Then, due to Lemma 8.2.7, $\Pi$ is essentially one dimensional and depends only on the basket value

$$P_1 X(U(t)) = \exp(-r(t - U(t))) P_1 y,$$

namely

$$\Pi = h(P_1 y, U(t) - t, U(T) - T). \quad (8.30)$$

The combination of (8.29) and (8.30) thus implies that

$$w(t, y) = \exp(-r(T - t)) E[h(P_1 y, U(t) - t, U(T) - T)],$$

meaning that $w$ only depends on $P_1 y$, which is what we wanted to prove. \qed

**Remark 8.2.9 (On the generality of the Stochastic Clock).** We note that in proving Corollary 8.2.8, we allow the stochastic clock $\mathcal{U}$ to be quite general.

However, we note that for stochastic clocks with discontinuous trajectories, the dynamics of $Y$ becomes discontinuous and thus the Gyöngy lemma no longer holds.
An example of $U$ with continuous trajectories is simply

$$dU(t) = \left( c + V^2(t) \right) dt,$$

$$U(0) = 0$$

both where $c > 0$ and $V$ is a one-dimensional Ornstein-Uhlenbeck process.

**Remark 8.2.10** (On the density of Bachelier model augmented by stochastic clock). In the preceding discussion above, we have assumed the density of the forward process to be known. For most choices of the stochastic clock process, this assumption will be violated. However, we still have access to the density conditioned on the value of the stochastic clock process. As a result, one may still evaluate the value of the projected volatility, introducing one additional quadrature and integrating over the possible values of the stochastic process.

**Remark 8.2.11** (Stochastic interest rates). For time dependent, stochastic interest rates independent of the price process, one may adopt essentially the same procedure as for the stochastic clock in Corollary 8.2.8, essentially averaging over possible values for the independent interest rate process.

For other models, such as the Black-Scholes model, there is no guarantee that Markovian projection method for pricing American basket options is exact. However, the similarity of the Black-Scholes and Bachelier models has been pointed out in the simpler European setting in earlier works by Teichmann and others. [20][38][41]

### 8.2.3.3 Other models in reduced dimension

We have demonstrated that the value function of an American basket option depends only on time and one state variable in the Bachelier model. Here, we present some particular cases in which this property holds for a more general stochastic model. We
first show that the reducibility in dimension is a phenomenon, that arises purely from
the dynamics of the system, not the early exercise property of the option.

Using this result, we characterize certain parametrizations of the Black-Scholes
model that reproduce the reduced dimension behavior familiar from the Bachelier
model discussed in the preceding section.

Lemma 8.2.12 (Decoupling of dimension reduction and early exercise). If a d-
dimensional SDE has a generator $\mathcal{L}$ that is essentially one dimensional, then the
corresponding backward operator, $\mathcal{H}$, for the American value function,

$$(\mathcal{H}v) (t, \mathbf{x}) = (\mathcal{L}v) (t, \mathbf{x}) \mathbf{1}_{\max((\mathcal{L}v)(t,\mathbf{x}), v(t,\mathbf{x}) - g(\mathbf{x})) > 0}$$

is essentially one dimensional.

Proof. First, define a coordinate rotation, $Q$, $Q^T Q = 1$, such that the portfolio value
is given by the first coordinate in the transformed coordinates $y = Qx$, with $Q$ chosen
so that the first row of $Q$ and $P_1^T$ are collinear. In these coordinates, denote the
Black-Scholes equation for the European value function as

$$-\partial_t u(t, y) = \mathcal{L}_y (t, y) u(t, y), \quad (t, Q^T y) \in [0, T] \times D,$$

$$u(T, y) = g(y_1), \quad Q^T y \in D.$$  \hfill (8.31)

To continue the proof, let us consider a Bermudan value function, $v_N$, with discrete
equispaced monitoring times, $t_j = \frac{jT}{N}$, $0 \leq j \leq N$, which solves [4]

$$-\partial_t v_N (t, y) = \mathcal{L}_y (t, y) v_N (t, y), \quad (t, Q^T y) \in (t_i, t_{i+1}) \times D, \quad 0 \leq i \leq N,$$

$$v_N (T, y) = g(y_1), \quad Q^T y \in D,$$

$$v_N (t_i, y) = \max \left( v_N (t_i^+, y), g(y_1) \right), \quad 0 \leq i \leq N, \quad Q^T y \in D. \hfill (8.32)$$

The terminal value $g(y_1)$ is essentially one dimensional, and by the assumption on
we know that \( v_N(t, y) \) is essentially one dimensional for \( t \in (t_{N-1}, t_N) \). Thus, the function \( v_N(t_{N-1}, y) \) is the maximum of two essentially one-dimensional functions that depend only on the \( y_1 \) coordinate. Therefore, we can conclude that

\[ \partial_{y_j} v_N(t_{N-1}, y) = 0, \quad j > 1, \quad Q^T y \in D \quad (8.33) \]

and, by using the same argument for all the subsequent intervals \( (t_{i-1}, t_i) \), we have that

\[ \partial_{y_j} v_N(t, y) = 0, \quad j > 1, \quad Q^T y \in D, \quad \forall t \in [0, T]. \quad (8.34) \]

The American option value function, \( v \), solves

\[ -\partial_t v(t, y) = \mathcal{H}_y(t, y) v(t, y), \quad (t, Q^T y) \in [0, T] \times D, \]
\[ v(T, y) = g(y_1), \quad Q^T y \in D, \]

where \( \mathcal{H}_y \) is the \( y \)-coordinate representation of the operator defined in \( (8.9) \). \( v \) is given as the limit of Bermudan value functions as the number of exercising times, \( N \), tends to infinity:

\[ v(t, y) = \lim_{N \to \infty} v_N(t, y), \quad (t, Q^T y) \in [0, T] \times D. \quad (8.35) \]

The combination of \( (8.34) \) and \( (8.35) \) yields

\[ \partial_{y_j} v(t, y) = 0, \quad j > 1, \quad Q^T y \in D, \quad \forall t \in [0, T], \]

which concludes the proof.

We have already seen that the Bachelier model is one example, in which the
Hamiltonian operator, $\mathcal{H}$, is essentially one-dimensional. Next, we proceed to other examples of stochastic models where the generator $\mathcal{L}$ is essentially one-dimensional, guaranteeing dimension reduction in the American option value function.

### 8.2.3.4 Black-Scholes model

Next, we turn our focus to the Black-Scholes model itself and examine how it behaves under Markovian projection and whether there exist parametrizations of the model that are essentially one dimensional.

First, let us state the relevant Black-Scholes PDE (8.6) corresponding to the Black-Scholes model:

$$\begin{align*}
-\partial_t w(t, x) &= -rw(t, x) + r \sum_i x_i \partial_{x_i} w(t, x) + \sum_{ij} \Omega_{ij} x_i x_j \partial_{x_i x_j} w(t, x), \quad t \in [0, T], \ x \in D_{BS}^d, \\
&\equiv (L_{BS} w)(t, x)
\end{align*}$$

(8.36)

where the symmetric matrix, $\Omega \in \mathbb{R}^{d \times d}$, is understood as the quadratic form corresponding to a volatility matrix, $\Sigma \in \mathbb{R}^{d \times k}$, of Equation (8.4), $\Omega = \Sigma \Sigma^T$. The domain is given as $D = D_{BS}^d = \mathbb{R}_+^d$.

**Remark 8.2.13.** A trivial example of a parametrization of the Black-Scholes model for which the value function is essentially one-dimensional is the case when portfolio weights vanish except for one, $P_1 = [1, 0, 0, \ldots, 0, 0]$. For such a portfolio, we can write a one-dimensional PDE describing the cost-to-go function.

For an arbitrary set of portfolio weights, $P_1$, of the Black-Scholes model Remark 8.2.13 certainly does not apply. However, we may apply a coordinate transformation to transform the portfolio weights to the particular choice in Remark 8.2.13. If the resulting transformed PDE is of the form (8.36), this is sufficient to show that the
value function is essentially one-dimensional.

Below, we demonstrate this and give a particular class of parametrizations, for which the transformation is possible. For other parametrizations, we note that these parametrizations can be approximated by ones where portfolio returns are log-normal. For a discussion of approximating the linear combination of variables from a multivariate log-normal, we refer the reader to [32].

We rotate the coordinates of the Black-Scholes equation (8.36) using the coordinate transformation, $Q$, from the proof of Lemma 8.2.12. We have

$$L_{BS,y} u(t,y) = -ru(t,y)$$
$$+ r \sum_{ikl} Q_{ki} Q_{il} y_k \partial_{y_i} u(t,y)$$
$$+ \sum_{ijklmn} \Omega_{ij} Q_{ki} Q_{lj} Q_{jm} Q_{in} y_k y_l \partial_{y_m y_n}^2 u(t,y), \quad t \in [0,T], \quad Q^T y \in D_{BS}^d.$$  

Thanks to the orthogonality of the transformation matrix $Q$, the first-order operator simplifies to

$$\sum_{ikl} Q_{ki} Q_{il} y_k \partial_{y_i} u(t,y) = \sum_i y_i \partial_{y_i} u(t,y).$$

However, the transformed second-order term does not take the form given in (8.36) in the general case. By writing in a tensorized form

$$\sum_{ijklmn} \Omega_{ij} Q_{ki} Q_{lj} Q_{jm} Q_{in} y_k y_l \partial_{y_m y_n}^2 u(t,y) = \Gamma_{klmn} y_k y_l \partial_{y_m y_n}^2 u(t,y)$$  \hspace{1cm} (8.37)

we have that $\Gamma$ has in general non-diagonal terms that couple $y_k$ and $y_l$ to $\partial_{y_m y_n}^2 u$ for $\{k,l\} \neq \{m,n\}$. Another way to write the second-order term is

$$\text{Tr} \left( \Omega \text{diag} \left( Q^T y \right) \left( Q^T (Hu) Q \right) \text{diag} \left( Q^T y \right) \right).$$
Using this notation, we give a particular example of a class of parametrizations of the Black-Scholes model for which the second-order term has the diagonal structure such that the generator $\mathcal{L}_{BS}$ is essentially one-dimensional.

**Corollary 8.2.14** (Effective one-dimensionality of Black-Scholes model when the quadratic form has equal elements). A Black-Scholes model such that the quadratic form in (8.36) satisfies $\Omega_{ij} = C$ for $1 \leq i, j \leq d$ has an essentially one-dimensional generator.

**Proof.** The proof is direct. Writing out the second-order term (8.37) we get

\[
\sum_{ijklmn} \Omega_{ij} Q_{ki} Q_{lj} Q_{jm} Q_{in} y_k y_l \partial_{y_m y_n}^2 u(t, y) = C \sum_{ijklmn} Q_{ki} \left( \sum_{j} Q_{lj} Q_{jm} \right) Q_{in} y_k y_l \partial_{y_m y_n}^2 u(t, y) = C \sum_{kl} \delta_{kn} y_k y_l \partial_{y_m y_n}^2 u(t, y) = C \sum_{kl} y_k y_l \partial_{y_m y_n}^2 u(t, y).
\]

We have demonstrated that there is a non-trivial set of parametrizations of the Black-Scholes model such that their corresponding generators $\mathcal{L}_{BS}$ are essentially one-dimensional.

For parametrizations that are not essentially one-dimensional, we still note that the upper and lower bounds (8.21) and (8.25) still hold. However, there is no a priori reason to believe that they coincide. In the next section, we evaluate the bound for a range of parametrizations and argue that these bounds are often close.
enough to get a practical estimate of the option price. This is expected due to the Multivariate Black-Scholes model being well approximated by an univariate Black-Scholes model on the one hand and the multivariate Bachelier model on the other. We have established above that the Markovian projection works for pricing in both the multivariate Bachlier model as well as the univariate Black-Scholes model. We demonstrate that this property carries over to the multivariate Black-Scholes model as a good approximation.

8.3 Numerical implementation

Here, we present a numerical implementation of our proposed method. First, we describe in Section 8.3.1 the methods used to evaluate the coefficients of the relevant PDE (8.18) in $D$. We briefly introduce the solution of the projected HJB equation in Section 8.3.2 and proceed in Section 8.3.3 to describe the evaluation of the lower and upper bounds using forward-Euler Monte Carlo simulation. We finally discuss the errors arising in the numerical methods in Section 8.3.4 and apply the proposed methods to Bachelier and Black-Scholes models of relevance in Section 8.3.5.

8.3.1 Evaluation of local volatility

So far, we have bypassed the issue of how to evaluate the local projected volatility $\bar{b}^{(x_0)}$ in (8.13). In this section we first describe in Section 8.3.1.1 how we may efficiently evaluate the high-dimensional integrals involved in the definition of the projected volatility $\bar{b}^{(x_0)}$. This discussion is followed by an interpolation scheme for extending pointwise evaluations of $\bar{b}^{(x_0)}$ into the projected domain $D$ in Section 8.3.1.2.

8.3.1.1 Laplace approximation

To approximate $u_A$ with $\bar{u}_A$, we must efficiently evaluate the conditional expectations (8.12) and (8.13) that involve high-dimensional integrals. For the risk-neutral case...
that is of most interest in financial applications and options pricing, the drift part will trivially project as

\[ \pi^{(x_0)}(t, s) = \mathbb{E}[\mathbf{P}_1 a(t, X(t)) | \mathbf{P}_1 X(t) = s, \ X(0) = x_0] \]
\[ = \mathbb{E}[\mathbf{P}_1 (rX(t)) | \mathbf{P}_1 X(t) = s, \ X(0) = x_0] \]
\[ = rs. \]

For the volatility, \( \tilde{b}^{(x_0)} \), we employ the Laplace approximation, by essentially finding an extremal point of the relevant unimodal integrands and applying a second-order approximation around that extremal point. Along this line, we make the following assumption.

**Assumption 8.3.1.** We assume that the transition density from \( x_0 \) to \( y \phi(y, x_0) \) \( \phi : \mathbb{R}^d \rightarrow \mathbb{R} \) corresponding to the process (8.1) is a smooth function for \( 0 \leq t \leq T \) and it is known explicitly.

The precise implementation of this approximation can be done in various ways, but the underlying principle remains the same. Some of these approaches allow to relax Assumption 8.3.1. Below we outline the Laplace approximation for the case where the assumption holds. For a more detailed account of the use of Laplace approximation, we refer the reader to [39] and [19].

Let

\[ \gamma(s) = \mathbb{E}[\Psi(X(t)) | \mathbf{P}_1 X(t) = s, \ X(0) = x_0], \]

with \( \Psi(X(t)) \in L^2(\mathbb{R}) \). Then, this conditional expectation satisfies

\[ \mathbb{E} [\Psi(X(t)) \theta(\mathbf{P}_1 X(t)) | X(0) = x_0] = \mathbb{E} [\gamma(\mathbf{P}_1 X(t)) \theta(\mathbf{P}_1 X(t)) | X(0) = x_0], \]

(8.38)
for all \( \theta \) such that \( \theta (P_1 \cdot) \in L^2 (\mathbb{R}) \). Taking in (8.38) \( \theta_h (x) = \frac{1}{h} 1_{2|x-s| < h} \) for \( h > 0 \) and letting \( h \to 0^+ \) the left-hand of the previous identity becomes a surface integral over a hyperplane

\[
\lim_{h \to 0^+} \mathbb{E} [\Psi (X (t)) \theta_h (P_1 X (t))] = \int_{P_1 x = s} \Psi (x) \phi (x; x_0) \, dA (x),
\]

where \( dA \) denotes the differential element of the hyperplane. For the right-hand side we have similarly

\[
\lim_{h \to 0^+} \mathbb{E} [\gamma (P_1 X (t)) \theta_h (P_1 X (t))] = \gamma (s) \int_{P_1 x = s} \phi (x; x_0) \, dA (x).
\]

Setting \( \Psi (\cdot) = P_1 b b^T P_1^T (t, \cdot) \) and solving for \( \gamma (s) \) in (8.38), we have

\[
\left( \frac{v^T (x_0)}{b^T (x_0)} \right)^2 (t, s) = \frac{\int_{\mathbb{R}^{d-1}} \phi (x (z); x_0) \left( P_1 b b^T P_1^T \right) (t, x (z)) \, dz}{\int_{\mathbb{R}^{d-1}} \phi (x (z); x_0) \, dz}, \tag{8.39}
\]

where we treat the first variable of \( x \) above as the dependent variable,

\[
x_i (z) = z_i, \quad \forall i > 1,
\]

\[
x_1 (z) = (P_{11})^{-1} \left( s - \sum_{j=2}^{d} P_{1j} z_j \right).
\]

Emphasizing that we work in \( \mathbb{R}^d \), rather than the possibly bounded domain \( D \), we approximate the integrals in (8.39), using Laplace approximation. We replace the unimodal integrands by suitable Gaussian functions centered at their maximizing configurations, \( z^* \in \mathbb{R}^{d-1} \) and \( z^* \in \mathbb{R}^{d-1} \).

Denoting the integrand by \( \exp (f) \) and exploiting the negative-definiteness of the Hessian \( \mathbf{H} f \), we may then approximate the integrand by expanding its logarithm \( f \)
as follows.

\[
\int_{\mathbb{R}^{d-1}} \exp (f(z)) \, dz \approx \int_{\mathbb{R}^{d-1}} \exp \left( f(z^*) + \frac{(z - z^*)^T (Hf) (z^*) (z - z^*)}{2} \right) \, dz
\]

\[
= \exp \left( f(z^*) \right) \sqrt{\frac{(2\pi)^{d-1}}{\det |(Hf)(z^*)|}}.
\]

(8.40)

We employ the same approximation for both the denominator and the numerator of (8.39) and get

\[
\frac{\int_{\mathbb{R}^{d-1}} \exp (f(z)) \, dz}{\int_{\mathbb{R}^{d-1}} \exp \left( \tilde{f}(z) \right) \, dz} \approx \exp \left( f(z^*) - \tilde{f}(z^*) \right) \sqrt{\frac{\det \left| \left( H\tilde{f} \right)(z^*) \right|}{\det \left| \left( Hf \right)(z^*) \right|}} \equiv \tilde{b}_1^2 (t, s),
\]

(8.41)

where

\[
\tilde{f}(z) = \log \left( \phi (x(z); x_0) \right)
\]

\[
f(z) = \log \left( \phi (x(z); x_0) \right) P_1 b b^T P_1^T (t, x(z))
\]

and \(z^*\) and \(z^*\) are the critical points for \(\tilde{f}\) and \(f\) respectively.

In practice, the critical configurations can be found rapidly by expanding the known integrand, \(f\), to second order and applying the Newton's iteration scheme,

\[
z^{(n+1)} = \left( Hf \left( z^{(n)} \right) \right)^{-1} \nabla f \left( z^{(n)} \right).
\]

(8.42)

The iteration quickly converges to an extremal point, typically within a few dozens of iterations allowing fast evaluation. Note that in the case of the Black-Scholes model, the density \(\phi\) contains a quadratic term, which makes the Newton iteration very robust to the choice of initial configuration \(z^{(0)}\) in (8.42).

We note that the approximation is rather simple for the case where the density of the process is normal or log-normal, i.e. the original process (8.1) corresponds to
Bachelier or Black-Scholes model. 5 consider the CEV model using the heat kernel approximation (see, for example, 42) for the transition density.

For numerical results on the accuracy of the Laplace approximation, we refer the reader to Appendix 8.A where the alternate choices of coordinates for the second-order expansion are discussed, along with their respective accuracies.

8.3.1.2 Extrapolation-interpolation to projected domain

To solve for the projected cost-to-go function, $\pi_A(t, s)$ in (8.18), we use the Laplace approximation introduced above to evaluate the projected local volatility in a few points in the domain, $\overline{D}$. We extend these values to a truncated domain in which we solve the low dimensional Equation (8.18). Thanks to the smooth behavior of the projected volatility, $\bar{b}(x_0)$, we only need a relatively low number of evaluations to achieve high accuracy.

However, to verify that the resulting projected cost-to-go function $\pi_A$ is indeed a good approximation of $u_A$ using the lower and upper bounds requires Monte Carlo simulation, which is typically costly compared to the solution of the projected backward problem (8.44).

To evaluate the projected volatility, $\bar{b}(x_0)$, we generate a small Monte Carlo forward-Euler sample of trajectories of the original process (8.1), as $X(t_n, \omega_i), 0 \leq t_n \leq N_t$ and $1 \leq i \leq M$ for $M \approx 100$, and to evaluate the essential support $[S^-(t_n), S^+(t_n)] \subset \overline{D}$ of the basket process that satisfies

\[
S^-(t_n) = \min_i P_1 X(t_n, \omega_i) \quad 0 \leq n \leq N_t, \quad 1 \leq i \leq M,
\]

\[
S^+(t_n) = \max_i P_1 X(t_n, \omega_i) \quad 0 \leq n \leq N_t, \quad 1 \leq i \leq M.
\]

We select a few dozen points equispaced in the intervals $[S^-(t_n), S^+(t_n)]$ for each time step $t_n$ and create a polynomial fit for $\bar{b}(x_0)$ for each of these instances of time.
(a) Third-order polynomial interpolation for the three-dimensional Black-Scholes model. Each red line corresponds to an instant of time from 0 to $T = \frac{1}{2}$ and is obtained through regression of a corresponding set of evaluations indicated through blue crosses.

(b) Local volatility for the projected dynamics in the high likelihood region of the 3-to-1 dimensional example (8.56). For the corresponding implied volatilities, see Figure 8.2(b).

Figure 8.1: Projected volatility $\tilde{b}_1(t, s)$ defined in (8.41) and its interpolation in space and time for the 3-to-1 dimensional Black-Scholes model (8.56). In both the figures, the plots are done for the range of essential support of the density, which expands as $t$ increases.

Remark 8.3.2. We note that the projected volatility can only be reliably evaluated inside the area where the density for $P_1X(t)$ is not negligible. At the most extreme case, at the initial time, the density of $P_1X(0)$ focuses on a single point. In reality, the appropriate domain for $\bar{D}$ has the schematic shape depicted in Figure 8.1(b).

However, we carry out our evaluation of $\pi A$ in a rectangular domain $[0, T] \times \bar{D}$ and extrapolate the local volatility into the whole rectangle. In carrying out the extrapolation, we set a small minimum value for $\left(\tilde{b}(x_0)\right)^2$ to guarantee numerical stability in the backward solver.

Note that the envelope (8.43) is only used to get a rough estimate of where the probability mass of $P_1X(t)$ for $0 \leq t \leq T$ lies and has a very indirect effect on the numerical solution as such. The resulting numbers of time steps $N_t$ and samples $M$ invested in (8.43) are small in comparison to the forward-Euler solution of the upper and lower bounds discussed later in Section 8.3.3.
8.3.2 Numerical value function

Once we define the interpolated-extrapolated approximate projected volatility \( \tilde{b} \) by interpolating the approximate projected volatility in (8.39), we set to define a finite-difference approximation \( \overline{u}_A \) of the value function \( u_A \) that solves (8.18). Based on the finite difference operator

\[
\left( \mathcal{L} \overline{u} \right) (t, s_n) = \left( \frac{\tilde{b}^2 (t, s_n)}{2\Delta s^2} + \frac{rs_n}{2\Delta s} \right) \overline{u} (t, s_{n-1}) - \left( r + \frac{\tilde{b}^2 (t, s_n)}{\Delta s^2} \right) \overline{u} (t, s_n) + \left( \frac{\tilde{b}^2 (t, s_n)}{2\Delta s^2} - \frac{rs_n}{2\Delta s} \right) \overline{u} (t, s_{n+1}),
\]

\( 1 < n < N_s \),

that parallels [?, Equation (12)] and whose continuous counterpart is \( \mathcal{L} \) of (8.16), we use a stable backward Euler scheme,

\[
\overline{u}_A (t_{n-1}, s_m) = \overline{u}_A (t_{n}, s_m) + \left( \mathcal{L} \overline{u} \right) (t_{n-1}, s_m) \Delta t_n, \quad 1 \leq n \leq N_t, \quad 1 \leq m \leq N_s,
\]

\[
\overline{u}_A (t_{n-1}, s_m) = \max \left( \overline{u}_A (t_{n-1}, s_m), g (s_m) \right), \quad 1 \leq n \leq N_t, \quad 1 \leq m \leq N_s,
\]

\[
\overline{u}_A (t_{N_t}, s_m) = g (s_m) \quad 1 \leq m \leq N_s, \tag{8.44}
\]

with the artificial Dirichlet-type boundary condition (see Remark 8.2.2) imposed by the payoff

\[
\overline{u}_A (t_n, s_1) = g (s_1), \tag{8.45}
\]

\[
\overline{u}_A (t_n, s_{N_s}) = g (s_{N_s})
\]

and a homogeneously spaced, time-independent, mesh \( s_m = m\Delta s \). The choice of the boundary condition has been discussed in the variational setting by [?, pp. 316]. The upper bound \( s_{N_s} \) has to be chosen based on the magnitude of the drift and the
volatility for the problem at hand.

The pointwise value function is later extended to the whole domain $\mathcal{D}$ of (8.36) using a low order interpolant, allowing the evaluation of a discrete early exercise region

$$
\mathcal{D}_{\text{Ex}} = \{(t_n, s_m) : 0 \leq n \leq N_T, \; 1 \leq m \leq N_s, \; \overline{u}_A(t_n, s_m) = g(s_m)\}
$$

(8.46)

Similarly, for the construction of the dual bound given by (8.26), we approximate derivatives of $\overline{u}_A$ (Eq. (8.36)) using finite differences of $\overline{u}_A$ (Eq. (8.34)).

### 8.3.3 Forward-Euler approximation

The discrete American put option value $\overline{u}_A$ that solves the backward-Euler scheme (8.44) implies a corresponding discrete early exercise region $\mathcal{D}_{\text{Ex}}$ of (8.46).

To verify the accuracy of the early exercise boundary implied by the discrete option value $\overline{u}_A$ as an approximation to the exercise boundary in $u_A$ and to set a confidence interval for the option price, we evaluate the lower and upper bounds in Equations (8.21) and (8.26) using Monte Carlo simulations based on (Forward) Euler-Maruyama. The numerical time-stepping for the asset prices, $X(t)$, is done on a uniform mesh. Setting the total number of time steps to coincide with the ones used in the finite difference approximation of $\overline{u}_A$ defined in (8.44), avoids the need for temporal interpolation of $\overline{u}_A$. As mentioned above, we use the following discretization of (8.1):

$$
\overline{X}(t_{n+1}) = \overline{X}(t_n) + r \overline{X}(t_n) \Delta t_n + b(t_n, \overline{X}(t_n)) \Delta W(t_n), \quad 0 \leq n < N_t,
$$

$$
\overline{X}(t_0) = x,
$$

(8.47)

with $\Delta t_n = t_{n+1} - t_n$ and $\Delta W(t_n) = W(t_{n+1}) - W(t_n) \sim \mathcal{N}(0, \Delta t_n)$ and the
number of time steps $N_t$. Correspondingly, we approximate (8.23) as

$$Z(t_n) = \exp(-rt_n) g\left(P_1 \overline{X}(t_n)\right), \quad 0 \leq n < N_t. \quad (8.48)$$

We use the same underlying Brownian motion to generate approximate trajectories for both the asset $\overline{X}$ and the approximation to the martingale $R$ in (8.26) used to construct the upper bound for the option price:

$$\overline{R}(t_{n+1}) = \overline{R}(t_n) + \exp(-rt_n) \left(\nabla u_A\right)^T(t_n, P_1 \overline{X}(t_n)) b(t_n, \overline{X}) \Delta W(t_n),$$

$$0 \leq n < N_t, \quad \overline{R}(0) = 0. \quad (8.49)$$

With the discrete approximations (8.47) and (8.49), we can estimate an upper bound, $A^+$, and a lower bound, $A^-$, for the option price, $u_A(0, X(0))$, using sample averages of $M$ i.i.d samples, namely

$$A^+_{M,N_t} = \frac{1}{M} \sum_{i=1}^{M} u^+ (\omega_i),$$

$$u^+ = \max_{0 \leq j \leq N_t} \left(\overline{Z}(t_j) - \overline{R}(t_j)\right) \quad (8.50)$$

and

$$A^-_{M,N_t} = \frac{1}{M} \sum_{i=1}^{M} u^- (\omega_i),$$

$$u^- = \exp\left(-r\tau\right) g\left(P_1 \overline{X}(\tau)\right),$$

$$\partial D_{Ex.}(t_n) = \max\{1 \leq m \leq N_s : (t_n, s_m) \in D_{Ex.}\}$$

$$\overline{\tau} = \min\{0 \leq j \leq N_t : P_1 \overline{X}(t_j) \leq \partial D_{Ex.}(t_n)\}. \quad (8.51)$$

To estimate the bias in the discretized approximations of the price bounds, we generate Monte Carlo samples corresponding to different values of $N_t$ and estimate the
difference between the resulting estimators, $|A_{M,2N_t}^+ - A_{M,N_t}^+|$ and $|A_{M,2N_t}^- - A_{M,N_t}^-|$. For a discussion on using the forward-Euler scheme for evaluating hitting times as the one in Equation (8.51), we refer the reader to [7,12].

In order to accelerate the computations of the bounds, we note the possibility of using multilevel estimators instead of those in (8.51) and (8.50) [17]. This is out of the scope of this work.

In Section 8.3.5, we present a selected set of test cases for which we evaluate the estimators (8.50) and (8.51). We focus in particular on the multivariate Black-Scholes that is both relevant and non-trivial and satisfies Assumption 8.3.1. The parametrizations of the Black-Scholes model we study do not feature essentially one-dimensional value functions and thus serve as a test case of our method when the accuracy of the method is not guaranteed a priori. Still, using the lower and upper bounds, we can analyze the accuracy of our method and verify its accuracy. For verification purposes, we include tests on the constant-volatility Bachelier model, for which the Markovian projection reproduces the American option prices exactly.

### 8.3.4 Error decomposition

Before proceeding further into the numerical examples we provide a brief summary of the errors incurred in the numerical solution of our price bounds, decomposing the total error into its constituent parts. Denoting the estimators of (8.50) and (8.51) as

$$A_{\infty,\infty}^\pm = \lim_{M,N_t \to \infty} A_{M,N_t}^\pm,$$

we have that the option price $u_A$ satisfies

$$A_{\infty,\infty}^- \leq u_A (0, x) \leq A_{\infty,\infty}^+.$$

In practice, we rely on estimators based on finite $M$ and $N_t$. The magnitude of the
gap \( |A_\infty^{+,\infty} - A_\infty^{-,\infty}| \) is dictated by the approximate value function \( \overline{u}_A \) that gives rise to the inexact stopping time (8.20) as well as the dual martingale \( M \). In general, finding an approximate function \( \overline{u}_A \) that approximates the true solution \( u_A \) closely might not be possible. Furthermore, even when a sound one-dimensional approximation \( \overline{u}_A \) exists, we rely on an approximate integration formula to recover it. Thus, for a general model, we are not able to control the error of our method and the magnitude of the gap \( |A_\infty^{+,\infty} - A_\infty^{-,\infty}| \). However, we are interested in choosing numerical parameters such that we get a reliable and useful estimate of the magnitude of this gap.

In addition to the gap between \( A_\infty^{+,\infty} \) and \( A_\infty^{-,\infty} \), the difference between \( A_\infty^{\pm,\infty} \) and the corresponding estimators \( A_{M,N}^{\pm,\infty} \) is of interest. Below, we outline the numerical approximations that give rise to these differences. Besides the fundamental error implied by approximating \( \tau^* \) of (8.19) with \( \tau^\dagger \) of (8.20), there are four main numerical approximations employed in the procedure, with each of them giving rise to a distinct component to the error. These are:

1. the statistical error due to finite number of samples, \( M \), in (8.50) and (8.51),

2. the step size bias introduced in the forward-Euler approximation (8.47),

3. the discretization errors of the solution \( \overline{u} \), giving rise to inexact approximations to the early-exercise region and the sensitivity in (8.49),

4. the Laplace approximation error when evaluating the integrals for the coefficients of the projected dynamics and the corresponding backward solution in (8.40).

Noting that the choice of the time-stepping scheme implies an optimal dependence between the number of temporal and spatial discretization steps, \( N_t \) and \( N_s \), and using the optimal \( N_s \), we expand the notation for the estimators \( A^- \) and \( A^+ \) to

\[
A_{M,N_t}^{\pm,\infty} = A_{M,N_t,N_s,\tilde{b}_t}^{\pm,\infty},
\]
(a) Convergence of the expected hitting time ($\tau_a$, green) to the early exercise region and the expected maximum ($X_{\text{max}}$, blue) over the interval $0 \leq t \leq \frac{1}{2}$ for a 3-dimensional correlated Black-Scholes model (8.56) along with the $N_{t}^{-\frac{1}{2}}$ reference line (dashed red).

(b) The implied volatility for the American put option corresponding to the local volatility of the projected 3-dimensional Black-Scholes model (8.56). Each of the values for $\sigma_{\text{imp}}$ produces the option prices for their respective strike price, $K$, for the American option price, when the local volatility is given by the projected dynamics.

Figure 8.2:

where the first $N_t$ refers to the number of forward-Euler time steps and the latter to the corresponding steps in the backward solver. With the triangle inequality, we decompose

$$\left| A_{\infty,\infty}^{\pm} - A_{M,N_{t}}^{\pm} \right| = \left| A_{\infty,\infty,\infty}^{\pm,\infty,\infty} - A_{M,N_{t},N_{t}}^{\pm,\infty,\infty} \right|$$

$$\leq \left| A_{\infty,\infty,\infty}^{\pm,\infty,\infty} - A_{\infty,\infty,\infty}^{\pm,\infty,\infty,\tilde{b}_{1}} \right| + \left| A_{\infty,\infty,\infty,\tilde{b}_{1}}^{\pm,\infty,\infty,\infty} - A_{\infty,\infty,\infty,\tilde{b}_{1}}^{\pm,\infty,\infty,\infty} \right|$$

$$+ \left| A_{\infty,\infty,\infty,\tilde{b}_{1}}^{\pm,\infty,\infty,\infty} - A_{\infty,\tilde{b}_{1}}^{\pm,\infty,\infty,\infty} \right| + \left| A_{\infty,\tilde{b}_{1}}^{\pm,\infty,\infty,\infty} - A_{\infty,\tilde{b}_{1}}^{\pm,\infty,\infty,\infty} \right| + \left| A_{\infty,\tilde{b}_{1}}^{\pm,\infty,\infty,\infty} - A_{\infty,\tilde{b}_{1}}^{\pm,\infty,\infty,\infty} \right|.$$

For the Laplace error $\left| A_{\infty,\infty,\infty}^{\pm,\infty,\infty} - A_{\infty,\infty,\infty,\tilde{b}_{1}}^{\pm,\infty,\infty,\infty} \right|$, there is no simple and practical way to control the error. We estimate the error through the numerical experiments as presented in the appendix 8.A. All the other components are well defined and can be controlled using standard arguments in their respective numerical methods. Firstly, with regard to the finite sample size, we can, given a confidence parameter, exploit the central limit theorem (CLT) and control the statistical error in probability by
increasing the sample size,

\[ |A^\pm_{\infty,N_t,N_t,\tilde{b}_1} - A^\pm_{M,N_t,N_t,\tilde{b}_1}| = \mathcal{O}_P\left(M^{-\frac{1}{2}}\right). \]  
(8.52)

As for the temporal discretization parameter, for the backward-Euler method, we set \( N_t \) in (8.44) to \( N_t^2 = cN_t \), giving rise to the discretization error,

\[ |A^\pm_{\infty,\infty,\tilde{b}_1} - A^\pm_{\infty,\infty,\tilde{b}_1}| = \mathcal{O}\left(N_t^{-1}\right). \]  
(8.53)

Finally, for the simulation of the extremal point of the dual martingale in (8.26) and the hitting time into the early exercise region implied by \( \bar{u}_A \), we have

\[ |A^\pm_{\infty,\infty,N_t,\tilde{b}_1} - A^\pm_{\infty,\infty,N_t,\tilde{b}_1}| = \mathcal{O}\left(N_t^{-\frac{1}{2}}\right), \]  
(8.54)

for each, as shown in Figure 8.2(a).

The novel contribution of this work is the use of the projected process for determining an implied exercise strategy for the true pricing problem (8.7) using the projected value function \( \bar{u}_A \) that solves (8.18). In the following sections, we wish to demonstrate the feasibility of this approach, and measure the resulting error, choosing parameters such that the errors (8.52), (8.53) and (8.54) are small compared to the error implied by the use of the surrogate process and its approximate evaluation using Laplace approximation. We proceed to do this in the following section.

\section*{8.3.5 Examples}

This section demonstrates the performance of our proposed method for pricing American put options written on a basket. First, we verify our results using a 50-dimensional Bachelier model in Section 8.3.5.1. Having verified that our numerical implementation reproduces the results expected based on Lemma 8.2.7, we proceed to apply the
(a) The upper $A_{128000,Nt}^+$ (Blue) and lower bound $A_{128000,Nt}^-$ (Green) for the American put price for varying numbers of time steps, $N_t$, in the forward-Euler discretization. Error bounds correspond to 95 percent confidence level. For the corresponding behavior of the relative width of the confidence interval, see Figure 8.3(b).

(b) The distance of the error bounds relative to the underlying option price for the at-the-money put for the test case presented in Figure 8.3(a). The estimate for the uncertainty is achieved as a combination of the upper and lower bounds presented in 8.3(a) together with an estimate of the statistical error and bias for both.

Figure 8.3: Convergence of the upper and lower bounds for the Bachelier model described in Section 8.3.5.1 and the resulting relative errors for the American and at-the-money put options.

method in multivariate Black-Scholes model in Sections 8.3.5.2-8.3.5.4.

8.3.5.1 American put on a basket in the Bachelier model

Here we wish to verify the numerical implementation of the finite difference solver for the approximate value function $\overline{u}_A$ of (8.44) and the resulting Monte Carlo estimators, (8.51) and (8.50), for the upper and lower bounds, respectively. We examine the solution of a 50-dimensional American put option in the Bachelier model (see Eqs. (8.2) and (8.3)). As our prime test case, we focus on the at-the-money put with maturity $T = \frac{1}{4}$. To guarantee a non-trivial early exercise region, we set a relatively high interest rate of $r = 0.05$. We choose an upper diagonal $\Sigma$ with the diagonal elements $\Sigma_{ii} = 20$ for all assets $1 \leq i \leq 50$ and draw the off-diagonal components $\Sigma_{ij}$, $j > i$ from a standard normal distribution.

Simulating the asset dynamics, $\overline{X}$, for a sequence of time discretizations, $N_t =$
1000 \times 2^k, 4 \leq k \leq 11, we observe that as $N_t$ increases, the difference between our upper and lower bounds for $u_{A}(0,x)$ becomes negligible. Figure 8.3(a) shows this behavior of converging bounds, alongside the statistical error of the upper bound estimator, $A^+$, which is far overshadowed by the corresponding statistical error from the lower bound estimator, $A^-$. Indeed, as the number of time steps in the forward simulation increases, we see the upper bound intersecting the confidence interval of
the lower bound, resulting in the sub-one-percent relative error of the method.

8.3.5.2 3-to-1 dimensional Black-Scholes model

As the first test parametrization of the Black-Scholes model we consider the case of a correlated 3-dimensional Black-Scholes model (see Eqs. (8.2) and (8.4)). We decompose the volatility function into the individual volatilities, $\sigma$, and the correlation structure of asset returns. We denote with $G$ the Cholesky decomposition of the correlation matrix of the log-returns

$$\Sigma_{ij} = \sigma_i G_{ij}. \quad (8.55)$$

We set the numerical parameters of our test case to

$$r = 0.05,$$

$$\sigma = (0.2, 0.15, 0.1)^T,$$

$$GG^T = \begin{pmatrix}
1 & 0.8 & 0.3 \\
0.8 & 1 & 0.1 \\
0.3 & 0.1 & 1
\end{pmatrix}, \quad (8.56)$$

and a portfolio of equally weighted assets

$$P_1 = [1, 1, 1], \quad (8.57)$$

as a representative test case of three moderately correlated assets in a high short rate environment. The projected local volatility features noticeable skew, as shown in Figures 8.1(b) and 8.2(b).
We evaluate the Laplace-approximated projected volatility, $\tilde{b}$, on a mesh of a few dozen nodes in the region where the density of the portfolio differs significantly from zero. Performing a regression to a third-degree polynomial on this mesh provides a close fit as seen in Figure 8.1(a). The third-order approximation also allows us to extend the evaluation of the projected volatility outside the domain in which the Laplace approximation is well behaved. Furthermore, the coefficients of the low-order polynomial fit to the projected volatility are well approximated by a constant, or a linear function of time. This means that for large times we can solve for the projected volatility $\tilde{b}^{(x_0)}$ particularly sparsely in time and still have an acceptable interpolation error.

To assess the accuracy of the method, we focus on a set of put options at $T = \frac{1}{2}$ with varying moneyness and report relative numerical accuracy in the approximation of around one percent. For the results of the prices and the corresponding relative errors, we refer to Figure 8.4.
Evaluating the American (green) and European (blue) option prices using forward-Euler Monte Carlo approximation and projected volatility based stopping rule and a martingale bound. At high strike, \( K \) we observe a trivial stopping time = 1.

Figure 8.4: Both European and American put option prices for the test case (8.56) and the corresponding relative errors. For comparison of the solvers, identical spatial and temporal meshes, sample sizes and number of Monte Carlo realizations are used for solving both the European and the American options.

(a) Finite-difference approximation to the American value function of the 3-to-1-dimensional projected problem (8.56). Note that the values of the value function are used to determine an early-exercise boundary only and have no real-world interpretation except at the point \( s = 300, t = 0 \).

(b) Numerical finite-difference approximation of early exercise boundary of the 3-to-1-dimensional projected problem (8.56) with maturity \( T = 0.5 \) for at-the-money put option. A slight kink at \( t < 0.05 \) resulting from the drop in projected volatility as seen in Figure 8.1 clearly visible.

Figure 8.5: The value function of the 3-to-1-dimensional Black-Scholes example (8.56) and the corresponding early exercise boundary.
233

8.3.5.3 10-to-1 dimensional Black-Scholes model

Next, we consider an example similar to (8.56), increasing the number of dimensions to ten. Continuing with the decomposition (8.55), we set

\[ r = 0.05, \]
\[ \sigma_i = 0.125, \quad 1 \leq i \leq 10, \]
\[ \mathbf{G G}^T = \begin{pmatrix}
1 & 0.2 & 0.2 & 0.35 & 0.2 & 0.25 & 0.2 & 0.2 & 0.3 & 0.2 \\
0.2 & 1 & 0.2 & 0.2 & 0.2 & 0.125 & 0.45 & 0.2 & 0.2 & 0.45 \\
0.2 & 0.2 & 1 & 0.2 & 0.2 & 0.2 & 0.2 & 0.2 & 0.45 & 0.2 \\
0.35 & 0.2 & 0.2 & 1 & 0.2 & 0.2 & 0.2 & 0.2 & 0.425 & 0.2 \\
0.25 & 0.125 & 0.2 & 0.2 & 1 & 0.2 & 0.2 & 0.5 & 0.35 & 0.2 \\
0.2 & 0.45 & 0.2 & 0.2 & 0.2 & 1 & 0.2 & 0.2 & 0.2 & 0.2 \\
0.2 & 0.45 & 0.2 & 0.2 & 0.2 & 0.2 & 1 & 0.2 & 0.2 & 0.2 \\
0.2 & 0.2 & 0.2 & 0.2 & 0.2 & 0.2 & 0.2 & 1 & 0.2 & -0.1 \\
0.3 & 0.2 & 0.45 & 0.425 & 0.5 & 0.35 & 0.2 & 0.2 & 1 & 0.2 \\
0.2 & 0.45 & 0.2 & 0.2 & 0.2 & 0.2 & 0.2 & -0.1 & 0.2 & 1 
\end{pmatrix} \] (8.58)

We evaluate a sequence of put options with varying moneyness for the equally weighted portfolio of assets namely we set \( P_{1i} = 1 \), for all indices. As before, we observe a relative accuracy of a few percent, with decreasing relative error as moneyness increases. As in the previous case, with extreme moneyness, we notice the tendency for an exercise at the initial time, resulting in a variance drop of the estimators and subsequently the relative error, as shown in Figure 8.6. The behavior of the price uncertainty of the American and European options in the 10-dimensional case, as a function of the number of time steps, \( N_t \), is illustrated in Figure 8.7(b).
(a) European (Blue) and American (Green) put option prices for the 10-dimensional Black-Scholes test case (8.58) at $T = 0.5$, using forward-Euler Monte Carlo approximation and projected volatility based stopping rule and a martingale bound.

(b) Relative errors in evaluating the American (green) and European (blue) option prices for the 10-dimensional Black-Scholes model (8.58) using forward-Euler Monte Carlo approximation with varying ranges of moneyness.

Figure 8.6: Convergence of the upper and lower bounds for the 10-to-1 dimensional Black-Scholes model (8.58) and the resulting relative errors. As in Figure 8.4 same numerical parameters have been used for solving both the European and the American prices for ease of comparison.

(a) Price uncertainty as a function of the number of time steps $N_t$ for the 10-dimensional Black-Scholes model for the American option (Blue) and the corresponding European option (Green).

(b) Relative error in estimating the American (Blue) and the European (Green) option price.

Figure 8.7: Price uncertainty in the 10-dimensional Black-Scholes model (8.58) with varying numbers of time steps $N_t$ in the forward-Euler monte Carlo.
Finally, we consider a case with a high dimension that is certainly beyond the reach of most PDE solvers. We choose the 25-dimensional GBM considered by [6]. For the remaining parameters, we set

\[ X_i(0) = 100, \quad i \in \{1, 2, \ldots, 25\} \]
\[ r = 0.05, \quad (8.59) \]

and evaluate the options with equal portfolio weights, \( P_{1i} = 1, \ i \in \{1, 2, \ldots, 25\} \).

With the 25-dimensional model, we continue to observe numerical performance of a few percent of relative errors with the projected stopping rule for basket put options of maturity \( T = \frac{1}{2} \) as well as a significant early-exercise premium clearly exceeding the accuracy of the method. Results for the option price estimates for the American and European options and the corresponding error bounds are presented in Figures

(a) European (Blue) and American (Green) put option prices for the 25-dimensional Black-Scholes test case at \( T = 0.5 \) of Section 8.3.5.4, using forward-Euler Monte Carlo approximation and projected volatility based stopping rule and a martingale bound.

(b) Relative errors in evaluating the American (green) and European (blue) option prices for the 25-dimensional Black-Scholes model using forward-Euler Monte Carlo approximation for varying ranges of moneyness. As in earlier Figures 8.4 and 8.6 identical numerical parameters are used for both the American and European options.

Figure 8.8: Convergence of the upper and lower bounds for the 25-to-1-dimensional Black-Scholes model and the resulting relative errors.
Figure 8.9: American put prices (8.9(a)) and corresponding relative errors (8.9(b)) for 43 independent randomized repetitions on evaluating the American put on the 25-to-1-dimensional Black-Scholes model with 258 individual option price valuations for varying strike, $K$. In addition to the random structure of the test problem by [6] and the parameters (8.59) and $T = \frac{1}{4}$, we also randomize the portfolio weights. For each of the runs, we choose $P_{1i}$ independently from an uniform distribution $U[\frac{1}{2}, \frac{3}{2}]$ and finally rescale the weights so that $\sum_{i=1}^{25} P_{1i} = 25$.

8.8(a) and 8.8(b) respectively. To demonstrate the consistency and robustness of our approach towards the particular choice of parameters, we replicate the runs multiple times with various portfolio weights. The results of these repeated trials are illustrated in Figure 8.9.

We note that even though we have not proven asymptotic convergence for a general multivariate model, the approximation of the true problem with the one-dimensional stopping rule gives consistently results that are comparable to the bid-ask spread of the most liquid American index options, and well below those of less-liquid regional indices and ETFs tracking them. We also note that the relative accuracy for the American put price is greatest in the crucial region of in-the-money, where the violation of the put-call parity is most profound.
8.4 Conclusions

In this work, we have demonstrated the practicability of using Markovian Projection in the framework of pricing American options written on a basket. In the implementation of the numerical examples, we have exploited the explicitly known density of the Black-Scholes model, as well as the specific structure of the Bachelier model. Using the known density, we devised a Laplace approximation to evaluate the volatility of a Markovian projection process that describes the projected and approximate dynamics of the basket.

We have shown that for the Bachelier model the Markovian projection gives rise to exact projected option prices, even when considering options with path-dependence. We have also demonstrated how the vanishing derivatives of the cost-to-go function are a manifest of the process dynamics, not the early exercise nature of the option. Leveraging this result, we have demonstrated the existence of nontrivial characterisations of the Black-Scholes model that are essentially of low dimension.

Using the Markovian projection in conjunction with the Laplace approximation, we have implemented low-dimensional approximations of various parametrizations of the multivariate Black-Scholes model. With numerical experiments, we have shown that these approximations perform surprisingly well in evaluating prices of American options written on a basket. We interpret these results as a manifestation of the Black-Scholes model being well approximated by a corresponding Bachelier model. What sets these results apart from many of the earlier works is the fact that we approximate the full trajectory of a basket of assets in the Black-Scholes model, not only instantaneous returns.

The primary method used to solve such problems so far has been the least-squares Monte Carlo method that shares some common attributes with our proposed method. Unlike least-squares Monte Carlo, our proposed method does not rely on a choice of basis vectors that are used to evaluate the holding price of an option, but only on the
direction or directions along which we evaluate the projected dynamics.

Our results leave the door open for future developments including the extension of the current research into models beyond the GBM model. We validate the accuracy of our stopping rule using a forward-simulation. One possible extension of this work would be to use the forward sample also to evaluate the projected volatilities, an approach used in calibration of correlation structures by [21]. As the only non-controlled error in our method is the bias incurred in evaluating the local volatility $\tilde{\sigma}^{(x_0)}$, the possibility to implement such an evaluation efficiently but without introducing bias would be very useful. From a theoretical viewpoint, our work raises the question of whether the approximation improves if the projection dimension is increased.

In this work, we have not aimed to demonstrate the use of Markovian-projected models for evaluating implied stopping times. In doing so, we have not aimed for the greatest possible computational efficiency, and many possibilities for further optimization exist in this area. In terms of orders of convergence, the bottleneck of the computation is the forward Euler simulation and subsequent evaluation of maxima and hitting times of realizations of an SDE. These Monte Carlo methods could be enhanced through adaptivity, multi-level methods, use of quasi-Monte Carlo [?], or analytic approximations. Likewise, there is a possibility for optimization of the numerical solver to evaluate the value function using a highly optimized backward solver [28]. For the possibility of extending the projection to higher dimensions to allow for higher-dimensional approximation of the early exercise boundary, we refer reader to [23]. We also note the possibility of using a binomial tree method [26], that naturally takes into account the shape of the domain $\overline{D}$ for the projected PDE.

We have focused on the commercially most relevant application of American options that are widely quoted on the market. For the case of binary options the analysis remains identical, only the functional form of the payoff $g$ changes. It would also be of interest to study the performance of the Markovian-projected dynamics in pricing
other path-dependent options such as Asian and knockoff options. Study of more general payoff functions is possible, assuming the projected volatility corresponding to these state variables could be efficiently evaluated.

8.5 Acknowledgements

We thank Professors Ernesto Mordecki and Fabián Croce for their feedback which significantly improved this manuscript. Gillis Danielsen provided much-valued practitioner’s views.
Bibliography


The Taylor expansion of the integrands in equation (8.39) can be done in various ways, and we discuss and illustrate some natural choices here. For the test case, let
us consider the equal-volatility, equal weight non-correlated two-dimensional Black-Scholes model with \( r = 0 \) and

\[
P_1 = [1, 1],
\]

\[
\Sigma = \text{diag}\left( [\sigma, \sigma]^T \right),
\]

\[
X(0) = [100, 100]^T,
\]

with the volatility, \( \sigma = 0.1 \). For such a simple test case, we can evaluate the relevant expansion by hand. For a high-dimensional model, we need to resort to quadratures or Monte Carlo.

Fixing the portfolio value to \( P_1X(0) \), the relevant unimodal integrands in terms of the natural price of the second asset \( s_2 \) are given as

\[
f_1(s_2) = \frac{\exp\left( -\left( \frac{\log(2 - \frac{s_2^{100}}{2\sigma^2})}{2\sigma^2} \right)^2 - \left( \frac{\log(s_2^{100})}{2\sigma^2} \right)^2 + 2\log\sigma + \log(s_2^2 - 400s^2 + 40000) - \log(200 - s_2) - \log(s_2) \right)}{2\pi\sigma^2}
\]

(8.60) for the numerator and

\[
\hat{f}_1(s_2) = \frac{\exp\left( -\left( \frac{\log(2 - \frac{s_2}{2\sigma^2})}{2\sigma^2} \right)^2 - \left( \frac{\log(s_2^{100})}{2\sigma^2} \right)^2 - \log(200 - s_2) - \log(s_2) \right)}{2\pi\sigma^2}
\]

(8.61) for the denominator. Alternatively, we can express the integrals in terms of log-price

\[
x_2 = \log(s_2^{100}),
\]

\[
f_2(x_2) = \frac{-\left( \frac{\log(2 - e^{x_2})}{2\sigma^2} \right)^2 - \frac{x_2^2}{2\sigma^2} + 2\log\sigma + \log(2e^{2x_2} - 4e^{x_2} + 4) - \log(2 - e^{x_2}) - x_2}{2\pi\sigma^2}
\]

(8.62)
for the numerator and

$$f_2(x_2) = - \frac{(\log(2 - e^{x_2}))^2 - x_2^2}{2\sigma^2} - \frac{\log(2 - e^{x_2}) - x_2}{2\pi\sigma^2}$$

for the denominator. With these definitions we have the unit-time projected volatility

$$\left( \tilde{b}^{(\pi)} \right)^2 (1, 200) = \frac{\int_R f_1(s_2) \, ds_2}{\int_R f_1(s_2) \, ds_2} = \frac{\int_R f_2(x_2) \, dx_2}{\int_R f_2(x_2) \, dx_2}.$$

The integrands $f_1$, $\tilde{f}_1$ and their respective second-order approximations of the form $\exp(\eta + \kappa (z_2 - z^*)^2)$ are illustrated in Figure 8.10 for the price expansion and in Figure 8.11 log-price respectively.
Figure 8.11: Functions $f_2$ of (8.63) (8.11(a)) and $\tilde{f}_2$ of (8.62) (8.11(b)) in blue and and their respective approximations based on the second-order Taylor expansions of their logarithms in dashed red.

The approximations are given as

\begin{align*}
2\pi\sigma^2 f_1(s_2) &\approx \exp\left(2\log\sigma - \left(\frac{1}{100^2\sigma^2} + \frac{2}{100^2}\right)(s_2 - 100)^2\right), \\
2\pi\sigma^2 \tilde{f}_1(s_2) &\approx \exp\left(-2\log200 - \left(\frac{1}{100^2\sigma^2} + \frac{1}{100^2}\right)(s_2 - 100)^2\right), \\
2\pi\sigma^2 f_2(x_2) &\approx \exp\left(2\log\sigma + \log200 - \left(\frac{1}{\sigma^2} + 2\right)x_2^2\right), \\
2\pi\sigma^2 \tilde{f}_2(x_2) &\approx \exp\left(-\left(\frac{1}{\sigma^2} + 1\right)x_2^2\right),
\end{align*}

giving for both approximations

$$
\tilde{b}^2_1(1,100) = \tilde{b}^2_2(1,100) = 20000\sigma^2\sqrt{\frac{1+2\sigma^2}{1+\sigma^2}} \approx 200.99.
$$

In contrast, with quadrature, we get a reference value of 200.98, giving a close agreement with the Laplace-approximated value.