Spectral Element Method for Pricing

European Options and Their Greeks

by

Tianyao Yue

Department of Electrical and Computer Engineering
Duke University

Date: __________________________
Approved:

________________________________________
Qing H. Liu, Supervisor

________________________________________
A. Craig Burnside

________________________________________
Anita T. Layton

________________________________________
Xuejun Liao

________________________________________
Yaroslav A. Urzhumov

________________________________________
Gary A. Ybarra

Dissertation submitted in partial fulfillment of the requirements for the degree of
Doctor of Philosophy in the Department of Electrical and Computer
Engineering in the Graduate School of Duke University

2012
Abstract

Spectral Element Method for Pricing European Options and Their Greeks

by

Tianyao Yue

Department of Electrical and Computer Engineering
Duke University

Date: ______________________

Approved:

__________________________
Qing H. Liu, Supervisor

__________________________
A. Craig Burnside

__________________________
Anita T. Layton

__________________________
Xuejun Liao

__________________________
Yaroslav A. Urzhumov

__________________________
Gary A. Ybarra

An abstract of a dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Electrical and Computer Engineering in the Graduate School of Duke University

2012
Abstract

Numerical methods such as Monte Carlo method (MCM), finite difference method (FDM) and finite element method (FEM) have been successfully implemented to solve financial partial differential equations (PDEs). Sophisticated computational algorithms are strongly desired to further improve accuracy and efficiency.

The relatively new spectral element method (SEM) combines the exponential convergence of spectral method and the geometric flexibility of FEM. This dissertation carefully investigates SEM on the pricing of European options and their Greeks (Delta, Gamma and Theta). The essential techniques, Gauss quadrature rules, are thoroughly discussed and developed. The spectral element method and its error analysis are briefly introduced first and expanded in details afterwards.

Multi-element spectral element method (ME-SEM) for the Black-Scholes PDE is derived on European put options with and without dividend and on a condor option with a more complicated payoff. Under the same Crank-Nicolson approach for the time integration, the SEM shows significant accuracy increase and time cost reduction over the FDM. A novel discontinuous payoff spectral element method (DP-SEM) is invented and numerically validated on a European binary put option. The SEM is also applied to the constant elasticity of variance (CEV) model and verified with the MCM and the valuation formula. The Stochastic Alpha Beta Rho (SABR) model is solved with multi-dimensional spectral element method (MD-SEM) on a European put option. Error convergence for option prices and Greeks with respect to the number of grid points and the time step is analyzed and illustrated.
To the Memory of Louis Jean-Baptiste Alphonse Bachelier (1870-1946)
# Contents

Abstract iv  
List of Tables x  
List of Figures xi  
List of Abbreviations and Symbols xxii  
Acknowledgements xxvi  

1 Introduction 1  
  1.1 Option Theory History 1  
  1.2 Numerical Methodologies 3  
  1.3 Innovative Contributions 6  
  1.4 Outline of the Dissertation 8  

2 The Black-Scholes Model 9  
  2.1 Derivation of PDE 9  
  2.2 Heat Transformation 12  
  2.3 Closed-Form Solution 12  
  2.4 Option Sensitivities (Greeks) 14  
  2.5 Black-Scholes with Dividends 15  
  2.6 An Illustrative Option Example 16  

3 Gauss Quadrature Method 21  
  3.1 Riemann Integral 21
3.2 Newton-Cotes Formula ...................................................... 22
3.3 Gauss Quadrature Rule .................................................... 24
  3.3.1 Orthogonal Polynomials ............................................ 24
  3.3.2 General Gauss Quadrature ......................................... 31
  3.3.3 Calculations of Gauss Quadrature ................................. 35
3.4 Classical Gauss Quadratures ........................................... 41
  3.4.1 Gauss-Chebyshev Quadrature ...................................... 41
  3.4.2 Gauss-Hermite Quadrature ......................................... 46
  3.4.3 Stable Gauss-Hermite Quadrature .................................. 49
  3.4.4 Gauss-Laguerre Quadrature ........................................ 51
  3.4.5 Stable Gauss-Laguerre Quadrature ................................. 54
  3.4.6 Gauss-Legendre Quadrature ........................................ 56
3.5 Radau and Lobatto Quadratures ........................................ 60
  3.5.1 Proof of Gauss-Radau Quadrature ................................. 61
  3.5.2 Gauss-Radau-Laguerre Quadrature ................................ 67
  3.5.3 Stable Gauss-Radau-Laguerre Quadrature ......................... 69
  3.5.4 Proof of Gauss-Lobatto Quadrature .............................. 70
  3.5.5 Gauss-Lobatto-Legendre Quadrature ............................. 80
3.6 Orthogonal Polynomial Basis Expansion ............................. 83
  3.6.1 Gauss-Hermite Basis Function .................................... 83
  3.6.2 Stable Gauss-Hermite Basis Function ............................ 86
  3.6.3 Gauss-Radau-Laguerre Basis Function ............................ 90
  3.6.4 Stable Gauss-Radau-Laguerre Basis Function .................... 94
  3.6.5 Gauss-Lobatto-Legendre Basis Function ......................... 98
3.7 Other Related Topics for Gauss Quadratures ........................ 104
### 3.7.1 The Newton-Raphson Method

104

### 3.7.2 Scaling Factors and Domain Transformations

105

### 3.7.3 Computations of First and Second Derivatives

108

### 3.8 Numerical Results for Gauss Quadrature Method

109

---

### 4 Introduction to Spectral Element Method

117

#### 4.1 Historical Backgrounds

117

#### 4.2 Modern Approaches

118

#### 4.3 Simple One-Element Example

119

#### 4.4 Error and Convergence Analysis

123

### 5 Spectral Element Method on Black-Scholes

125

#### 5.1 One-Element SEM on Black-Scholes PDE

126

- 5.1.1 Weak Formulation

126

- 5.1.2 Time Discretization

130

- 5.1.3 SEM Option Greeks

131

- 5.1.4 Computation Result

132

#### 5.2 Multi-Element SEM on Black-Scholes PDE

135

- 5.2.1 Weak Formulation

135

- 5.2.2 Time Discretization

146

- 5.2.3 SEM Option Greeks

146

- 5.2.4 Results for Put Prices

147

- 5.2.5 Results for Put Greeks

151

#### 5.3 Comparison with Finite Difference Method

159

#### 5.4 SEM on Black-Scholes PDE with Dividends

162

#### 5.5 SEM on A European Condor Spread Option

173

---
6 Spectral Element on European Binary Option 183

6.1 Introduction and Challenges ............................. 183
6.2 Approximations and Results .............................. 186
  6.2.1 Approximations ...................................... 186
  6.2.2 Numerical Results .................................... 187
6.3 DP-SEM Method and Results .............................. 195
  6.3.1 The Final Condition ................................. 195
  6.3.2 SEM Option Greeks ................................. 201
  6.3.3 Numerical Results .................................... 201
6.4 Comparison of Methods ................................. 211

7 SEM on Constant Elasticity of Variance (CEV) 213

7.1 Model Introduction ...................................... 213
7.2 Derivation of PDE ....................................... 214
7.3 SEM on CEV PDE ......................................... 216
7.4 Numerical Results ....................................... 219

8 The Stochastic Alpha Beta Rho (SABR) Model 225

8.1 Model Introduction ...................................... 225
8.2 Illustrative Example ..................................... 227
8.3 Derivation of PDE ....................................... 230
8.4 MD-SEM on SABR ......................................... 231
8.5 Numerical Results ....................................... 234

9 Conclusion, Discussion and Further Work 237

Bibliography 239

Biography 246
List of Tables

2.1 Commonly Used Greeks for European Option .................. 14
2.2 Common Greeks for European Option on Stock with Dividends ... 16
3.1 Classical orthogonal polynomials and weighting functions .......... 31
5.1 Errors for the European Put Option with Dividend ............... 163
5.2 Errors for the European Condor Spread Option .................. 174
6.1 Option Greeks for European Binary Put .......................... 185
List of Figures

2.1 Call option price surface for $K = 50$, $r = 10\%$, and $\sigma = 50\%$ . . . . . 17
2.2 Put option price surface for $K = 50$, $r = 10\%$, and $\sigma = 50\%$ . . . . . 17
2.3 Call option Delta surface for $K = 50$, $r = 10\%$, and $\sigma = 50\%$ . . . . . 18
2.4 Put option Delta surface for $K = 50$, $r = 10\%$, and $\sigma = 50\%$ . . . . . 18
2.5 Option Gamma surface for $K = 50$, $r = 10\%$, and $\sigma = 50\%$ . . . . . 19
2.6 Option Vega surface for $K = 50$, $r = 10\%$, and $\sigma = 50\%$ . . . . . 19
2.7 Call Theta surface for $K = 50$, $r = 10\%$, and $\sigma = 50\%$ . . . . . 20
2.8 Put Theta surface for $K = 50$, $r = 10\%$, and $\sigma = 50\%$ . . . . . 20
3.1 Modified Hermite polynomials $\mathcal{H}_n(x)$, $n = 0, \ldots, 3$ . . . . . . . . . . 50
3.2 Modified Laguerre polynomials $\mathcal{L}_n(x)$, $n = 0, \ldots, 3$ . . . . . . . . . . 55
3.3 Sample Legendre polynomials $P_n(x)$, $n = 0, \ldots, 4$ . . . . . . . . . . . . . . . 60
3.4 Modified Gauss-Hermite basis function $\psi_i(x)$, $i = 0, 1, 2$ . . . . . . . . . . 88
3.5 Modified Gauss-Radau-Laguerre basis function $\psi_i(x)$, $i = 0, 1, 2$ . . . . 96
3.6 Gauss-Lobatto-Legendre basis function $\phi_i(x)$, $i = 0, \ldots, 3$ . . . . . . . . 99
3.7 $f(x)$ and its approximates expanded by stable Gauss-Hermite bases. . 109
3.8 $f'(x)$ and its approximated values using stable Gauss-Hermite bases. . 110
3.9 $f''(x)$ and its approximated values using stable Gauss-Hermite bases. 110
3.10 Comparison between the eigenvalue method alone and the combina-
tion method for the function $f(x)$ using stable Gauss-Hermite basis
expansion. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 111
3.11 Comparison between the eigenvalue method and combination method for the first derivative $f'(x)$ using stable Gauss-Hermite basis function expansion. .................................................. 112

3.12 Comparison between the eigenvalue method and combination method for the second derivative $f''(x)$ using stable Gauss-Hermite basis function expansion. .................................................. 112

3.13 $f(x)$ and its approximates expanded by stable GRL basis functions. . 113

3.14 $f'(x)$ and its approximated values using stable GRL basis functions. . 113

3.15 $f''(x)$ and its approximated values using stable GRL basis functions. . 114

3.16 Comparison between the eigenvalue method alone and the combination method for the function $f(x)$ using stable GRL basis function expansion. .................................................. 114

3.17 $f(x)$ and its approximates expanded by GLL basis functions. . . . . 115

3.18 $f'(x)$ and its approximated values using GLL basis functions. . . . . 115

3.19 $f''(x)$ and its approximated values using GLL basis functions. . . . . 116

3.20 Comparison between the eigenvalue method alone and the combination method for the function $f(x)$ using Gauss-Lobatto-Legendre basis expansion. .................................................. 116

5.1 The standard European put option prices over time-to-maturity and stock price using one-element spectral element method with stable GRL quadrature. .................................................. 133

5.2 Comparison between analytical Black-Scholes prices and one-element spectral element method prices for the standard European put option at inception. .................................................. 133

5.3 The $L_2$-Errors for the standard European put option prices with respect to the total number of grid points using one-element method with $\delta t \approx 10^{-5}$ year. .................................................. 134

5.4 The $L_2$-Errors for the standard European put option prices with respect to time step using the one-element method with the total number of grid points = 61. .................................................. 134

5.5 A multi-element basis function with the GLL basis function on the left with order $n^a = 3$ and the stable GRL basis function on the right with order $n^b = 3$. .................................................. 138
5.6 The standard European put option prices over time-to-maturity and stock price using two-element spectral element method with GLL and stable GRL. ................................................................. 149

5.7 Comparison between analytical Black-Scholes prices and two-element spectral element method prices for the standard European put option at inception. ................................................................. 149

5.8 The $L_2$-Errors for the standard European put option prices with respect to the total number of grid points using two-element method with $\delta t = 10^{-5}$ year. ................................................................. 150

5.9 The $L_2$-Errors for the standard European put option prices with respect to time step using the two-element method with the total number of grid points = 61. ................................................................. 150

5.10 Comparison between analytical Black-Scholes Deltas and two-element spectral element method Deltas for the standard European put option at inception. ................................................................. 152

5.11 The $L_2$-Errors for the standard European put Deltas with respect to the total number of grid points using the two-element method with $\delta t = 10^{-5}$ year. ................................................................. 152

5.12 The $L_2$-Errors for the standard European put Deltas with respect to time step using the two-element method with the total number of grid points = 61. ................................................................. 153

5.13 Comparison between the analytical Black-Scholes Gammas and from the two-element spectral element method for the European put option at inception. ................................................................. 154

5.14 The $L_2$-Errors for the standard European put Gammas with respect to the total number of grid points using two-element method with $\delta t = 10^{-5}$ year. ................................................................. 154

5.15 The $L_2$-Errors for the standard European put Gammas with respect to time step using two-element method with the total number of grid points = 61. ................................................................. 155

5.16 The $L_2$-Errors for the put Gammas of stock prices no less than $1 with respect to total number of grid points using two-element with $\delta t = 10^{-5}$ year. ................................................................. 156
5.17 The $L_2$-Errors for the put Gammas of stock prices no less than $1$ with respect to time step using two-element with total number of grid points = 61. .................................................. 156

5.18 Comparison between analytical Black-Scholes Thetas and two-element spectral element method Thetas for the standard European put option at inception. .................................................. 157

5.19 The $L_2$-Errors for the standard European put Thetas with respect to the total number of grid points using the two-element method with $\delta t \doteq 10^{-5}$ year. .................................................. 158

5.20 The $L_2$-Errors for the standard European put Thetas with respect to time step using the two-element method with the total number of grid points = 61. .................................................. 158

5.21 Comparison of $L_2$-Errors for the standard European put price with respect to the total number of grid points using FDM and SEM with $\delta t \doteq 10^{-5}$ year. .................................................. 160

5.22 Comparison of computational time for the standard put option with respect to the total number of grid points using FDM and SEM with $\delta t \doteq 10^{-5}$ year. .................................................. 160

5.23 Comparison of $L_2$-Errors for the standard European put price with respect to the total number of grid points using FDM and SEM with $\delta t \doteq 10^{-5}$ year. .................................................. 161

5.24 Comparison of computational time for the standard put option with respect to the total number of grid points using FDM and SEM with $\delta t \doteq 10^{-5}$ year. .................................................. 161

5.25 The standard European put option prices over time-to-maturity and stock price which has constant dividend using two-element spectral element method. .................................................. 164

5.26 Comparison between analytical Black-Scholes prices and two-element spectral element method prices for the standard European put option with dividend. .................................................. 164

5.27 The $L_2$-Errors for the standard European put option with dividend with respect to the total number of grid points using the SEM with $\delta t \doteq 10^{-5}$ year. .................................................. 165
5.28 The $L_2$-Errors for the standard European put option with dividend with respect to time step using the SEM with the total number of grid points = 61. ................................. 165

5.29 Comparison between analytical Black-Scholes Deltas and two-element spectral element method Deltas for the standard European put option with dividend. ................................. 166

5.30 The $L_2$-Errors for the standard European put Deltas with dividend with respect to the total number of grid points using the SEM with $\delta t = 10^{-5}$ year. ................................. 166

5.31 The $L_2$-Errors for the standard European put Deltas with dividend with respect to time step using the SEM with the total number of grid points = 61. ................................. 167

5.32 Comparison between the analytical Black-Scholes Gammas and from the two-element spectral element method for the put option with constant dividend. ................................. 167

5.33 The $L_2$-Errors for the standard European put Gammas with dividend with respect to the total number of grid points using the SEM with $\delta t = 10^{-5}$ year. ................................. 168

5.34 The $L_2$-Errors for the standard European put Gammas with dividend with respect to time step using the SEM with the total number of grid points = 61. ................................. 168

5.35 Comparison between analytical Black-Scholes Thetas and two-element spectral element method Thetas for the standard European put option with dividend. ................................. 169

5.36 The $L_2$-Errors for the standard European put Thetas with dividend with respect to the total number of grid points using the SEM with $\delta t = 10^{-5}$ year. ................................. 169

5.37 The $L_2$-Errors for the standard European put Thetas with dividend with respect to time step using the SEM with the total number of grid points = 61. ................................. 170

5.38 Comparison of $L_2$-Errors for the standard European put option price with dividend with respect to the total number of grid points using FDM and SEM. ................................. 171
5.39 Comparison of computational time for the standard put option with constant dividend with respect to total number of grid points using FDM and SEM. ................................................................. 171
5.40 Comparison of \( L_2 \)-Errors for the standard European put option price with dividend with respect to the total number of grid points using FDM and SEM. ................................................................. 172
5.41 Comparison of computational time for the standard put option with constant dividend with respect to total number of grid points using FDM and SEM. ................................................................. 172
5.42 Final payoff of a European condor spread option. ......................... 173
5.43 The European condor option prices over time-to-maturity and stock price using the five-element spectral element method with 45 grid points per element. ................................................................. 175
5.44 Comparison between the analytical option prices and the spectral element method prices for the European condor option with 45 grid points per element. ................................................................. 176
5.45 The \( L_2 \)-Errors for the European condor option with respect to the number of grid points per element using the SEM with the time step \( \delta t \doteq 10^{-5} \) year. ................................................................. 176
5.46 The \( L_2 \)-Errors for price of the European condor spread option with respect to the time step using the SEM with number of grid points per element = 45. ................................................................. 177
5.47 Comparison between the analytical option Deltas and the spectral element method Deltas for the European condor option with 45 points per element. ................................................................. 177
5.48 The \( L_2 \)-Errors for the European condor Delta with respect to the number of grid points per element using the SEM with the time step \( \delta t \doteq 10^{-5} \) year. ................................................................. 178
5.49 The \( L_2 \)-Errors for price of the European condor spread Delta with respect to the time step using the SEM with number of grid points per element = 45. ................................................................. 178
5.50 Comparison between the analytical option Gammas and the spectral element method Gammas for the European condor option with 45 points per element. ................................................................. 179
5.51 The $L_2$-Errors for the European condor Gamma with respect to the number of grid points per element using the SEM with the time step $\delta t \doteq 10^{-5}$ year. 

5.52 The $L_2$-Errors for price of the European condor spread Gamma with respect to the time step using the SEM with number of grid points per element $= 45$. 

5.53 Comparison between the analytical option Thetas and the spectral element method Thetas for the European condor option with 45 points per element. 

5.54 The $L_2$-Errors for the European condor Theta with respect to the number of grid points per element using the SEM with the time step $\delta t \doteq 10^{-5}$ year. 

5.55 The $L_2$-Errors for price of the European condor spread Theta with respect to the time step using the SEM with number of grid points per element $= 45$. 

5.56 Comparison of $L_2$-Errors for the European condor option price with respect to the total number of grid points using FDM and spectral element method. 

5.57 Comparison of computational time for the European condor option with respect to total number of grid points using FDM and spectral element method. 

6.1 European cash-or-nothing binary put option final payoff. 

6.2 European cash-or-nothing binary put final payoff approximation. 

6.3 The standard European binary put prices over time-to-maturity and stock price using two-element spectral element method with GLL and stable GRL. 

6.4 Comparison between analytical closed-form prices and two-element spectral element method prices for the standard European binary put at inception. 

6.5 The $L_2$-Errors for the standard European binary put prices with respect to the total number of grid points using two-element method with $\delta t \doteq 10^{-4}$ year.
6.6 The $L_2$-Errors for the standard European binary put prices with respect to time step using the two-element method with the total number of grid points = 61.

6.7 Comparison between analytical Deltas and the two-element spectral element method Deltas for the standard European binary put option at inception.

6.8 The $L_2$-Errors for the European binary put Deltas with respect to the total number of the grid points using the two-element method with $\delta t = 10^{-4}$ year.

6.9 Comparison between analytical closed-form Gammas and two-element spectral element method Gammas for the European binary put option at inception.

6.10 The $L_2$-Errors for the European binary put Gammas with respect to the total number of grid points using the two-element method with $\delta t = 10^{-4}$ year.

6.11 Comparison between analytical closed-form Thetas and two-element spectral element method Thetas for the European binary put option at the inception.

6.12 The $L_2$-Errors for the European binary put Thetas with respect to the total number of grid points using the two-element method with $\delta t = 10^{-4}$ year.

6.13 The standard European binary put prices over time-to-maturity and stock price using two-element spectral element method with GLL and stable GRL.

6.14 Comparison between analytical closed-form prices and two-element spectral element method prices for the standard European binary put at inception.

6.15 The $L_2$-Errors for the European binary put option prices with respect to the total number of grid points using the two-element method with $\delta t = 10^{-5}$ year.

6.16 The $L_2$-Errors for the European binary put option prices with respect to time step using the two-element method with the total number of grid points = 61.
6.17 Comparison between analytical Deltas and the two-element spectral element method Deltas for the standard European binary put option at the inception. .................................................. 206

6.18 The $L_2$-Errors for the European binary put Deltas with respect to the total number of the grid points using two-element method with $\delta t = 10^{-5}$ year. .......................................................... 206

6.19 The $L_2$-Errors for the European binary put Deltas with respect to time step using the two-element method with the total number of grid points = 61. .................................................. 207

6.20 Comparison between analytical closed-form Gammas and two-element spectral element method Gammas for the European binary put option at inception. .................................................. 208

6.21 The $L_2$-Errors for the European binary put Gammas with respect to the total number of grid points using the two-element method with $\delta t = 10^{-5}$ year. .......................................................... 208

6.22 The $L_2$-Errors for the European binary put Gammas with respect to the time step using two-element method with the total number of grid points = 61. .................................................. 209

6.23 Comparison between analytical closed-form Thetas and two-element spectral element method Thetas for the European binary put option at the inception. .................................................. 210

6.24 The $L_2$-Errors for the European binary put Thetas with respect to the total number of grid points using the two-element method with $\delta t = 10^{-5}$ year. .......................................................... 210

6.25 The $L_2$-Errors for the European binary put Thetas with respect to time step using the two-element method with the total number of grid points = 61. .................................................. 211

6.26 Comparison of $L_2$-Errors of four different spectral element methods for the binary put option price using the two-element method with $\delta t = 10^{-5}$ year. .......................................................... 212

6.27 Comparison of $L_2$-Errors of DP-SEM and approximation spectral element method for the binary put using the two-element method with respect to $\delta t$. .................................................. 212
<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.1</td>
<td>The European binary put option prices over time-to-maturity and stock price using the two-element spectral element method under the lognormal model.</td>
</tr>
<tr>
<td>7.2</td>
<td>Comparison of option prices between the analytical values and the spectral element method for the European binary put under the lognormal model.</td>
</tr>
<tr>
<td>7.3</td>
<td>The European binary put option prices over time-to-maturity and stock price using the two-element spectral element method under the CEV with $\beta = 0.75$.</td>
</tr>
<tr>
<td>7.4</td>
<td>Comparison of option prices between SEM and MCM for the European binary put under the CEV with $\beta = 0.75$ along with the standard binary put price.</td>
</tr>
<tr>
<td>7.5</td>
<td>Comparison of option prices between the analytical values and the spectral element method for the European binary put under the CEV with $\beta = 0.75$.</td>
</tr>
<tr>
<td>7.6</td>
<td>The $L_2$-Errors for the European binary put option prices with respect to the total number of grid points under the CEV with $\beta = 0.75$ and $\delta t = 10^{-5}$ year.</td>
</tr>
<tr>
<td>7.7</td>
<td>The $L_2$-Errors for the European binary put option prices with respect to time step under the CEV with $\beta = 0.75$ and the total number of grid points = 61.</td>
</tr>
<tr>
<td>7.8</td>
<td>The computed Deltas from the spectral element method for the European binary put option under the CEV model with $\beta = 0.75$ and $\delta t = 10^{-5}$ year.</td>
</tr>
<tr>
<td>7.9</td>
<td>The computed Gammas from the spectral element method for the European binary put option under CEV model with $\beta = 0.75$ and $\delta t = 10^{-5}$ year.</td>
</tr>
<tr>
<td>7.10</td>
<td>The computed Thetas from the spectral element method for the European binary put option under CEV model with $\beta = 0.75$ and $\delta t = 10^{-5}$ year.</td>
</tr>
<tr>
<td>8.1</td>
<td>Approximated European put option price surface over the forward price and the initial volatility under the SABR model using the asymptotic formula.</td>
</tr>
<tr>
<td>8.2</td>
<td>Approximated European put prices for selected initial volatilities.</td>
</tr>
</tbody>
</table>
8.3 Approximated Black equivalent volatility surface over the forward price and the forward initial volatility under SABR model using the asymptotic formula. ........................................... 229

8.4 Approximated Black equivalent volatilities for selected initial vols. . . 229

8.5 Two-dimensional option price mesh with optimal traversing algorithm. 234

8.6 Option price surface from two-element MD-SEM with \((36 + 36) \times 16\) points for two elements in the forward price and the initial volatility respectively. .................................................. 235

8.7 Option prices from two-element MD-SEM with \((36+36) \times 16\) points for selected initial forward volatilities with comparison with the asymptotic formula. .................................................. 235

8.8 The MD-SEM \(L_2\)-Errors for the European put option price with respect to the total number of grid points on the forward price \(F\) under the SABR model. .................................................. 236

8.9 The MD-SEM \(L_2\)-Errors for the European put option price with respect to the time step \(\delta t\) under the SABR model with \((36 + 36) \times 16\) points on \(F\) and \(\alpha\). .................................................. 236
List of Abbreviations and Symbols

Symbols

Standard European option related symbols in the Black-Scholes model,

\[ S(t) \quad \text{Stock price at time } t. \]
\[ B(t) \quad \text{Money market account at } t. \]
\[ K \quad \text{Option strike price.} \]
\[ r \quad \text{Risk-free interest rate.} \]
\[ q \quad \text{Continuous dividend yield.} \]
\[ \sigma \quad \text{Constant volatility of stock.} \]
\[ T \quad \text{Option time to maturity.} \]
\[ \mathcal{N}(\cdot) \quad \text{Standard normal CDF.} \]
\[ C(\cdot) \quad \text{Call-type option price.} \]
\[ P(\cdot) \quad \text{Put-type option price.} \]
\[ \Pi(\cdot) \quad \text{Call or put option price.} \]
\[ \Delta \quad \text{Option Greek Delta.} \]
\[ \Gamma \quad \text{Option Greek Gamma.} \]
\[ \Theta \quad \text{Option Greek Theta.} \]

Orthogonal polynomial and Gauss quadrature related symbols,

\[ p(x) \quad \text{An arbitrary polynomial.} \]
\[ q(x), r(x) \quad \text{More arbitrary polynomials.} \]
\( q_n(x), r_n(x) \) Arbitrary \( n^{th} \) degree polynomials.

\( \ell_i(x) \) \( i^{th} \) Lagrange polynomial.

\( w(x) \) A weighting function.

\( p_n(x) \) An orthogonal polynomial of degree \( n \).

\( \pi_n(x) \) A monic orthogonal polynomial of degree \( n \).

\( \varphi_n(x) \) An orthonormal polynomial of degree \( n \).

\( x_i \) \( i^{th} \) grid point in Gauss-type quadrature.

\( w_i \) \( i^{th} \) weight in Gauss-type quadrature.

\( J_n \) \( n^{th} \) Jacob matrix for orthogonal polynomials.

\( a_n \) \( n^{th} \) constant in monic three-term recurrence.

\( b_n \) \( n^{th} \) constant in monic three-term recurrence.

\( u_0 \) First moment for orthogonal polynomials.

\( T_n(x) \) Chebyshev polynomial of degree \( n \).

\( H_n(x) \) Hermite polynomial of degree \( n \).

\( L_n(x) \) Laguerre polynomial of degree \( n \).

\( P_n(x) \) Legendre polynomial of degree \( n \).

\( \mathcal{H}_n(x) \) Stable Hermite polynomial of degree \( n \).

\( \mathcal{L}_n(x) \) Stable Laguerre polynomial of degree \( n \).

\( \phi_j(x) \) \( j^{th} \) orthogonal polynomial basis function.

\( D_{ij} \) First derivative of \( \phi_j(x) \) at \( i^{th} \) grid point.

\( B_{ij} \) Second derivative of \( \phi_j(x) \) at \( i^{th} \) grid point.

\( \omega_i \) \( i^{th} \) weight in stable Gauss-type quadrature.

\( \psi_j(x) \) Stable \( j^{th} \) orthogonal polynomial basis function.

\( D_{ij} \) First derivative of \( \psi_j(x) \) at \( i^{th} \) grid point.

\( B_{ij} \) Second derivative of \( \psi_j(x) \) at \( i^{th} \) grid point.

\( J \) The Jacobian of domain transformation.
Spectral element method related symbols,

\begin{align*}
M & \quad \text{Mass matrix.} \\
S & \quad \text{Stiffness matrix.} \\
v(S) & \quad \text{Test function in the spectral element method.} \\
\alpha & \quad \text{Convergence rate with the order number.} \\
\beta & \quad \text{Convergence rate with the time step.} \\
\gamma & \quad \text{Convergence rate with the element number.} \\
n & \quad \text{Polynomial order for the basis function.} \\
\delta t & \quad \text{Time step for the Crank-Nicolson method.} \\
m & \quad \text{Number of elements in the SEM.} \\
L_2\text{-Error} & \quad \text{Convergence error, } L_2 \text{ relative error.} \\
L_\infty\text{-Error} & \quad \text{Convergence error, } L_\infty \text{ relative error.}
\end{align*}

CEV and SABR model related symbols,

\begin{align*}
\beta & \quad \text{Elasticity parameter in CEV.} \\
\chi^2(\cdot) & \quad \text{Noncentral chi-squared CDF.} \\
\theta & \quad \text{Degree of freedom in } \chi^2(\cdot). \\
\lambda & \quad \text{Noncentrality parameter in } \chi^2(\cdot). \\
F(t) & \quad \text{Forward price at time } t. \\
\alpha(t) & \quad \text{Forward initial volatility at } t. \\
\beta & \quad \text{Exponent parameter in SABR.} \\
\nu & \quad \text{Volatility of volatility parameter.} \\
\rho & \quad \text{Correlation of two Wiener processes.} \\
\sigma_B & \quad \text{Black equivalent volatility.} \\
\phi(x)\varphi(\xi) & \quad \text{Two-dimensional basis function.}
\end{align*}
<table>
<thead>
<tr>
<th>Abbreviations</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABM</td>
<td>Arithmetic Brownian Motion.</td>
</tr>
<tr>
<td>AOA</td>
<td>Absence Of Arbitrage.</td>
</tr>
<tr>
<td>BSM</td>
<td>Black-Scholes-Merton.</td>
</tr>
<tr>
<td>CDF</td>
<td>Cumulative Distribution Function.</td>
</tr>
<tr>
<td>CEV</td>
<td>Constant Elasticity of Variance.</td>
</tr>
<tr>
<td>DP-SEM</td>
<td>Discontinuous Payoff SEM.</td>
</tr>
<tr>
<td>FDM</td>
<td>Finite Difference Method.</td>
</tr>
<tr>
<td>FEM</td>
<td>Finite Element Method.</td>
</tr>
<tr>
<td>GBM</td>
<td>Geometric Brownian Motion.</td>
</tr>
<tr>
<td>GLL</td>
<td>Gauss-Lobatto-Legendre.</td>
</tr>
<tr>
<td>GRL</td>
<td>Gauss-Radau-Laguerre.</td>
</tr>
<tr>
<td>MCM</td>
<td>Monte Carlo Method.</td>
</tr>
<tr>
<td>MD-SEM</td>
<td>Multi-Dimensional SEM.</td>
</tr>
<tr>
<td>ME-SEM</td>
<td>Multi-Element SEM.</td>
</tr>
<tr>
<td>PDE</td>
<td>Partial Differential Equation.</td>
</tr>
<tr>
<td>SABR</td>
<td>Stochastic Alpha Beta Rho.</td>
</tr>
<tr>
<td>SDE</td>
<td>Stochastic Differential Equation.</td>
</tr>
<tr>
<td>SEM</td>
<td>Spectral Element Method.</td>
</tr>
<tr>
<td>SEM-NI</td>
<td>SEM with Numerical Integration.</td>
</tr>
</tbody>
</table>
Acknowledgements

I am sincerely indebted to my dissertation committee, Qing H. Liu (Supervisor), A. Craig Burnside, Anita T. Layton, Xuejun Liao, Yaroslav A. Urzhumov and Gary A. Ybarra for their support, guidance and encouragement throughout the dissertation process. I am very fortunate to have these distinguished scholars to work with me on this exciting interdisciplinary endeavor. I greatly appreciate the assistance from professors and colleagues at Duke University, especially Lawrence Carin, David J. Brady, Jeffrey L. Krolik, Silvia Ferrari, Stacy L. Tantum, Wenyi Hu, Robert T. Clemen, Peng Sun, Arlie O. Petters, Christopher K. Carter and Aysenil Belger. I would like to thank Jaime Casassus, Domingo A. Tavella, Linda Kreitzman and Richard K. Lyons at the University of California, Berkeley for their help and advice. I benefited tremendously from the stimulating discussions with fellow researchers, Peter Carr, Alireza Javaheri, Wuming Zhu and Feng Chen. Finally, I am deeply grateful to my family for their sacrifice and understanding.
1

Introduction

1.1 Option Theory History

In the early 1950s, Leonard J. Savage sent a postcard to Paul A. Samuelson, which suggested he should read Louis Bachelier’s Ph.D. thesis, *Théorie de la Spéculation* [1]. It was successfully defended at the Sorbonne on March 29, 1900, and subsequently published in the prestigious French scientific journal, *Annales Scientifiques de l’École Normale Supérieure* [2]. Bachelier worked at the Paris Bourse during that time, which inspired his idea on his doctoral research about option pricing model. He made some revolutionary assumptions of how stock price evolves as a “random walk”, which essentially is a Brownian motion, i.e. an arithmetic Brownian motion (ABM). The Brownian motion was “first” observed on plant pollen suspended in water under microscope by the Scottish botanist, Robert Brown, in late 1820s. Louis Bachelier demonstrated the pricing formulas for the standard European options under arithmetic Brownian motion, which are very similar to and provided the foundation for the follow-up option formulas including the famous *Black-Scholes-Merton (BSM)*’s formula. Bachelier didn’t realize that he solved an important physics problem. His
original study on “random walk” went beyond and anticipated Albert Einstein’s celebrated work of Brownian motion in 1905 [3]. Some of the equations Einstein derived are identical to the ones in Bachelier’s doctoral thesis. Brownian motion was notably further developed by Norbert Wiener in 1920s.

Despite the high appreciation from Bachelier’s thesis advisor, Henri Poincaré, his work was ignored for about half a century. It was Paul A. Samuelson who introduced it to the economic sciences and began to analyze things in economics as stochastic processes such as the movements of the stock markets. The arithmetic Brownian motion (ABM), which is still used in finance, is probably reasonable to model a stock spread, but not suitable for modeling a stock itself, for a stock price is never negative. In the late 1950s, M. F. M. Osborne showed the logarithm of stock price follows arithmetic Brownian motion, while the stock price movement is actually a lognormal process [4]. In the early 1960s, Case M. Sprenkle and Paul A. Samuelson also adapted the Bachelier approach to the assumption of geometric Brownian motion (GBM), where stock price will never go below zero [5, 6]. During this rapid development period for the theory of derivative pricing, series of option pricing formulas were invented by various researchers, most of which are very close to the option pricing formula developed by Fischer Black, Myron Scholes [7] and Robert C. Merton [8] in 1973. It is only the groundbreaking work of the Black-Scholes-Merton model that explicitly introduced the concept of dynamic hedging strategy which leads to the potential for practical use.

From the moment Paul A. Samuelson received the postcard, not only did the Louis Bachelier’s work begin to become part of Samuelson’s own research, but also it was passed to his student Robert C. Merton and other pioneering researchers including Fischer Black and Myron Scholes. It started the new age of mathematical finance, though the real birthdate should be traced back to March 29, 1900, when Louis Bachelier defended his doctoral thesis [2].
1.2 Numerical Methodologies

Modern finance is based on assumptions regarding the statistical behavior of real-world risks of equities, credits, commodataries, interest and exchange rates. A faithful reproduction of the statistical behavior allows the formulation of mathematical models. Many of these models can be expressed as *partial differential equations (PDEs)*. Partial differential equations have been studied for centuries by mathematicians, physicists and engineers, therefore a great deal of knowledge from these areas is available and can be implemented to finance. Any new developments will continue to be applied onto the modern financial world.

Option in finance is a contingent claim (a derivative contract) on an underlying security. For example, a European call option allows the buyer of the contract to purchase the underlying security at an agreed strike price at a maturity time in the future, while a European put option gives the buyer the right to sell the underlying asset at the preset strike price at the maturity. Under the assumption of geometric Brownian motion (GBM), the stock dynamics is [9],

\[
\frac{dS(t)}{S(t)} = \mu dt + \sigma dW(t)
\]  

(1.1)

where \( \mu \) is the drift of the stock, \( \sigma \) is the stock volatility and \( W(t) \) is the standard Brownian motion, i.e., Wiener process. The stock dynamics leads to the following PDE for the standard European call and put options [9],

\[
\frac{1}{2} \sigma^2 S^2(t) \frac{\partial^2 \Pi}{\partial S^2} + rS(t) \frac{\partial \Pi}{\partial S} + \frac{\partial \Pi}{\partial t} - r\Pi = 0
\]  

(1.2)

where \( \Pi \) is denoted as the call or put option price, with certain final payoff conditions at the maturity. The standard European options under the GBM have analytical solutions, i.e., the partial differential equation (PDE) has closed formulas for the standard European call and put options.
However, certain exchanged and over-the-counter (OTC) options have no closed-form solutions, where numerical techniques are employed. The enormous success of financial derivatives pushes the strong need for sophisticated computational algorithms. Fast and accurate numerical methods have become essential tools for the derivative pricing and risk management in practice.

One commonly used method is the binomial tree model and its extension, the trinomial tree model, both of which are considered as lattice methods. The lattice approach is first proposed by Parkinson in 1977 [10] and Cox et al. in 1979 [11]. The other popular method is the Monte Carlo method (MCM), which is first applied on option pricing by Phelim P. Boyle in 1977 [12] and further discussed along with variance reduction techniques in [13, 14]. The MCM and its varieties are especially excellent for high dimensional problems and path-dependent options [15].

Lattice method is in fact simple explicit finite difference method (FDM) [16]. The FDM is first introduced in option pricing by Brennan and Schwartz in 1977 and 1978 [17, 18]. FDMs extend the lattice methods to discretize the stock price and the time line simultaneously and approximate the derivatives by those discrete points [19]. The FDM has three major approaches. One is the explicit method, which is easy to implement, but only conditionally stable. The other two are implicit methods including the fully implicit method and the Crank-Nicolson method. While both implicit methods are unconditionally stable, the Crank-Nicolson method is generally considered more accurate, which combines the other two methods [20].

Both lattice methods and FDMs are affected by non-smoothness in option’s final payoff, which includes discontinuousness in payoff or its derivative [16]. The finite element method (FEM) can easily incorporate initial and boundary conditions involving complicated geometries including the non-smoothness [21]. Hence, the FEM is especially well-suited for pricing some exotic options, like barrier options studied in [22]. Another alternative is the spectral method, which uses high-order polynomi-
als to achieve high efficiency [23, 24]. However, the spectral method works only for globally smooth functions, and it has difficulties with discontinuities.

The relatively new *spectral element method (SEM)* enjoys the spectral accuracy of the spectral method and the geometric flexibility of the finite element method [25, 26]. The spectral element method achieves exponential convergence on error with respect to the polynomial order [27]. Thus the spectral related methods significantly reduce computing time cost and memory requirement compared to the conventional finite difference method and finite element method [27, 28].

The spectral element method has been applied to fluid dynamics and was actively implemented in semiconductor nanodevice simulations [27] and electromagnetics computations [29] for its significant efficiency. In this dissertation, spectral element method (SEM) and related techniques are developed and implemented for more accurate and efficient solutions for financial partial differential equations, specifically on European options and their Greeks.

Spectral element method (SEM) for option pricing was first published in Zhu’s Ph.D. dissertation in 2007 [30]. It is innovative at that time, while the aruthor started working on option pricing using SEM independently in 2007 [31]. Zhu and Kopriva subsequently published the work solving Merton’s jump diffusion model and Heston’s stochastic volatility model in 2009 [32, 33]. The SEM was then applied to a European rainbow option in 2010 [34]. Adaptive SEMs were implemented to price American-style options in Willyard’s Ph.D. dissertation in 2011 [35]. When Legendre functions are applied in SEM, the domain of stock price has to be truncated with approximate boundary conditions. In 2012, Chen et al. applied Laguerre functions on European options, which avoided the domain truncation error [26]. This dissertation further improves the existing implementations of the spectral element method for European options and their Greeks and invents new exciting spectral element related techniques for new classes of derivative pricing problems.
1.3 Innovative Contributions

- The general Gauss quadrature theories, with and without pre-assigned nodes, are summarized and presented in a unified approach. A general method of combining the eigenvalue method and the Newton-Raphson method is proposed to locate the grid points accurately for Gauss-type quadratures. The stable Gauss-Hermite quadrature, stable Gauss-Radau-Laguerre (GRL) quadrature and Gauss-Lobatto-Legendre (GLL) quadrature are developed along with the analytical first-order and second-order spectral differential matrices of the associated basis functions. Those techniques are crucial for implementing the spectral element method (SEM) with high accuracy.

- Spectral element methods are derived on the Black-Scholes partial differential equations (PDEs) in great details from the one-element approach to the two-element approach. The comparison between the one-element and two-element method for a standard European put option is presented to demonstrate the necessity of multiple elements for options with non-smooth payoffs. The comparison with finite difference method (FDM) using the same Crank-Nicolson time-integration approach shows significant increase of accuracy and reduction of computational time for the spectral element method (SEM).

- The spectral element method on an extension of the Black-Scholes model with continuous compounding dividend yield is implemented for a European put option. The SEM is also compared with the finite different method (FDM) for a European put option with dividend. A European condor spread option is solved by a similar five-element spectral element method, which demonstrates the flexibility of the multi-element spectral element method (ME-SEM) for complicated payoffs with multiple non-smooth points.
• The discontinuous payoff spectral element method (DP-SEM) is invented specifically for options with discontinuous payoffs. The option payoffs at the discontinuous point are correctly assigned to the associated elements under the DP-SEM. The novel approach dwarfs the native method and the average method and is better than the proposed approximation method as well. A European binary put option is computed with the DP-SEM, which achieves the high accuracy comparable with that from the regular European put option.

• The partial differential equation for European options with the constant elasticity of variance (CEV) local volatility model is derived. A European binary option under the CEV model is priced with the spectral element method (SEM). The computed results are compared with those from the Monte Carlo method (MCM) and verified from the option valuation formula.

• The stochastic alpha beta rho (SABR) volatility model on forward price is introduced and its partial differential equation is derived. The multi-dimensional spectral element method (MD-SEM) is carefully derived and implemented on the SABR stochastic volatility model. The computational results are compared and verified with the values from the asymptotic formula.

• The techniques to compute option Greeks, such as Delta, Gamma and Theta, are developed under the spectral element method (SEM). These Greek computational methods are implemented for European options together with the option pricing. The error convergence with respect to the number of grid points for option prices and option Greeks is illustrated. It shows the exponential convergence property of the spectral element method (SEM). The relative errors with respect to the time step are sketched and show a second-order convergence for the Crank-Nicolson time-integration method.
1.4 Outline of the Dissertation

The rest of this dissertation is organized as follows. First the Black-Scholes model is introduced in Chapter 2 with some illustrations. The essential techniques of the spectral element method, Gauss quadrature rules, are discussed and developed in Chapter 3. An introduction of spectral element method (SEM) is presented in Chapter 4. The techniques to implement spectral element methods on Black-Scholes partial differential equations (PDEs) in one-element and multi-element are extensively investigated in Chapter 5. The invention of discontinuous payoff spectral element method (DP-SEM) is shown in Chapter 6 and numerically validated on a European binary put option. The constant elasticity of variance (CEV) local volatility model is reviewed and implemented with the spectral element method in Chapter 7. An extension of the CEV model, the stochastic alpha beta rho (SABR) volatility model is solved using the multi-dimensional spectral element method (MD-SEM) in Chapter 8. Finally, in Chapter 9, this dissertation’s major contributions are summarized, followed by the conclusion and the discussion of further work.
2

The Black-Scholes Model

2.1 Derivation of PDE

In the financial market, a standard European call option gives the option buyer the right but not the obligation to purchase a stock $S(t)$ in the future at maturity $T$ at a preset strike price $K$. So the pay off the call option at maturity will be,

$$C(S, T) = \max\{S(T) - K, 0\}$$  \hspace{1cm} (2.1)

The buyer of option will only exercise the option when the call is in-the-money, i.e. the stock price at maturity is greater than the strike price, $S(T) > K$. Since the buyer has the right of the option at the beginning, so the call must have an initial price $C = C(0)$. The Black-Scholes Model is to solve this pricing problem.

The Black-Scholes partial differential equation (PDE) is originally developed by Fischer Black and Myron Scholes [7], and Robert C. Merton [8]. Here is one derivation of the Black-Scholes partial differential equation by replicating.

First, for amount of $B(0)$ dollars invested in a risk-free money market with interest rate $r$ continuously compounding at time $t = 0$, the cash will grow like,

$$B(t) = B(0)e^{rt}, \quad \text{and then} \quad dB(t) = rB(t)dt$$  \hspace{1cm} (2.2)
In the Black-Scholes world, the stock price is assumed to follow *Geometric Brownian Motion* (GBM), so that the stock price will never be negative, that is,

\[
\frac{dS(t)}{S(t)} = \mu dt + \sigma dW(t)
\]  
(2.3)

where \( \mu \) is the drift of the stock and \( \sigma \) is the stock’s volatility, both of which are assumed constant, and the interest rate \( r \) is assumed constant too.

If a portfolio \( V(t) = \Delta(t)S(t) + B(t) \) is *self-financing*, i.e. no outside asset in and out of the portfolio, and \( \Delta(t) \) is the number of shares of stock held at \( t \), then,

\[
dV(t) = \Delta(t) dS(t) + dB(t)
\]  
(2.4)

By longing one call option and selling an appropriate number of shares \( \Delta(t) \), the following corresponding portfolio can be made instantaneously risk-free,

\[
V(t) = C(S, t) - \Delta(t)S(t)
\]  
(2.5)

Using *Itô’s Lemma* and because of self-financing condition, then

\[
dV(t) = dC(S, t) - \Delta(t) dS(t)
\]

\[
= C_t dt + C_S dS + \frac{1}{2} C_{SS} dSdS - \Delta(t)dS(t)
\]

\[
= C_t dt + C_S \mu S dt + C_S \sigma S dW + \frac{1}{2} C_{SS} \sigma^2 S^2 dt - \Delta(t)dS(t)
\]

\[
= \left[ \frac{1}{2} \sigma^2 S^2(t) C_{SS} + \mu S(t) C_S + C_t \right] dt
\]

\[
+ \sigma S(t) C_S dW(t) - \Delta(t) \left[ \mu S(t) dt + \sigma S(t) dW(t) \right]
\]

\[
= \left[ \frac{1}{2} \sigma^2 S^2(t) C_{SS} + \mu S(t) C_S + C_t - \Delta(t) \mu S(t) \right] dt
\]

\[
+ \left[ \sigma S(t) C_S - \Delta(t) \sigma S(t) \right] dW(t)
\]  
(2.6)

Eliminating the diffusion term because of the risk-free portfolio, then,

\[
\sigma S(t) C_S - \Delta(t) \sigma S(t) = 0 \quad \Rightarrow \quad C_S = \Delta(t)
\]  
(2.7)
therefore,
\[ dV(t) = \left[ \frac{1}{2} \sigma^2 S(t) C_{SS} + C_t \right] dt \]  

(2.8)

Because of absence of arbitrage (AOA),
\[ dV(t) = rV(t)dt = r[C - S(t)C_S]dt \]  

(2.9)

therefore,
\[ \frac{1}{2} \sigma^2 S(t) C_{SS} + C_t = r[C - S(t)C_S] \]  

(2.10)

Then, the Black-Scholes PDE becomes,
\[ \frac{1}{2} \sigma^2 S(t) C_{SS} + rS(t)C_S + C_t - rC = 0 \]  

(2.11)

with a set of final and boundary conditions,
\[ C(S, T) = \max\{S(T) - K, 0\} \]  

(2.12)
\[ C(0, t) = 0, \quad \forall \ 0 \leq t \leq T \]  

(2.13)
\[ C(S, t) = S, \quad \text{as} \quad S \to \infty \]  

(2.14)

The first final condition is an essential condition and the rest two boundary conditions are the natural boundary conditions.

Similarly, a standard European put option gives the option buyer the right but not the obligation to sell a stock \( S(t) \) at the maturity \( T \) with a strike price \( K \), the put price \( P \) follows the same partial differential equation (PDE),
\[ \frac{1}{2} \sigma^2 S(t) P_{SS} + rS(t)P_S + P_t - rP = 0 \]  

(2.15)

with a set of unique final and boundary conditions,
\[ P(S, T) = \max\{K - S(T), 0\} \]  

(2.16)
\[ P(0, t) = Ke^{-r(T-t)}, \quad \forall \ 0 \leq t \leq T \]  

(2.17)
\[ P(S, t) = 0, \quad \text{as} \quad S \to \infty \]  

(2.18)
2.2 Heat Transformation

The Black-Scholes partial differential equation is a parabolic PDE as the one dimensional heat equation. Therefore, if $\tau$, $x(\tau)$, $v(x, \tau)$, $k$, $u(x, \tau)$ are defined as follows,

$$\tau = \frac{\sigma^2}{2} (T - t)$$

(2.19)

$$x = \ln \frac{S}{K}$$

(2.20)

$$v = \frac{C}{K}$$

(2.21)

$$k = \frac{r}{\sigma^2/2}$$

(2.22)

$$u = v \cdot e^{\frac{1}{2}(k-1)x + \frac{1}{4}(k+1)^2 \tau}$$

(2.23)

The Black-Scholes equation becomes a heat equation,

$$\frac{\partial u}{\partial \tau} = \frac{\partial^2 u}{\partial x^2}, \quad 0 \leq \tau \leq \frac{\sigma^2}{2} T, \quad -\infty < x < \infty$$

(2.24)

with a final condition for a European call,

$$u(x, 0) = \max\{e^{(k+1)\frac{x}{2}} - e^{-(k-1)\frac{x}{2}}, 0\}$$

(2.25)

and a final condition for a European put,

$$u(x, 0) = \max\{e^{-(k-1)\frac{x}{2}} - e^{(k+1)\frac{x}{2}}, 0\}$$

(2.26)

2.3 Closed-Form Solution

A standard European call price follows the following PDE,

$$\frac{1}{2} \sigma^2 S^2(t) C_{SS} + r S(t) C_S + C_t - r C = 0$$

(2.27)

with a final condition,

$$C(S, T) = \max\{S(T) - K, 0\}$$

(2.28)
The closed-form solution, which can be derived using variety of approaches, is

\[ C(S,t) = S(t)N(d_1) - e^{-r(T-t)}KN(d_2) \quad (2.29) \]

where \( N(\cdot) \) is the cumulative distribution function (CDF) of a standard normal random variable,

\[ N(d) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{d} e^{-z^2/2} dz \quad (2.30) \]

and where,

\[ d_1 = \frac{\ln \frac{S(t)}{K} + (r + \frac{1}{2}\sigma^2)(T-t)}{\sigma \sqrt{T-t}} \quad (2.31) \]

\[ d_2 = d_1 - \sigma \sqrt{T-t} \quad (2.32) \]

Likewise, a standard European put price follows the following PDE,

\[ \frac{1}{2} \sigma^2 S^2(t)P_{SS} + rS(t)P_S + P_t - rP = 0 \quad (2.33) \]

with a final condition,

\[ P(S,T) = \max\{K - S(T), 0\} \quad (2.34) \]

The closed-form solution is

\[ P(S,t) = e^{-r(T-t)}KN(-d_2) - S(t)N(-d_1) \quad (2.35) \]

with same \( d_1 \) and \( d_2 \) in the closed-form call option price’s formula.

From the closed-form solutions of the standard European put and call, the well-known relation named as put-call parity in the Black-Scholes world is as follows,

\[ C(S,t) - P(S,t) = S(t) - e^{-r(T-t)} K, \quad \forall \quad 0 \leq t \leq T \quad (2.36) \]
2.4 Option Sensitivities (Greeks)

In the real world of option investment, besides the stock price, some other parameters, which are assumed constant in the Black-Scholes formula, can be changing over time too. For example, the volatility of stock probably will not stay constant during the life of option. It may depend on the level of stock price and other factors.

The partial derivatives of option value measure the sensitivities of option price with respect to the underlying independent variables. These partial derivatives are named as the *Greeks* because the most sensitivities are often denoted by Greek letters. Table 2.1 lists commonly used European option Greeks [36], where Π is either the call option (C) or the put option (P).

<table>
<thead>
<tr>
<th>Name</th>
<th>Formula</th>
<th>Call Option</th>
<th>Put Option</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delta</td>
<td>$\Delta = \frac{\partial \Pi}{\partial S}$</td>
<td>$N(d_1)$</td>
<td>$-N(-d_1)$</td>
</tr>
<tr>
<td>Gamma</td>
<td>$\Gamma = \frac{\partial^2 \Pi}{\partial S^2}$</td>
<td>$\frac{N'(d_1)}{\sigma \sqrt{T-t}}$</td>
<td></td>
</tr>
<tr>
<td>Vega</td>
<td>$\nu = \frac{\partial \Pi}{\partial \sigma}$</td>
<td>$SN'(d_1)\sqrt{T-t} = e^{-r(T-t)}KN'(d_2)\sqrt{T-t}$</td>
<td></td>
</tr>
<tr>
<td>Theta</td>
<td>$\Theta = \frac{\partial \Pi}{\partial t}$</td>
<td>$-\frac{\sigma SN'(d_1)}{2\sqrt{T-t}} - \frac{rKN(d_2)}{e^{r(T-t)}}$</td>
<td>$-\frac{\sigma SN'(d_1)}{2\sqrt{T-t}} + \frac{rKN(-d_2)}{e^{r(T-t)}}$</td>
</tr>
<tr>
<td>Rho</td>
<td>$\rho = \frac{\partial \Pi}{\partial r}$</td>
<td>$(T-t)e^{-r(T-t)}KN(d_2)$</td>
<td>$-(T-t)e^{-r(T-t)}KN(-d_2)$</td>
</tr>
<tr>
<td>Lambda</td>
<td>$\lambda = \frac{\partial \ln \Pi}{\partial \ln S}$</td>
<td>$SN'(d_1)/C$</td>
<td>$-SN(-d_1)/P$</td>
</tr>
</tbody>
</table>

The option Greeks are vital concepts and tools in the risk management of financial portfolio. The Greeks can be used to rebalance the portfolio to achieve desired exposure to a certain risk. More importantly, knowing the Greek, a particular expo-
sure can be \textit{hedged} from adverse changes in the market by using appropriate amount
of other related financial instruments.

The most popular and frequently used Greeks are Delta ($\Delta$), Gamma ($\Gamma$) and
Vega ($\nu$). From the Black-Scholes PDE, Delta ($\Delta$), Gamma ($\Gamma$) and Theta ($\Theta$) are
related by the following equation,

$$\frac{1}{2} \sigma^2 S^2 \Gamma + r S \Delta + \Theta - r \Pi = 0 \hspace{1cm} (2.37)$$

2.5 Black-Scholes with Dividends

If the underlying stock pays dividends with continuously compounding yield of $q$,
then the stock price follows the generalized geometric Brownian motion as follows,

$$\frac{dS(t)}{S(t)} = (\mu - q)dt + \sigma dW(t) \hspace{1cm} (2.38)$$

The corresponding PDE becomes,

$$\frac{1}{2} \sigma^2 S^2(\Pi_{SS} + (r-q)S(\Pi_S + \Pi_t - r \Pi) = 0 \hspace{1cm} (2.39)$$

The closed-form solutions for the call and put options are,

$$C(S, t) = S(t)e^{-q(T-t)}\mathcal{N}(d_1) - e^{-r(T-t)}K\mathcal{N}(d_2) \hspace{1cm} (2.40)$$
$$P(S, t) = e^{-r(T-t)}K\mathcal{N}(-d_2) - S(t)e^{-q(T-t)}\mathcal{N}(-d_1) \hspace{1cm} (2.41)$$

where,

$$d_1 = \frac{\ln \left(\frac{S(t)}{K}\right) + (r - q + \frac{1}{2} \sigma^2)(T - t)}{\sigma \sqrt{T-t}} \hspace{1cm} (2.42)$$
$$d_2 = d_1 - \sigma \sqrt{T-t} \hspace{1cm} (2.43)$$

The associated Greeks are very similar to the Greeks without dividends. Table 2.2
listed those commonly used European option Greeks [36],

15


<table>
<thead>
<tr>
<th>Name</th>
<th>Formula</th>
<th>Call Option</th>
<th>Put Option</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delta</td>
<td>$\Delta = \frac{\partial \Pi}{\partial S}$</td>
<td>$e^{-q(T-t)}N(d_1)$</td>
<td>$-e^{-q(T-t)}N(-d_1)$</td>
</tr>
<tr>
<td>Gamma</td>
<td>$\Gamma = \frac{\partial^2 \Pi}{\partial S^2}$</td>
<td>$\frac{e^{-q(T-t)}N'(d_1)}{\sigma S\sqrt{T-t}}$</td>
<td></td>
</tr>
<tr>
<td>Vega</td>
<td>$\nu = \frac{\partial \Pi}{\partial \sigma}$</td>
<td>$Se^{-q(T-t)}N'(d_1)\sqrt{T-t} = e^{-r(T-t)}KN'(d_2)\sqrt{T-t}$</td>
<td></td>
</tr>
<tr>
<td>Theta</td>
<td>$\Theta = \frac{\partial \Pi}{\partial t}$</td>
<td>$\frac{-\sigma Se^{-q(T-t)}N'(d_1)}{2\sqrt{T-t}} + \frac{qSN'(d_1)}{e^{q(T-t)}} - \frac{rKN'(d_2)}{e^{r(T-t)}}$</td>
<td>$\frac{-\sigma Se^{-q(T-t)}N'(d_1)}{2\sqrt{T-t}} - \frac{qSN'(d_1)}{e^{q(T-t)}} + \frac{rKN'(d_2)}{e^{r(T-t)}}$</td>
</tr>
<tr>
<td>Rho</td>
<td>$\rho = \frac{\partial \Pi}{\partial r}$</td>
<td>$(T-t)e^{-r(T-t)}KN'(d_2)$</td>
<td>$-(T-t)e^{-r(T-t)}KN'(-d_2)$</td>
</tr>
<tr>
<td>Lambda</td>
<td>$\lambda = \frac{\partial \ln \Pi}{\partial \ln S}$</td>
<td>$Se^{-q(T-t)}N'(d_1)/\Pi$</td>
<td>$-Se^{-q(T-t)}N(-d_1)/\Pi$</td>
</tr>
</tbody>
</table>

### 2.6 An Illustrative Option Example

For the purpose of illustration, a volatile stock $S(t)$ and associated European call and put options have parameter values and their units as follows,

$$K = 50, \quad r = 10\%$$

$$\sigma = 50\%, \quad T = 1.5 \text{ years} \quad (2.44)$$

The call and put option surfaces are plotted in Figure 2.1 and 2.2, and their Delta surfaces are in Figure 2.3 and 2.4. The Gamma surface for the call and put option is drawn in Figure 2.5, the Vega surface is in Figure 2.6, and the call and put Theta surfaces are graphed in Figure 2.7 and Figure 2.8.

For the purpose of readability, the time lines are labeled as time to maturity, $T - t$, instead of the actual time $t$, the option Theta coordinates and colormaps for both call and put options are plotted reversed.
Figure 2.1: Call option price surface for $K = \$50$, $r = 10\%$, and $\sigma = 50\%$

Figure 2.2: Put option price surface for $K = \$50$, $r = 10\%$, and $\sigma = 50\%$
Figure 2.3: Call option Delta surface for $K = $50, $r = 10\%$, and $\sigma = 50\%$

Figure 2.4: Put option Delta surface for $K = $50, $r = 10\%$, and $\sigma = 50\%$
Figure 2.5: Option Gamma surface for $K = $50, $r = 10\%$, and $\sigma = 50\%$

Figure 2.6: Option Vega surface for $K = $50, $r = 10\%$, and $\sigma = 50\%$
Figure 2.7: Call Theta surface for $K = 50$, $r = 10\%$, and $\sigma = 50\%$

Figure 2.8: Put Theta surface for $K = 50$, $r = 10\%$, and $\sigma = 50\%$
3

Gauss Quadrature Method

3.1 Riemann Integral

The Riemann integral is named after Georg Friedrich Berhard Riemann, a German mathematician, who provided a rigorous definition of the integral [37]. The Riemann sum for a function \( f \) defined on the interval \([a, b]\) is

\[
\sum_{i=1}^{n} f(\xi_i)(x_i - x_{i-1}), \quad \text{where} \quad a = x_0 < x_1 < \cdots < x_n = b
\]

(3.1)

The \( \xi_i \) is a point selected from the \( i \)th subinterval \([x_{i-1}, x_i] \); the collection of the subintervals is called a partition \( P \); the norm of the partition is

\[
|P| = \max_i \{\Delta x_i\}, \quad \text{where} \quad \Delta x_i = x_i - x_{i-1}, \quad 1 \leq i \leq n
\]

(3.2)

The Riemann integral is the limit of the Riemann sum, i.e., when \(|P| \to 0\),

\[
\sum_{i=1}^{n} f(\xi_i)(x_i - x_{i-1}) \quad \rightarrow \quad \int_{a}^{b} f(x)dx
\]

(3.3)

To approximate the integral, usually the \( \xi_i \) is selected as a left-end point, a right-end
point or a middle point in the $i$th subinterval, and the partition $P$ is equally spaced, that is, $\Delta x_1 = \Delta x_2 = \cdots = \Delta x_n$.

If letting $w_i = \Delta x_i$, we can rewrite (3.3) as follows,

$$\int_a^b f(x)dx \approx \sum_{i=1}^{n} w_i f(\xi_i) \quad (3.4)$$

Here, the Riemann integral is approximated as a weighted sum of the function $f$ at grid points $\{\xi_i\}$ with associated weights $\{w_i\}$, $1 \leq i \leq n$.

### 3.2 Newton-Cotes Formula

*Newton-Cotes formula* uses the Lagrange interpolation polynomial of degree $n$ to derive an integral approximation that is exact when $f(x)$ has a degree of $n$ or less, which is also based on evaluating the integrand at equally-spaced points [38].

Let $x_0, x_1, \ldots, x_n$ be equally-spaced nodes in $[a, b]$, and the Lagrange interpolation of a function $f(x)$ at these nodes is given by the polynomial $p(x)$,

$$p(x) = \sum_{i=0}^{n} f(x_i) \ell_i(x), \quad a = x_0 < x_1 < \cdots < x_n = b \quad (3.5)$$

where the Lagrange polynomial $\ell_i(x)$ is,

$$\ell_i(x) = \prod_{\substack{j=0 \atop j \neq i}}^{n} \frac{x-x_j}{x_i-x_j}, \quad i = 0, 1, \ldots, n \quad (3.6)$$

Then we have,

$$\int_a^b f(x)dx \approx \int_a^b p(x)dx = \int_a^b \sum_{i=0}^{n} f(x_i)\ell_i(x)dx = \sum_{i=0}^{n} w_i f(x_i) \quad (3.7)$$
where the weight $w_i$ is determined by,

$$w_i = \int_a^b \ell_i(x) \, dx \quad (3.8)$$

For the Lagrange interpolation in (3.5), $p(x)$ will be exact for a function $f(x)$ which is a polynomial with a degree $n$ or less, that is,

$$f(x) = \sum_{i=0}^n f(x_i) \ell_i(x) \quad (3.9)$$

In this case, the Newton-Cotes formula will evaluate the integral (3.7) precisely, i.e.,

$$\int_a^b f(x) \, dx = \sum_{i=0}^n w_i f(x_i) \quad (3.10)$$

Newton-Cotes formulas with their error terms for degree of $n = 1, 2, 3$ are listed from (3.11) to (3.14). For $n = 1$, it is the Trapezoidal rule,

$$\int_a^b f(x) \, dx = \frac{h}{2} [f(x_0) + f(x_1)] - \frac{h^3}{12} f''(\xi) \quad (3.11)$$

where

$$h = \frac{b - a}{n}, \quad \text{and} \quad a < \xi < b \quad (3.12)$$

and for $n = 2$, it is the celebrated formula of Simpson's rule,

$$\int_a^b f(x) \, dx = \frac{h}{3} [f(x_0) + 4f(x_1) + f(x_2)] - \frac{h^5}{90} f^{(4)}(\xi) \quad (3.13)$$

and for $n = 3$, it is the Simpson's $3/8$ rule,

$$\int_a^b f(x) \, dx = \frac{3h}{8} [f(x_0) + 3f(x_1) + 3f(x_2) + f(x_3)] - \frac{3h^5}{80} f^{(4)}(\xi) \quad (3.14)$$
3.3 Gauss Quadrature Rule

A polynomial of degree \( n \) is completely determined by \( n+1 \) specified values at distinct points, therefore it is not a surprise to expect the Newton-Cotes formula based on \( n+1 \) points can integrate precisely on a polynomial of degree less or equal to \( n \).

However, Gauss quadrature can exactly integrate a polynomial of degree up to \( 2n+1 \) with \( n+1 \) points by craftily selecting the integration points and the associated weights. It is about twice Newton-Cotes formula can do. No quadrature formula has order of accuracy higher than Gauss quadrature [39].

Moreover, Newton-Cotes integration uses values of the integrand at equally-spaced points, which can significantly decrease the accuracy of the approximation, while Gauss quadrature choose the integration points in an optimal way rather than simply equally spaced. The points and weights are carefully chosen to minimize the approximation error [38].

3.3.1 Orthogonal Polynomials

A system of infinite polynomials \( \{p_n(x), \ n = 0,1,2,...\} \) is called orthogonal on the finite or infinite interval \( [a,b] \) with respect to a non-negative weighting function \( w(x) \), if the inner product of \( p_n(x) \) and \( p_m(x) \) is zero when \( n \) does not equal to \( m \), i.e.,

\[
\langle p_n, p_m \rangle := \int_a^b w(x)p_n(x)p_m(x)dx = \|p_n\| \cdot \|p_m\| \cdot \delta_{nm} \tag{3.15}
\]

where all the moments, \( \{u_n\} \), exist and are finite, that is,

\[
u_n := \int_a^b x^n w(x)dx < \infty, \quad n = 0,1,2,\cdots \tag{3.16}
\]

and the first moment is positive, that is,

\[
u_0 = \int_a^b w(x)dx > 0, \quad \text{and} \quad \|p_n\| = \sqrt{\langle p_n, p_n \rangle} \tag{3.17}
\]
$p_n(x)$ is a polynomial of precise degree $n$, which is called *monic* if the coefficient of $x^n$ therein is equal to one. $\|p_n\|$ is called the norm of the polynomial $p_n(x)$ [40].

**Corollary 3.3.1.** Any polynomial $q(x)$, which is smooth by definition, is nonnegative on $[a, b]$, equivalently, nonnegative on $(a, b)$, and is not a zero polynomial,\(^1\) i.e., $q(x) \neq 0$, then the following holds,

$$
\int_a^b w(x)q(x)dx > 0
$$

(3.18)

and consequently, the norm of any polynomial except zero polynomial is positive. Moreover, zero polynomial, which has no degree, does not belong to the set of orthogonal polynomials, therefore the norm of an orthogonal polynomial, $p_n(x)$, is always positive, that is, $\|p_n\| > 0$ [41].

There exists a unique set of monic orthogonal polynomials $\{\pi_n(x)\}$, which can be constructed by *Gram-Schmidt Orthogonalization* of monomials [42],

$$1, x, x^2, \ldots, x^n, \ldots
$$

(3.19)

by letting $\pi_0(x) = 1$ by definition, and [43]

$$
\pi_n(x) = x^n - \sum_{k=0}^{n-1} \langle x^k, \pi_k \rangle \pi_k(x), \quad n \geq 1
$$

(3.20)

From the construction of monic orthogonal polynomials, we have the following,

**Lemma 3.3.2.** The set of monic orthogonal polynomials, $\{\pi_0, \pi_1, \ldots, \pi_n\}$, is linearly independent and any polynomial of degree $n$ or less can be uniquely represented by the linear combination of the set. In other words, $\{\pi_0, \pi_1, \ldots, \pi_n\}$ forms a basis of the space of all polynomials of degree $n$ or less [42]. These properties are actually extended to any set of orthogonal polynomials, $\{p_0, p_1, \ldots, p_n\}$.

\(^1\) Zero polynomial is a polynomial with all coefficients equal to zero, the rest of polynomials are nonzero polynomials. A polynomial usually refers to a nonzero polynomial.
Therefore, any $m^{th}$ degree polynomial $q_m(x)$ can be expanded in terms of the set of orthogonal polynomials \{p_0, p_1, ..., p_m\}, the orthogonal polynomial $p_n(x)$ will be orthogonal over $[a, b]$ to all polynomials of degree inferior to $n$ \cite{14}, that is,

$$
\int_a^b w(x)p_n(x)q_m(x)dx = 0, \quad \forall \quad 0 \leq m < n
$$

(3.21)

**Theorem 3.3.3.** The orthogonal polynomial $p_n(x)$, $n \geq 1$, has $n$ distinct real roots which are located in the interior of the support interval between $a$ and $b$, i.e., $(a, b)$.

**Proof.** Let $x_0, x_1, ..., x_{m-1}$ be the distinct roots of $p_n(x)$ in $(a, b)$ and $p_n(x)$ changes signs at those places, i.e., $p_n(x_i^-)p_n(x_i^+) < 0$, for $0 \leq i \leq m - 1$. Since $p_n(x)$ is an $n^{th}$ degree polynomial, $m$ only can be anywhere from zero to a maximum of $n$, i.e., $0 \leq m \leq n$. However, if $m < n$, by orthogonality,

$$
\int_a^b w(x)p_n(x)(x - x_0)(x - x_1)\cdots(x - x_{m-1})dx = 0
$$

(3.22)

which is impossible to be zero from Corollary 3.3.1 since the combined polynomial $p_n(x) \prod_{i=0}^{m-1}(x - x_i)$ in the integrand never changes signs in $(a, b)$. Therefore, $m$ must be equal to $n$, and consequently $p_n(x)$ has $n$ distinct real roots in $(a, b)$. \hfill \Box

**Theorem 3.3.4.** The $(n + 1) \times (n + 1)$ matrix

$$
A := \begin{bmatrix}
p_0(t_0) & \cdots & p_0(t_n) \\
\vdots & \ddots & \vdots \\
p_n(t_0) & \cdots & p_n(t_n)
\end{bmatrix}
$$

(3.23)

is nonsingular for mutually distinct arguments $t_i$, $i = 0, 1, ..., n$ \cite{41}.

**Theorem 3.3.5** (monic three-term recurrence). Let \{\pi_n(x), n \geq 0\} be the monic orthogonal polynomials with respect to the weighting function $w(x)$, and assume $\pi_{-1}(x) \equiv 0$, then we have the monic three-term recurrence relation as follows,
\[ \pi_0(x) \equiv 1, \quad \pi_1(x) = (x - a_0)\pi_0(x) \quad (3.24) \]
\[ \pi_{n+1}(x) = (x - a_n)\pi_n(x) - b_n\pi_{n-1}(x), \quad n \geq 0 \quad (3.25) \]

where,
\[ a_n = \frac{\langle x\pi_n, \pi_n \rangle}{\langle \pi_n, \pi_n \rangle}, \quad n \geq 0, \quad \text{and} \quad b_n = \frac{\langle \pi_n, \pi_n \rangle}{\langle \pi_{n-1}, \pi_{n-1} \rangle} > 0, \quad n \geq 1 \quad (3.26) \]

**Proof.** \( \pi_0(x) \equiv 1 \) by definition and \( \pi_1(x) \) is the exact formula derived by the above Gram-Schmidt orthogonalization from monomials, which satisfies the three-term recurrence if \( \pi_{-1}(x) \equiv 0 \). For \( n \geq 1 \), since \( \pi_{n+1}(x) - x\pi_n(x) \) is an \( n \)th degree polynomial, it can be expanded in terms of the set of orthogonal polynomials \( \{\pi_0, \pi_1, \ldots, \pi_n\} \), i.e.,
\[ \pi_{n+1}(x) - x\pi_n(x) = \sum_{k=0}^{n} \frac{\langle \pi_{n+1} - x\pi_n, \pi_k \rangle}{\langle \pi_k, \pi_k \rangle} \pi_k \quad (3.27) \]

Therefore, by properties of the inner product and orthogonality,
\[ \pi_{n+1}(x) - x\pi_n(x) = \sum_{k=0}^{n} \frac{\langle -x\pi_n, \pi_k \rangle}{\langle \pi_k, \pi_k \rangle} \pi_k \]
\[ = - \sum_{k=0}^{n} \frac{\langle \pi_n, x\pi_k \rangle}{\langle \pi_k, \pi_k \rangle} \pi_k = - \sum_{k=n-1}^{n} \frac{\langle \pi_n, x\pi_k \rangle}{\langle \pi_k, \pi_k \rangle} \pi_k \]
\[ = \frac{\langle \pi_n, x\pi_n \rangle}{\langle \pi_n, \pi_n \rangle} \pi_n - \frac{\langle \pi_n, x\pi_{n-1} \rangle}{\langle \pi_{n-1}, \pi_{n-1} \rangle} \pi_{n-1} \]
\[ = -a_n\pi_n - \frac{\langle \pi_n, x\pi_{n-1} - \pi_n + \pi_n \rangle}{\langle \pi_{n-1}, \pi_{n-1} \rangle} \pi_{n-1} \]
\[ = -a_n\pi_n - \frac{\langle \pi_n, \pi_n \rangle}{\langle \pi_{n-1}, \pi_{n-1} \rangle} \pi_{n-1} \]
\[ = -a_n\pi_n - b_n\pi_{n-1} \]

consequently,
\[ \pi_{n+1}(x) = (x - a_n)\pi_n(x) - b_n\pi_{n-1}(x), \quad n \geq 0. \]
Theorem 3.3.6 (Christoffel-Darboux formula). If $\pi_n(x)$ is a monic orthogonal polynomial, then the following equality holds for $n \geq 0$,

$$\frac{\pi_{n+1}(x)\pi_n(t) - \pi_n(x)\pi_{n+1}(t)}{x-t} = \langle \pi_n, \pi_n \rangle \sum_{k=0}^{n} \frac{\pi_k(x)\pi_k(t)}{\langle \pi_k, \pi_k \rangle}$$

(3.29)

and the following inequality holds for $n \geq 0$,

$$\pi_{n+1}'(x)\pi_n(x) - \pi_n'(x)\pi_{n+1}(x) = \langle \pi_n, \pi_n \rangle \sum_{k=0}^{n} \frac{\pi_k^2(x)}{\langle \pi_k, \pi_k \rangle} > 0$$

(3.30)

Proof. The equality holds for $n = 0$. From the monic three-term recurrence,

$$\prod_{m=k+1}^{n} b_m = \frac{\langle \pi_n, \pi_n \rangle \langle \pi_{n-1}, \pi_{n-1} \rangle \ldots \langle \pi_{k+1}, \pi_{k+1} \rangle}{\langle \pi_{n-1}, \pi_{n-1} \rangle \langle \pi_{n-2}, \pi_{n-2} \rangle \ldots \langle \pi_k, \pi_k \rangle} = \frac{\langle \pi_n, \pi_n \rangle}{\langle \pi_k, \pi_k \rangle}$$

(3.31)

which equals to 1, when $k = n$. Then for $n \geq 1$,

$$\pi_{n+1}(x)\pi_n(t) - \pi_n(x)\pi_{n+1}(t) = [(x-a_n)\pi_n(x) - b_n\pi_{n-1}(x)]\pi_n(t)$$

$$- [(t-a_n)\pi_n(t) - b_n\pi_{n-1}(t)]\pi_n(x)$$

$$= (x-t)\pi_n(x)\pi_n(t)$$

$$+ b_n[\pi_n(x)\pi_{n-1}(t) - \pi_{n-1}(x)\pi_n(t)]$$

$$\vdots$$

$$= (x-t) \sum_{k=0}^{n} \left( \prod_{m=k+1}^{n} b_m \right) \pi_k(x)\pi_k(t)$$

(3.32)

$$= \langle \pi_n, \pi_n \rangle \sum_{k=0}^{n} \frac{\pi_k(x)\pi_k(t)}{\langle \pi_k, \pi_k \rangle}$$

(3.33)

therefore, for $n \geq 0$

$$\frac{\pi_{n+1}(x)\pi_n(t) - \pi_n(x)\pi_{n+1}(t)}{x-t} = \langle \pi_n, \pi_n \rangle \sum_{k=0}^{n} \frac{\pi_k(x)\pi_k(t)}{\langle \pi_k, \pi_k \rangle}$$

(3.34)

and taking the limit $x \to t$ for $n \geq 0$, we have,
\[
\lim_{x \to t} \frac{\pi_{n+1}(x)\pi_n(t) - \pi_n(x)\pi_{n+1}(t)}{x - t} = \pi'_{n+1}(x)\pi_n(x) - \pi'_n(x)\pi_{n+1}(x) = \langle \pi_n, \pi_n \rangle \sum_{k=0}^{n} \frac{\pi_k^2(x)}{\langle \pi_k, \pi_k \rangle} > 0
\] (3.35)

because \( \langle \pi_k, \pi_k \rangle > 0 \), \( \pi_k^2(x) \geq 0 \) for \( k \geq 0 \) and \( \pi_0^2(x) = 1 > 0 \).

\begin{proof}
From the above Christoffel-Darboux formula, we have,

\[
\int_a^b w(x) \frac{\pi_{n+1}(x)\pi_n(t) - \pi_n(x)\pi_{n+1}(t)}{x - t} dx = \langle \pi_n, \pi_n \rangle \]
(3.37)

\end{proof}

**Corollary 3.3.7.** The integration of Christoffel-Darboux formula is the inner product of the monic orthogonal polynomial, \( \pi_n(x) \), and itself, that is,

\[
\int_a^b w(x) \frac{\pi_{n+1}(x)\pi_n(t) - \pi_n(x)\pi_{n+1}(t)}{x - t} dx = \langle \pi_n, \pi_n \rangle
\] (3.37)

Proof. From the above Christoffel-Darboux formula, we have,

\[
\int_a^b w(x) \frac{\pi_{n+1}(x)\pi_n(t) - \pi_n(x)\pi_{n+1}(t)}{x - t} dx = \int_a^b w(x) \langle \pi_n, \pi_n \rangle \sum_{k=0}^{n} \frac{\pi_k(x)\pi_k(t)}{\langle \pi_k, \pi_k \rangle} dx
\]

\[
= \langle \pi_n, \pi_n \rangle \int_a^b w(x) dx = \langle \pi_n, \pi_n \rangle
\]

\begin{proof}
From Rolle’s Theorem, we conclude the following corollary about roots of the derivative of the orthogonal polynomial,

**Corollary 3.3.9.** The roots of \( p'_{n+1}(x) \) interleave with those of \( p_{n+1}(x) \), \( n \geq 1 \), i.e.,

\[
p^{(0)}_{n+1} < p^{(0)}_{n+1} < p^{(1)}_{n+1} < p^{(1)}_{n+1} \cdots < p^{(n-1)}_{n+1} < p^{(n)}_{n+1}
\] (3.38)

where \( p^{(i)}_n \), \( 0 \leq i \leq n-1 \) are the distinct real roots of \( p_n(x) \) in ascending order [42].

From Rolle’s Theorem, we conclude the following corollary about roots of the derivative of the orthogonal polynomial,
where \( p^{(i)}_n, 0 \leq i \leq n - 1 \) are the distinct real roots of \( p_n(x) \) in ascending order and \( p'^{(i)}_n, 0 \leq i \leq n - 2 \) are the distinct real roots of \( p'_n(x) \) in ascending order.

**Theorem 3.3.10.** The monic orthogonal polynomial \( \pi_n(x) \), \( n \geq 1 \), is the determinant of the matrix, \((xI - J_n)\), that is

\[
\pi_n(x) = |xI - J_n| \tag{3.40}
\]

where \( J_n \) is a tridiagonal matrix as follows,

\[
J_n := \begin{bmatrix}
a_0 & \sqrt{b_1} & 0 & \cdots & 0 \\
\sqrt{b_1} & a_1 & \sqrt{b_2} & \cdots & 0 \\
0 & \sqrt{b_2} & a_2 & \cdots & 0 \\
\cdots & \cdots & \cdots & \cdots & \cdots \\
0 & 0 & \cdots & \sqrt{b_{n-2}} & a_{n-1} \\
0 & 0 & \cdots & 0 & \sqrt{b_{n-1}}
\end{bmatrix} \tag{3.41}
\]

**Proof.** We know that \( \pi_0(x) = 1 \), \( \pi_1(x) = x - a_0 \), and,

\[
|xI - J_2| = (x - a_1)(x - a_0) - b_1 = \pi_2(x) \tag{3.42}
\]

by the monic three-term recurrence. As shown, \( \pi_2(x) \) and \( \pi_1(x) \) satisfy (3.40).

Now assume (3.40) holds for \( \pi_{k-1}(x) \) and \( \pi_{k-2}(x) \), \( k \geq 3 \), then expanding the determinant along the last column of \((xI - J_k)\), that is,

\[
|xI - J_k| = -\sqrt{b_{k-1}} \sqrt{b_{k-1}} |xI - J_{k-2}| + (x - a_{k-1}) |xI - J_{k-1}|
\]

\[
= (x - a_{k-1})\pi_{k-1}(x) - b_{k-1} \pi_{k-2}(x) = \pi_k(x) \tag{3.43}
\]

from the monic three-term recurrence. Therefore (3.40) holds for \( n \geq 1 \). Consequently and surprisingly, the eigenvalues of \( J_n \) are the roots of \( \pi_n(x) \).

The following table gives the most common orthogonal polynomials and their corresponding integration intervals and weighting functions,
Table 3.1: Classical orthogonal polynomials and weighting functions

<table>
<thead>
<tr>
<th>Polynomials</th>
<th>Notation</th>
<th>Interval</th>
<th>Weighting function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chebyshev</td>
<td>$T_n(x)$</td>
<td>$[-1, 1]$</td>
<td>$\frac{1}{\sqrt{1-x^2}}$</td>
</tr>
<tr>
<td>Hermite</td>
<td>$H_n(x)$</td>
<td>$(-\infty, \infty)$</td>
<td>$e^{-x^2}$</td>
</tr>
<tr>
<td>Laguerre</td>
<td>$L_n(x)$</td>
<td>$[0, \infty)$</td>
<td>$e^{-x}$</td>
</tr>
<tr>
<td>Generalized Laguerre</td>
<td>$L_n^{(\alpha)}(x)$</td>
<td>$[0, \infty)$</td>
<td>$x^\alpha e^{-x}, \alpha &gt; -1$</td>
</tr>
<tr>
<td>Legendre</td>
<td>$P_n(x)$</td>
<td>$[-1, 1]$</td>
<td>$1$</td>
</tr>
<tr>
<td>Jacobi</td>
<td>$P_n^{(\alpha,\beta)}(x)$</td>
<td>$[-1, 1]$</td>
<td>$(1 - x)^\alpha (1 + x)^\beta, \alpha &gt; -1, \beta &gt; -1$</td>
</tr>
</tbody>
</table>

3.3.2 General Gauss Quadrature

Gauss quadrature rule, named after a German mathematician and scientist Johann Carl Friedrich Gauss, is a quadrature rule over finite or infinite interval $[a, b]$ constructed to yield an exact result for polynomials of degree up to $2n + 1$ by choosing a suitable set of $n + 1$ grid points $\{x_i\}$ on $(a, b)$ and $n + 1$ associated weights $\{w_i\}$, that is, if $f(x)$ is a polynomial of degree $2n + 1$ or less, then,

$$\int_a^b w(x)f(x)dx = \sum_{i=0}^n w_i f(x_i) \quad (3.44)$$

where $w(x)$ is the same weighting function as in the definition of orthogonal polynomials. Therefore, if an arbitrary function $f(x)$ is approximated well enough by a finite high-degree polynomial, Gauss quadrature rule guarantees its integral with respect to $w(x)$ over $[a, b]$ will be nearly equal to the weighted summation.

The brute-force approach to determine the weights and grid points of a Gaussian quadrature for a particular weighting function is to solve the following $2n + 2$ simultaneous nonlinear equations as shown below. It can be approved that following set of nonlinear equations has a unique solution as long as the weighting function $w(x)$ with
respect to \([a, b]\) follows the conditions in the definition of orthogonal polynomials \([45]\).

\[
\int_a^b w(x)dx = \sum_{i=0}^{n} w_i \\
\int_a^b w(x)xdx = \sum_{i=0}^{n} w_ix_i \\
\int_a^b w(x)x^2dx = \sum_{i=0}^{n} w_ix_i^2 \\
\vdots \\
\int_a^b w(x)x^{2n+1}dx = \sum_{i=0}^{n} w_ix_i^{2n+1}
\]

(3.45)

However, to solve such a nonlinear system of (3.45) generally is extremely difficult. The classical approach is based on the theory of orthogonal polynomials as discussed in great detail in Section 3.3.1.

Gauss quadrature is actually a class of numerical integration techniques, and each one is specified for a unique type of integral with a specific weighting function. All Gauss quadrature rules are based on the following fundamental theorem \([41]\),

**Theorem 3.3.11.** Let \(x_0, x_1, ..., x_n\) be the distinct real roots in \((a, b)\) of an \((n + 1)\)th orthogonal polynomial \(p_{n+1}(x)\) over finite or infinite interval \([a, b]\), similar to the Newton-Cotes formula, the Lagrange interpolation polynomials based on those root points are,

\[
\ell_i(x) = \prod_{\substack{j=0 \atop j \neq i}}^{n} \frac{x - x_j}{x_i - x_j}, \quad i = 0, 1, \cdots, n
\]

(3.46)

and let the weight \(w_i\) determined by,

\[
w_i = \int_a^b w(x)\ell_i(x)dx
\]

(3.47)
then, the following equation holds exactly,

\[ \int_{a}^{b} w(x)f(x)dx = \sum_{i=0}^{n} w_i f(x_i) \]  \hfill (3.48)

for any polynomial \( f(x) \) of degree up to \( 2n + 1 \) and the weights \( \{w_i\} \) are positive.

**Proof.** Suppose \( f(x) \) is a polynomial of degree \( 2n + 1 \) or less, which is divided by the orthogonal polynomial \( p_{n+1}(x) \) as follows,

\[ f(x) = p_{n+1}(x)q(x) + r(x) \]  \hfill (3.49)

where both \( q(x) \) and \( r(x) \) are polynomials of degree \( n \) or less, therefore,

\[ f(x_i) = p_{n+1}(x_i)q(x_i) + r(x_i) = r(x_i), \quad i = 0, 1, \ldots, n \]  \hfill (3.50)

and \( r(x) \) can be exactly interpolated by the Lagrange polynomials,

\[ r(x) = \sum_{i=0}^{n} r(x_i) \ell_i(x) = \sum_{i=0}^{n} f(x_i) \ell_i(x) \]  \hfill (3.51)

then, because of orthogonality and the above properties,

\[ \int_{a}^{b} w(x)f(x)dx = \int_{a}^{b} w(x)p_{n+1}(x)q(x)dx + \int_{a}^{b} w(x)r(x)dx \]

\[ = 0 + \int_{a}^{b} w(x)r(x)dx \]

\[ = \int_{a}^{b} w(x) \sum_{i=0}^{n} f(x_i) \ell_i(x)dx \]

\[ = \sum_{i=0}^{n} f(x_i) \int_{a}^{b} w(x) \ell_i(x)dx \]

\[ = \sum_{i=0}^{n} w_i f(x_i) \]  \hfill (3.52)
and let \( g(x) \) be a polynomial of degree \( 2n + 2 \) as follows,

\[
g(x) = \prod_{j=0}^{n} (x - x_j)^2 \tag{3.53}
\]

then, because \( g(x) \) is a nonnegative polynomial,

\[
0 < \int_{a}^{b} w(x)g(x)dx = \sum_{i=0}^{n} w_i g(x_i) = 0 \tag{3.54}
\]

therefore (3.48) holds exactly for any polynomial of degree up to \( 2n + 1 \).

To prove the weight \( w_i \) is positive, let \( h_i(x) \) be as follows,

\[
h_i(x) = \prod_{\substack{j=0 \atop j \neq i}}^{n} (x - x_j)^2, \quad i = 0, 1, \cdots, n \tag{3.55}
\]

then, since \( h_i(x) \) is a nonnegative polynomial of degree \( 2n \),

\[
0 < \int_{a}^{b} w(x)h_i(x)dx = \sum_{k=0}^{n} w_k h_i(x_k) = w_i h_i(x_i) \tag{3.56}
\]

the roots are real and distinct, so \( h_i(x_i) > 0 \), therefore \( w_i \) is indeed positive. \qed

Using the similar technique as in the proof of Theorem 3.3.11, and replacing grid points \( \{x_i\} \) with arbitrary grid points \( \{\xi_i\} \) in (3.53) and (3.54), we have,

**Theorem 3.3.12.** No matter how we choose grid points and the associated weights, (3.48) cannot hold exact for all the polynomials of degree \( 2n + 2 \), that is, no quadrature formula has order accuracy higher than (3.48) \[46\].

**Theorem 3.3.13.** If \( f(x) \in C^{2n+2}[a, b] \), apply the mean-value theorem, then

\[
\int_{a}^{b} w(x)f(x)dx = \sum_{i=0}^{n} w_i f(x_i) + \frac{f^{(2n+2)}(\xi)}{(2n+2)!}(\pi_{n+1}, \pi_{n+1}) \tag{3.57}
\]

for some \( \xi \in (a, b) \) \[41\].
3.3.3 Calculations of Gauss Quadrature

To implement the Gauss quadrature rule, we need to know the grid points and weights for the particular set of orthogonal polynomials. From Theorem 3.3.11, we know the grid points are the roots of an \((n + 1)\)th orthogonal polynomial, so we can use Newton-Raphson method to find those roots if we choose the starting points correctly, alternatively, we can use Theorem 3.3.10 to find the eigenvalues of \(J_{n+1}\), which will be the roots of the orthogonal polynomial \(p_{n+1}(x)\). The associated weights can be calculated by the following corollary.

**Corollary 3.3.14.** The weights in the Gauss quadrature can be expressed in terms of the monic orthogonal polynomials as follows,

\[
 w_i = \frac{\langle \pi_n, \pi_n \rangle}{\pi'_{n+1}(x_i)\pi_n(x_i)}, \quad i = 0, 1, \ldots, n
\]  

(3.58)

where \(\{x_i\}\) are the distinct roots of the monic orthogonal polynomial \(\pi_{n+1}(x)\).

**Proof.** From Theorem 3.3.11, we know the weight \(w_i\) is,

\[
 w_i = \int_a^b w(x)\ell_i(x)dx, \quad i = 0, 1, \ldots, n
\]  

(3.59)

where

\[
 \ell_i(x) = \prod_{\substack{j=0\atop j \neq i}}^n \frac{x - x_j}{x_i - x_j}, \quad i = 0, 1, \ldots, n
\]  

(3.60)

As we know,

\[
 \pi_{n+1}(x) = (x - x_0)(x - x_1) \cdots (x - x_n) = \prod_{j=0}^n (x - x_j)
\]  

(3.61)

\[
 \pi'_{n+1}(x_i) = (x_i - x_0) \cdots (x_i - x_{i-1})(x_i - x_{i+1}) \cdots (x - x_n)
\]  

\[
 = \prod_{\substack{j=0\atop j \neq i}}^n (x_i - x_j)
\]  

(3.62)
so, it is easy to see that,

$$\ell_i(x) = \frac{\pi_{n+1}(x)}{(x - x_i)\pi'_{n+1}(x_i)}$$  \hspace{1cm} (3.63)

Since $\pi_{n+1}(x_i) = 0$, from Christoffel-Darboux formula in Theorem 3.3.6 and its integration formula in Corollary 3.3.7, we have the following,

$$w_i = \int_{\alpha}^{\beta} w(x) \frac{\pi_{n+1}(x)}{(x - x_i)\pi'_{n+1}(x_i)} dx$$

$$= \frac{1}{\pi'_{n+1}(x_i)\pi_n(x_i)} \int_{\alpha}^{\beta} w(x) \frac{\pi_{n+1}(x)\pi_n(x_i)}{(x - x_i)} dx$$

$$= \frac{1}{\pi'_{n+1}(x_i)\pi_n(x_i)} \int_{\alpha}^{\beta} w(x) \frac{\pi_{n+1}(x)\pi_n(x_i) - \pi_n(x)\pi_{n+1}(x_i)}{(x - x_i)} dx$$

$$= \frac{\langle \pi_n, \pi_n \rangle}{\pi'_{n+1}(x_i)\pi_n(x_i)}, \quad i = 0, 1, \ldots, n$$  \hspace{1cm} (3.64)

Therefore, we can calculate the weights according to the above formula. \qed

**Corollary 3.3.15.** The weights in the Gauss quadrature satisfy the following,

$$\frac{1}{w_i} = \sum_{k=0}^{n} \frac{p_k(x_i)p_k(x_i)}{\langle p_k, p_k \rangle}, \quad i = 0, 1, \ldots, n$$  \hspace{1cm} (3.65)

where $\{x_i\}$ are the distinct roots of the orthogonal polynomial $p_{n+1}(x)$.

**Proof.** From Corollary 3.3.14 and the Christoffel-Darboux formula,

$$\frac{1}{w_i} = \frac{\pi'_{n+1}(x_i)\pi_n(x_i)}{\langle \pi_n, \pi_n \rangle}$$

$$= \frac{\pi'_{n+1}(x_i)\pi_n(x_i) - \pi_n'(x_i)\pi_{n+1}(x_i)}{\langle \pi_n, \pi_n \rangle}$$

$$= \sum_{k=0}^{n} \frac{\pi_k(x_i)p_k(x_i)}{\langle \pi_k, \pi_k \rangle} = \sum_{k=0}^{n} \frac{p_k(x_i)p_k(x_i)}{\langle p_k, p_k \rangle}, \quad i = 0, 1, \ldots, n$$ \qed
Alternatively, given the monic three-term recurrence relation for the orthogonal polynomials with respect to the weighting function, the Gauss quadrature rule may be generated by computing the eigenvalues and first component of the orthonormalized eigenvectors of a symmetric tridiagonal matrix [47, 48, 49].

**Lemma 3.3.16** (discrete orthogonality). For $0 \leq i, j \leq n$, and $x_0, x_1, ..., x_n$ are the distinct roots of an orthogonal polynomial of degree $n + 1$, i.e., $p_{n+1}(x)$, we have,

$$
\sum_{k=0}^{n} w_k p_i(x_k)p_j(x_k) = \langle p_i, p_j \rangle = \|p_i\| \cdot \|p_j\| \cdot \delta_{ij}
$$  

(3.66)

**Proof.** Since $p_i(x)p_j(x)$, $0 \leq i, j \leq n$, is a polynomial of degree less than $2n + 1$,

$$
\langle p_i, p_j \rangle = \|p_i\| \cdot \|p_j\| \cdot \delta_{ij} = \int_{a}^{b} w(x)p_i(x)p_j(x)dx = \sum_{k=0}^{n} w_k p_i(x_k)p_j(x_k)
$$

\[ \square \]

**Lemma 3.3.17** (dual orthogonality). For $0 \leq i, j \leq n$, and $x_0, x_1, ..., x_n$ are the distinct roots of an orthogonal polynomial of degree $n + 1$, i.e., $p_{n+1}(x)$, we have,

$$
\sum_{k=0}^{n} \frac{p_k(x_i)p_k(x_j)}{\langle p_k, p_k \rangle} = \frac{\delta_{ij}}{\sqrt{w_iw_j}}
$$  

(3.67)

**Proof.** Let $Q_{ij}$, $0 \leq i, j \leq n$, be the entries of the matrix $Q$, and define,

$$
Q_{ij} = \frac{\sqrt{w_ip_j(x_i)}}{\|p_j\|}
$$  

(3.68)

then we have,

$$
[Q^TQ]_{i,j} = \sum_{k=0}^{n} Q_{kj}Q_{kj}
$$

37
\[
= \sum_{k=0}^{n} \frac{\sqrt{w_k} p_i(x_k)}{\|p_k\|} \frac{\sqrt{w_k} p_j(x_k)}{\|p_j\|}
= \sum_{k=0}^{n} w_k p_i(x_k) p_j(x_k) \frac{1}{\|p_i\| \cdot \|p_j\|} = \delta_{ij}
\]

therefore, \( Q^T Q = I \), on the other hand, \( QQ^T = I \) too, then,

\[
\begin{bmatrix} QQ^T \end{bmatrix}_{ij} = \sum_{k=0}^{n} Q_{ik} Q_{jk}
= \sum_{k=0}^{n} \frac{\sqrt{w_k} p_i(x_i) \sqrt{w_k} p_j(x_j)}{\|p_k\| \cdot \|p_k\|}
= \sum_{k=0}^{n} \frac{\sqrt{w_k} w_j p_i(x_i) p_j(x_j)}{\langle p_k, p_k \rangle} = \delta_{ij}
\]

therefore,

\[
\sum_{k=0}^{n} \frac{p_k(x_i) p_k(x_j)}{\langle p_k, p_k \rangle} = \frac{\delta_{ij}}{\sqrt{w_i w_j}}
\]

consequently, it also leads to the result of Corollary 3.3.15. \( \square \)

**Corollary 3.3.18** (orthonormal three-term recurrence). *The monic three-term recurrence relation can be rewritten as a three-term recurrence in terms of the orthonormal polynomials, \( \varphi_n(x) \), \( n \geq 0 \), as follows,*

\[
\sqrt{b_{n+1}} \varphi_{n+1}(x) = (x - a_n) \varphi_n(x) - \sqrt{b_n} \varphi_{n-1}(x), \quad n \geq 0
\]

*where,*

\[
\varphi_n(x) = \frac{\pi_n(x)}{\|\pi_n\|}, \quad n \geq 0, \quad \text{and} \quad \varphi_{-1}(x) = 0
\]

*Proof.* When \( n = 0 \), the equality holds. For \( n \geq 1 \), from the monic three-term recurrence, we have the following,

\[
\varphi_{n+1}(x) \|\pi_{n+1}\| = (x - a_n) \varphi_n(x) \|\pi_n\| - b_n \varphi_{n-1}(x) \|\pi_{n-1}\|
\]
Since \( b_n = \frac{\langle \pi_n, \pi_n \rangle}{\langle \pi_{n-1}, \pi_{n-1} \rangle} = \frac{\| \pi_n \|^2}{\| \pi_{n-1} \|^2} \), then we have,

\[
\sqrt{b_{n+1}} \varphi_{n+1}(x) = (x - a_n) \varphi_n(x) - \sqrt{b_n} \varphi_{n-1}(x)
\]

**Theorem 3.3.19** (the eigenvalue method). If \( a_n, n \geq 0 \) and \( b_n, n \geq 1 \) are known in the monic three-term recurrence relation for the orthogonal polynomials, then the weights \( w_i \) and the grid points \( x_i \), \( 0 \leq i \leq n \), can be obtained from the eigenvalue decomposition of the symmetric tridiagonal Jacobi matrix \( J_{n+1} \) as in Theorem 3.3.10. The eigenvalue decomposition of the symmetric matrix \( J_{n+1} \) is as follows,

\[
J_{n+1} = \begin{bmatrix}
    a_0 & \sqrt{b_1} & & \\
    \sqrt{b_1} & a_1 & \sqrt{b_2} & \\
           & \ddots & \ddots & \ddots \\
    \sqrt{b_n} & a_{n-1} & \sqrt{b_n} & a_n
\end{bmatrix} = V \Lambda V^T \tag{3.75}
\]

where \( \Lambda \) is the diagonal matrix of eigenvalues \( \{ \lambda_i, 0 \leq i \leq n \} \), that is,

\[
\Lambda = \begin{bmatrix}
    \lambda_0 \\
    \lambda_1 \\
    \vdots \\
    \lambda_n
\end{bmatrix}, \quad \text{where} \quad \lambda_0 < \lambda_1 < \cdots < \lambda_n \tag{3.76}
\]

and \( V \) is the orthogonal matrix of eigenvectors, i.e., \( V^TV = I \). Each column vector \( v_i \) of \( V \) is the normalized eigenvector respect to the corresponding eigenvalue \( \lambda_i \), i.e.,

\[
V = \begin{bmatrix}
    v_0 & v_1 & \cdots & v_n
\end{bmatrix}, \quad \text{where} \quad v_i^T v_j = \delta_{ij}, \quad 0 \leq i, j \leq n \tag{3.77}
\]

Then, \( x_i = \lambda_i \), and \( w_i = u_0 [v_i]_0^3 \), where \( u_0 \) is the first moment of the orthogonal polynomial and \( [v_i]_0 \) is the first element of the orthonormal eigenvector \( v_i \).

**Proof.** Rearrange (3.72) in Corollary 3.3.18, we have the following equation,

\[
x_k \varphi_k(x) = \sqrt{b_{k+1}} \varphi_{k+1}(x) + a_k \varphi_k(x) + \sqrt{b_k} \varphi_{k-1}(x), \quad k \geq 0 \tag{3.78}
\]
If we have \( k = 0, 1, ..., n \), a system of \((n + 1)\) equations will be,

\[
x \cdot \begin{bmatrix} \varphi_0(x) \\ \varphi_1(x) \\ \vdots \\ \varphi_{n-1}(x) \\ \varphi_n(x) \end{bmatrix} = \begin{bmatrix} a_0 & \sqrt{b_1} & \sqrt{b_2} & \cdots & a_{n-1} & \sqrt{b_n} \\ \sqrt{b_1} & a_1 & \sqrt{b_2} & \cdots & \sqrt{b_{n-1}} & a_n \\ \sqrt{b_2} & \sqrt{b_2} & \cdots & \sqrt{b_{n-1}} & \sqrt{b_n} & \sqrt{b_{n+1}} \end{bmatrix} \begin{bmatrix} \varphi_0(x) \\ \varphi_1(x) \\ \vdots \\ \varphi_{n-1}(x) \\ \varphi_n(x) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}
\]

that is,

\[
x \Psi(x) = J_{n+1} \Psi(x) + \Upsilon(x) \quad (3.79)
\]

where,

\[
\Psi(x) = \begin{bmatrix} \varphi_0(x) \\ \varphi_1(x) \\ \vdots \\ \varphi_{n-1}(x) \\ \varphi_n(x) \end{bmatrix}, \quad \text{and} \quad \Upsilon(x) = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}
\]

let \( x = x_i \), the root of \( \pi_{n+1}(x) \), which is also the root of \( \varphi_{n+1}(x) \), then,

\[
x_i \Psi(x_i) = J_{n+1} \Psi(x_i) \quad (3.80)
\]

hence \( x_i \) is an eigenvalue of \( J_{n+1} \) with eigenvector \( \Psi(x_i) \) and from Corollary 3.3.15,

\[
\Psi(x_i)^T \Psi(x_i) = \sum_{k=0}^{n} \frac{p_k(x_i)p_k(x_i)}{\|p_k\| \cdot \|p_k\|} = \sum_{k=0}^{n} \frac{p_k(x_i)p_k(x_i)}{\langle p_k, p_k \rangle} = \frac{1}{w_i} \quad (3.81)
\]

therefore, the orthonormal eigenvector \( v_i = \pm \sqrt{w_i} \Psi(x_i) \), then

\[
w_i = \frac{[v_i]_0^2}{\varphi_0^2(x_i)} = \frac{||\pi_0||^2}{\pi_0^2} [v_i]_0^2 = \langle \pi_0, \pi_0 \rangle [v_i]_0^2 = u_0[v_i]_0^2 \quad (3.82)
\]

where \( [v_i]_0 \) is the first element of the orthonormal vector \( v_i \). \( \square \)
3.4 Classical Gauss Quadratures

3.4.1 Gauss-Chebyshev Quadrature

Definitions

The Chebyshev polynomials \( T_n(x) \), \( n \geq 0 \), named after Pafnuty Lvovich Chebyshev (May 16, 1821 - December 8, 1894), a Russian mathematician, are orthogonal polynomials with respect to the weighting function \( w(x) = 1/\sqrt{1-x^2} \) on \([-1, 1]\), i.e.,

\[
\langle T_n, T_m \rangle = \int_{-1}^{1} \frac{1}{\sqrt{1-x^2}} T_n(x)T_m(x) dx = \begin{cases} 
0 & \text{if } m \neq n \\
\pi & \text{if } m = n = 0 \\
\pi/2 & \text{if } m = n \neq 0
\end{cases} \tag{3.83}
\]

Differential Equations

The Chebyshev polynomials satisfy the Chebyshev differential equation [50],

\[
(1-x^2)T''_n(x) - xT'_n(x) + n^2T_n(x) = 0, \quad n \geq 0 \tag{3.84}
\]

which is a special case of the Sturm-Liouville differential equation as follows,

\[
\left[ \sqrt{1-x^2} T'_n(x) \right]' + \frac{n^2}{\sqrt{1-x^2}} T_n(x) = 0, \quad n \geq 0 \tag{3.85}
\]

Rodrigues’ Formula

Rodrigues’ formula for the Chebyshev polynomials is [50],

\[
T_n(x) = 2^n n! \sqrt{1-x^2} \frac{d^n}{dx^n} \left[ (1-x^2)^{n-\frac{1}{2}} \right] \tag{3.86}
\]

Proof of Inner Product

It can be shown that, for \( x \in [-1, 1] \),

\[
T_n(x) = \cos(n\theta) = \cos[n \cos^{-1}(x)], \quad \text{where} \quad \theta = \cos^{-1}(x) \tag{3.87}
\]

then, the inner product of \( T_n \) and \( T_m \),

41
\[ \langle T_n, T_m \rangle = \int_{\pi}^{0} \frac{\cos(n\theta) \cos(m\theta)}{\sqrt{1 - \cos^2(\theta)}} \, (-\sin \theta) \, d\theta \]
\[ = \int_{0}^{\pi} \cos(n\theta) \cos(m\theta) \, d\theta \]
\[ = \begin{cases} 
0 & \text{if } m \neq n \\
\pi & \text{if } m = n = 0 \\
\pi/2 & \text{if } m = n \neq 0
\end{cases} \quad (3.88) \]

**First Few Polynomial Formulas**

The first few Chebyshev polynomials are,

\[ T_0(x) = 1 \]
\[ T_1(x) = x \]
\[ T_2(x) = 2x^2 - 1 \]
\[ T_3(x) = 4x^3 - 3x \]
\[ T_4(x) = 8x^4 - 8x^2 + 1 \]
\[ T_5(x) = 16x^5 - 20x^3 + 5x \]
\[ T_6(x) = 32x^6 - 48x^4 + 18x^2 - 1 \quad (3.89) \]

**Polynomial Recurrence Relations**

The Chebyshev polynomials can be generated by the following three-term recurrence,

\[ T_0(x) = 1 \quad (3.90) \]
\[ T_1(x) = x T_0(x) = x \quad (3.91) \]
\[ \vdots \]
\[ T_{n+1}(x) = 2x T_n(x) - T_{n-1}(x), \quad n \geq 1 \quad (3.92) \]

**Proof.** It is easy to see \( T_0(x) \) and \( T_1(x) \) satisfy (3.87), and for \( n \geq 1 \), we have,

\[ T_{n+1}(x) = \cos((n+1)\theta) \]
\[ = \cos(n\theta) \cos(\theta) - \sin(n\theta) \sin(\theta) \]
\[ = T_n(x) \cdot x - \sin(n\theta) \sin(\theta) \]
\[ \quad (3.93) \]
therefore,

\[ T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x) \]

**Derivative Recurrence Relations**

Since the derivative of the Chebyshev polynomial is,

\[ T'_n(x) = \frac{n \sin(n\theta)}{\sin(\theta)}, \quad n \geq 0 \quad (3.94) \]

it is easy to verify that the three-term recurrence for the derivative will be [39],

\[
\begin{align*}
T'_0(x) &= 0 \\
T'_1(x) &= 1 \\
\vdots \\
(1 - x^2)T'_{n+1}(x) &= -(n + 1)xT_{n+1}(x) + (n + 1)T_n(x), \quad n \geq 0 \quad (3.97)
\end{align*}
\]

The Chebyshev polynomials and derivatives also satisfy the following relationship,

\[
\begin{align*}
2T'_0(x) &= 0 \\
2T'_1(x) &= \frac{1}{2}T'_2(x) \\
\vdots \\
2T'_n(x) &= \frac{1}{n + 1}T'_{n+1}(x) - \frac{1}{n - 1}T'_{n-1}(x), \quad n \geq 2 \quad (3.100)
\end{align*}
\]

**Symmetries and Special Values**

From the three-term recurrence relation of the Chebyshev polynomials, it is easy to see the symmetry property of \( T_n(x) \) as follows,

\[ T_n(x) = (-1)^nT_n(-x), \quad n \geq 0 \quad (3.101) \]

and the values of the Chebyshev polynomials at special points are as follows,

\[ T_n(-1) = (-1)^n, \quad T_n(0) = \cos \frac{n\pi}{2}, \quad T_n(1) = 1, \quad n \geq 0 \quad (3.102) \]

43
The Eigenvalue Method

From the three-term recurrence relation of the Chebyshev polynomials, it is also easy to see the coefficient of $x^n$ in $T_n(x)$ is $2^{n-1}$ for $n \geq 1$, while $T_0 = 1$, that is,

$$\pi_n(x) = T_n(x)/2^{n-1}, \quad n \geq 1, \quad \text{while} \quad \pi_0(x) = T_0(x) = 1 \quad (3.103)$$

then, the three-term recurrence can be converted to the monic recurrence as follows,

$$\pi_0(x) = 1 \quad (3.104)$$
$$\pi_1(x) = x\pi_0(x) = x \quad (3.105)$$
$$\pi_2(x) = x\pi_1(x) - \frac{1}{2}\pi_0(x) = x^2 - \frac{1}{2} \quad (3.106)$$
$$\vdots$$
$$\pi_{n+1}(x) = x\pi_n(x) - \frac{1}{4}\pi_{n-1}(x), \quad n \geq 2 \quad (3.107)$$

The general formula of the monic three-term recurrence is as follows,

$$\pi_{n+1}(x) = (x - a_n)\pi_n(x) - b_n\pi_{n-1}(x), \quad n \geq 0 \quad (3.108)$$

hence, we have the $a_n$ and $b_n$ as follows,

$$a_n = 0, \quad n \geq 0, \quad \text{and} \quad b_1 = 1/2, \quad b_n = 1/4, \quad n \geq 2 \quad (3.109)$$

while the first moment is,

$$u_0 = \langle \pi_0, \pi_0 \rangle = \langle T_0, T_0 \rangle = \pi \quad (3.110)$$

Then, the grid points $\{x_i\}$ and the associated weights $\{w_i\}$ for the Gauss-Chebyshev quadrature can be computed by implementing Theorem 3.3.19.

The Analytical Formulas

The grid points, which are the roots of $T_{n+1}(x), \quad n \geq 0$, are,

$$x_i = \cos \left( \frac{2i + 1}{2n + 2} \pi \right), \quad i = 0, 1, \ldots, n \quad (3.111)$$
because as we can see $-1 \leq x_i \leq 1$, and so,

$$T_{n+1}(x_i) = \cos \left( \frac{2i + 1}{2} \pi \right) = 0$$  \hspace{1cm} (3.112)

From the dual orthogonality of Lemma 3.3.17, we have,

$$\sum_{k=0}^{n} \frac{T_k(x_i)T_k(x_i)}{\langle T_k, T_k \rangle} = \frac{1}{w_i}, \quad n \geq 0$$  \hspace{1cm} (3.113)

Therefore, the weights for $T_{n+1}(x)$ can be derived as follows,

$$\frac{1}{w_i} = \frac{2}{\pi} \sum_{k=0}^{n} T_k(x_i)T_k(x_i) - \frac{1}{\pi}$$

$$= \frac{2}{\pi} \sum_{k=0}^{n} \cos^2 \left( \frac{2i + 1}{2n + 2} k\pi \right) - \frac{1}{\pi}$$

$$= \frac{2}{\pi} \sum_{k=0}^{n+1} \cos^2 \left( \frac{2i + 1}{2n + 2} k\pi \right) - \frac{1}{\pi}$$

$$= \frac{1}{\pi} \sum_{k=0}^{n+1} \left[ \cos^2 \left( \frac{2i + 1}{2n + 2} k\pi \right) + \cos^2 \left( \frac{2i + 1}{2n + 2} (n + 1 - k)\pi \right) \right] - \frac{1}{\pi}$$

$$= \frac{1}{\pi} \sum_{k=0}^{n+1} \left[ \cos^2 \left( \frac{2i + 1}{2n + 2} k\pi \right) + \sin^2 \left( \frac{2i + 1}{2n + 2} k\pi \right) \right] - \frac{1}{\pi} = \frac{n + 1}{\pi}$$  \hspace{1cm} (3.114)

hence, the associated weights for the roots of $T_{n+1}(x)$,

$$w_i = \frac{\pi}{n + 1}, \quad 0 \leq i \leq n$$  \hspace{1cm} (3.115)

Alternatively, we can use (3.58) in Corollary 3.3.14, for $n = 0$,

$$w_0 = \frac{\langle \pi_0, \pi_0 \rangle}{\pi'(x_i)\pi_0(x_i)} = \pi$$  \hspace{1cm} (3.116)

and for $n \geq 1$,

$$w_i = \frac{\langle \pi_n, \pi_n \rangle}{\pi'_n(x_i)\pi_n(x_i)} = \frac{\pi}{T'_{n+1}(x_i)T_n(x_i)}$$  \hspace{1cm} (3.117)
and since,

\[ T'_{n+1}(x_i)T_n(x_i) = \frac{(n + 1) \sin[(n + 1)\theta_i]}{\sin(\theta_i)} \cos(n\theta_i) \]
\[ = (n + 1) \frac{\sin[(2n + 1)\theta_i] + \sin \theta_i}{2\sin(\theta_i)} \]
\[ = (n + 1) \frac{\sin(\theta_i) + \sin \theta_i}{2\sin(\theta_i)} = n + 1 \]

(3.118)

therefore, not surprisingly, \( w_i = \frac{\pi}{n + 1}, \ 0 \leq i \leq n. \)

3.4.2 Gauss-Hermite Quadrature

Definitions

The Hermite polynomials \( \{H_n(x), \ n \geq 0\} \), named after Charles Hermite (December 24, 1822 - January 14, 1901), a French mathematician, are the orthogonal polynomials with respect to the weighting function \( w(x) = e^{-x^2} \) on \((-\infty, \infty)\) as follows,

\[ \langle H_n, H_m \rangle = \int_{-\infty}^{\infty} e^{-x^2} H_n(x)H_m(x)dx = \sqrt{\pi}2^n n! \delta_{nm} \]

(3.119)

Differential Equations

The Hermite polynomials satisfy the Hermite differential equation [50],

\[ H''_n(x) - 2xH'_n(x) + 2nH_n(x) = 0, \quad n \geq 0 \]

(3.120)

which is a special case of the Sturm-Liouville differential equation as follows,

\[ \left[ e^{-x^2} H'_n(x) \right]' + 2ne^{-x^2} H_n(x) = 0, \quad n \geq 0 \]

(3.121)

Rodrigues’ Formula

Rodrigues’ formula for the Hermite polynomials is [50],

\[ H_n(x) = (-1)^n e^{x^2} \frac{d^n}{dx^n} e^{-x^2} \]

(3.122)
First Few Polynomial Formulas

The first few Hermite polynomials are,

\[ H_0(x) = 1 \]
\[ H_1(x) = 2x \]
\[ H_2(x) = 4x^2 - 2 \]
\[ H_3(x) = 8x^3 - 12x \]
\[ H_4(x) = 16x^4 - 48x^2 + 12 \]
\[ H_5(x) = 32x^5 - 160x^3 + 120x \]
\[ H_6(x) = 64x^6 - 480x^4 + 720x^2 - 120 \] (3.123)

Polynomial Recurrence Relations

The Hermite polynomials satisfy the following three-term recurrence relation [50],

\[ H_0(x) = 1 \]
\[ H_1(x) = 2xH_0(x) = 2x \] (3.124)
\[ : \]
\[ H_{n+1}(x) = 2xH_n(x) - 2nH_{n-1}(x), \quad n \geq 0 \] (3.125)

Derivative Recurrence Relations

The recurrence relations for the derivatives are [50],

\[ H'_n(x) = 2nH_{n-1}(x), \quad n \geq 0 \] (3.127)
\[ H'_n(x) = 2xH_n(x) - H_{n+1}(x), \quad n \geq 0 \] (3.128)

(3.128) can be easily verified from (3.127) and the three-term recurrence (3.126).

Symmetries and Special Values

From the three-term recurrence relation of the Hermite polynomials, it is easy to see the symmetry property of \( H_n(x) \) as follows,

\[ H_n(x) = (-1)^n H_n(-x), \quad n \geq 0 \] (3.129)
and the values of the Hermite polynomials at the point of zero are as follows,

\[ H_{2m}(0) = \frac{(-1)^m(2m)!}{m!}, \quad H_{2m+1}(0) = 0, \quad m \geq 0 \quad (3.130) \]

**The Eigenvalue Method**

From the three-term recurrence relation of the Hermite polynomials, it is also easy to see the coefficient of \(x^n\) in \(H_n(x)\) is \(2^n\), \(n \geq 0\), that is,

\[ \pi_n(x) = H_n(x)/2^n, \quad n \geq 0 \quad (3.131) \]

then, the three-term recurrence can be converted to the monic recurrence as follows,

\[
\begin{align*}
\pi_0(x) &= 1 \quad (3.132) \\
\pi_1(x) &= x\pi_0(x) = x \quad (3.133) \\
\vdots \\
\pi_{n+1}(x) &= x\pi_n(x) - \frac{n}{2}\pi_{n-1}(x), \quad n \geq 0 \quad (3.134)
\end{align*}
\]

The general formula of the monic three-term recurrence is as follows,

\[ \pi_{n+1}(x) = (x - a_n)\pi_n(x) - b_n\pi_{n-1}(x), \quad n \geq 0 \quad (3.135) \]

hence, we have the \(a_n\) and \(b_n\) as follows,

\[ a_n = 0, \quad n \geq 0, \quad \text{and} \quad b_n = n/2, \quad n \geq 1 \quad (3.136) \]

while the first moment is,

\[ u_0 = \langle \pi_0, \pi_0 \rangle = \langle H_0, H_0 \rangle = \sqrt{\pi} \quad (3.137) \]

Then, the grid points \(\{x_i\}\) and the associated weights \(\{w_i\}\) for the Gauss-Hermite quadrature can be computed by implementing Theorem 3.3.19.
The Analytical Formulas

In Gauss-Hermite quadrature, the grid points will be the roots of $H_{n+1}(x)$, $n \geq 0$. For the weights, we can solve them from (3.58) in Corollary 3.3.14, i.e., for $n \geq 0$,

\[
    w_i = \frac{\langle \pi_n, \pi_n \rangle}{\pi_{n+1}(x_i)\pi_n(x_i)} = \frac{2 \langle H_n, H_n \rangle}{H'_{n+1}(x_i)H_n(x_i)} = \frac{2^n n! \sqrt{\pi}}{(n+1)H_{n+1}^2(x_i)}, \quad 0 \leq i \leq n
\]  

(3.138)

3.4.3 Stable Gauss-Hermite Quadrature

Definition and Recurrence

For the Gauss-Hermite quadrature, using the original Hermite polynomials, the weights decay very rapidly as the order $n$ increases, which brings numerical instability. To stabilize the Gauss-Hermite quadrature and related numerical calculations, we introduce the modified Hermite polynomials as follows [51, 52],

\[
    \mathcal{H}_n(x) = \frac{1}{\sqrt{2^n n!}} e^{-x^2/2} H_n(x), \quad n \geq 0
\]  

(3.139)

Therefore, the following equation for the modified Hermite functions holds,

\[
    \int_{-\infty}^{\infty} \mathcal{H}_n(x)\mathcal{H}_m(x)dx = \sqrt{\pi} \delta_{nm}
\]  

(3.140)

then the three-term recurrence for the modified Hermite polynomials is,

\[
    \mathcal{H}_0(x) = e^{-x^2/2}
\]

(3.141)

\[
    \mathcal{H}_1(x) = \sqrt{2}xe^{-x^2/2}
\]

(3.142)

\[
    \vdots
\]

\[
    \mathcal{H}_{n+1}(x) = \sqrt{\frac{2}{n+1}} x\mathcal{H}_n(x) - \sqrt{\frac{n}{n+1}} \mathcal{H}_{n-1}(x), \quad n \geq 0
\]  

(3.143)
Grid Points and Weights

By letting \( f(x) = e^{-x^2}g(x) \), we can rewrite the Gauss-Hermite quadrature,

\[
\int_{-\infty}^{\infty} e^{-x^2} g(x) dx = \sum_{i=0}^{n} w_i g(x_i) \tag{3.144}
\]

as the following modified integration equation with the modified weights \( \{\omega_i\} \),

\[
\int_{-\infty}^{\infty} f(x) dx = \sum_{i=0}^{n} w_i e^{x_i^2} f(x_i) = \sum_{i=0}^{n} \omega_i f(x_i) \tag{3.145}
\]

where the grid points \( \{x_i\} \) are still the roots of \( H_{n+1}(x) \) or \( \mathcal{H}_{n+1}(x) \), and the stabilized weights \( \{\omega_i\} \) for the modified Hermite polynomials can be computed as follows,

\[
\omega_i = w_i e^{x_i^2} = \frac{2^n n! \sqrt{\pi} e^{x_i^2}}{(n+1) H_n^2(x_i)}
\]

\[
= \frac{\sqrt{\pi}}{(n+1) H_n^2(x_i)}, \quad 0 \leq i \leq n \tag{3.146}
\]

Some sample modified Hermite polynomials \( \mathcal{H}_n(x) \) are depicted in Figure 3.1.

![Figure 3.1: Modified Hermite polynomials \( \mathcal{H}_n(x) \), \( n = 0, \ldots, 3 \)](image_url)
3.4.4 Gauss-Laguerre Quadrature

Definitions
The Laguerre polynomials \( L_n(x), n \geq 0 \), named after Edmond Nicolas Laguerre (April 9, 1834 - August 14, 1886), a French mathematician, are the orthogonal polynomials with respect to the weighting function \( w(x) = e^{-x} \) on \([0, \infty)\),

\[
\langle L_n, L_m \rangle = \int_0^\infty e^{-x} L_n(x) L_m(x) dx = \delta_{nm}
\] (3.147)

Differential Equations
The Laguerre polynomials satisfy the Laguerre differential equation [50],

\[
xL_n''(x) + (1 - x)L_n'(x) + nL_n(x) = 0, \quad n \geq 0
\] (3.148)

which is a special case of the Sturm-Liouville differential equation as follows,

\[
\left[ x e^{-x} L_n'(x) \right]' + ne^{-x} L_n(x) = 0, \quad n \geq 0
\] (3.149)

Rodrigues’ Formula
Rodrigues’ formula for the Laguerre polynomials is [50],

\[
L_n(x) = \frac{e^x}{n!} \frac{d^n}{dx^n} \left( e^{-x} x^n \right)
\] (3.150)

First Few Polynomial Formulas
The first few Laguerre polynomials are,

\[
L_0(x) = 1 \\
L_1(x) = -x + 1 \\
L_2(x) = \frac{1}{2} (x^2 - 4x + 2) \\
L_3(x) = \frac{1}{6} (-x^3 + 9x^2 - 18x + 6) \\
L_4(x) = \frac{1}{24} (x^4 - 16x^3 + 72x^2 - 96x + 24) \\
L_5(x) = \frac{1}{120} (-x^5 + 25x^4 - 200x^3 + 600x^2 - 600x + 120) \\
L_6(x) = \frac{1}{720} (x^6 - 36x^5 + 450x^4 - 2400x^3 + 5400x^2 - 4320x + 720)
\] (3.151)
Polynomial Recurrence Relations

The Laguerre polynomials satisfy the following three-term recurrence relation [50],

\[
L_0(x) = 1 \tag{3.152}
\]
\[
L_1(x) = (1 - x)L_0(x) = -x + 1 \tag{3.153}
\]
\[
\vdots
\]
\[
(n + 1)L_{n+1}(x) = (2n + 1 - x)L_n(x) - nL_{n-1}(x), \quad n \geq 0 \tag{3.154}
\]

Derivative Recurrence Relations

The recurrence relations for the derivatives are [50],

\[
xL'_n(x) = nL_n(x) - nL_{n-1}(x), \quad n \geq 0 \tag{3.155}
\]
\[
L_n(x) = L'_n(x) - L'_{n+1}(x), \quad n \geq 0 \tag{3.156}
\]

We can prove (3.156) from (3.155) and the three-term recurrence as follows,

\[
xL'_{n+1}(x) = (n + 1)L_{n+1}(x) - (n + 1)L_n(x)
= (n - x)L_n(x) - nL_{n-1}(x) \tag{3.157}
\]

Subtracting two of the above equations, we have,

\[
x[L'_n(x) - L'_{n+1}(x)] = xL_n(x) \tag{3.158}
\]

which is equal for all \( x \in [0, \infty) \), therefore,

\[
L'_n(x) - L'_{n+1}(x) = L_n(x) \tag{3.159}
\]

and hence,

\[
L'_{n+1}(x) = -\sum_{k=0}^{n} L_k(x) \tag{3.160}
\]

Symmetries and Special Values

From the recurrence relations of the Laguerre polynomials and derivatives, it is easy to see the values of the polynomials and derivatives at the point of zero are as follows,

\[
L_n(0) = 1, \quad L'_n(0) = -n, \quad n \geq 0 \tag{3.161}
\]
**The Eigenvalue Method**

From the three-term recurrence relation of the Laguerre polynomials, it is also easy to see the coefficient of $x^n$ in $L_n(x)$ is $(-1)^n n!$, $n \geq 0$, that is,

$$\pi_n(x) = (-1)^n n! L_n(x), \quad n \geq 0 \tag{3.162}$$

then, the three-term recurrence can be converted to the monic recurrence as follows,

$$\begin{align*}
\pi_0(x) &= 1 \\
\pi_1(x) &= (x - 1)\pi_0(x) = x - 1 \\
\vdots \\
\pi_{n+1}(x) &= [x - (2n + 1)]\pi_n(x) - n^2\pi_{n-1}(x), \quad n \geq 0 \tag{3.165}
\end{align*}$$

The general formula of the monic three-term recurrence is as follows,

$$\pi_{n+1}(x) = (x - a_n)\pi_n(x) - b_n\pi_{n-1}(x), \quad n \geq 0 \tag{3.166}$$

hence, we have the $a_n$ and $b_n$ as follows,

$$a_n = 2n + 1, \quad n \geq 0, \quad \text{and} \quad b_n = n^2, \quad n \geq 1 \tag{3.167}$$

while the first moment is,

$$u_0 = \langle \pi_0, \pi_0 \rangle = \langle L_0, L_0 \rangle = 1 \tag{3.168}$$

Then, the grid points $\{x_i\}$ and the associated weights $\{w_i\}$ for the Gauss-Laguerre quadrature can be computed by implementing Theorem 3.3.19.

**The Analytical Formulas**

In Gauss-Laguerre quadrature, the grid points will be the roots of $L_{n+1}(x)$, $n \geq 0$. For the weights, we can solve them from (3.58) in Corollary 3.3.14, i.e., for $n \geq 0$,
\[ w_i = \frac{\langle \pi_n, \pi_n \rangle}{\pi_{n+1}(x_i) \pi_n(x_i)} \]
\[ = -\frac{1}{n + 1} \frac{\langle L_n, L_n \rangle}{L_{n+1}'(x_i)L_n(x_i)} \]
\[ = -\frac{1}{n + 1} \frac{1}{L_{n+1}'(x_i)L_n(x_i)} \]
\[ = \frac{x_i}{[(n + 1)L_n(x_i)]^2}, \quad 0 \leq i \leq n \quad (3.169) \]

### 3.4.5 Stable Gauss-Laguerre Quadrature

#### Definition and Recurrence

For the Gauss-Laguerre quadrature, similar to the original Gauss-Hermite quadrature, the weights decay very rapidly as the order \( n \) increases, which brings numerical instability. To stabilize the Gauss-Laguerre quadrature and related numerical calculations, we introduce the modified Laguerre polynomials as follows [53, 54],

\[ \mathcal{L}_n(x) = e^{-x/2}L_n(x), \quad n \geq 0 \quad (3.170) \]

Therefore, the following equation for the modified Hermite functions is,

\[ \int_0^\infty \mathcal{L}_n(x)\mathcal{L}_m(x)dx = \delta_{nm} \quad (3.171) \]

then, the three-term recurrence for the modified Laguerre polynomials is,

\[ \mathcal{L}_0(x) = e^{-x/2} \quad (3.172) \]
\[ \mathcal{L}_1(x) = (-x + 1)e^{-x/2} \quad (3.173) \]
\[ \vdots \]
\[ (n + 1)\mathcal{L}_{n+1}(x) = (2n + 1 - x)\mathcal{L}_n(x) - n\mathcal{L}_{n-1}(x), \quad n \geq 0 \quad (3.174) \]
Grid Points and Weights

By letting \( f(x) = e^{-x}g(x) \), we can rewrite the Gauss-Laguerre quadrature,

\[
\int_{0}^{\infty} e^{-x}g(x)dx = \sum_{i=0}^{n} w_i g(x_i)
\]  

(3.175)

as the following modified integration equation with the modified weights \( \{\omega_i\} \),

\[
\int_{0}^{\infty} f(x)dx = \sum_{i=0}^{n} w_i e^{x_i} f(x_i) = \sum_{i=0}^{n} \omega_i f(x_i)
\]  

(3.176)

where the grid points \( \{x_i\} \) are still the roots of \( L_{n+1}(x) \) or \( \mathcal{L}_{n+1}(x) \), and the stabilized weights \( \{\omega_i\} \) for the modified Laguerre polynomials can be computed as follows,

\[
\omega_i = w_i e^{x_i} = \frac{x_i e^{x_i}}{[(n+1)L_n(x_i)]^2}
\]

\[
= \frac{x_i}{[(n+1)\mathcal{L}_n(x_i)]^2}, \quad 0 \leq i \leq n
\]  

(3.177)

Some sample modified Laguerre polynomials \( \mathcal{L}_n(x) \) are depicted in Figure 3.2.

![Figure 3.2: Modified Laguerre polynomials \( \mathcal{L}_n(x) \), \( n = 0, \ldots, 3 \)](image-url)
3.4.6 Gauss-Legendre Quadrature

Definitions

The Legendre polynomials \( P_n(x) \), \( n \geq 0 \), named after Adrien-Marie Legendre (September 18, 1752 - January 10, 1833), a French mathematician, are the orthogonal polynomials with respect to the weighting function \( w(x) = 1 \) on \([-1, 1]\) as follows,

\[
\langle P_n, P_m \rangle = \int_{-1}^{1} P_n(x)P_m(x)dx = \frac{2}{2n + 1}\delta_{nm}
\]

Differential Equations

The Legendre polynomials satisfy the Legendre differential equation [50],

\[
(1 - x^2)P''_n(x) - 2xP'_n(x) + n(n + 1)P_n(x) = 0, \quad n \geq 0
\]

which is a special case of the Sturm-Liouville differential equation as follows,

\[
[(1 - x^2)P'_n(x)]' + n(n + 1)P_n(x) = 0, \quad n \geq 0
\]

Rodrigues’ Formula

Rodrigues’ formula for the Legendre polynomials is [50],

\[
P_n(x) = \frac{1}{2^n n!} \frac{d^n}{dx^n} (x^2 - 1)^n
\]

First Few Polynomial Formulas

The first few Legendre polynomials are,

\[
\begin{align*}
P_0(x) &= 1 \\
P_1(x) &= x \\
P_2(x) &= \frac{1}{2}(3x^2 - 1) \\
P_3(x) &= \frac{1}{2}(5x^3 - 3x) \\
P_4(x) &= \frac{1}{8}(35x^4 - 30x^2 + 3) \\
P_5(x) &= \frac{1}{8}(63x^5 - 70x^3 + 15x) \\
P_6(x) &= \frac{1}{16}(231x^6 - 315x^4 + 105x^2 - 5)
\end{align*}
\]
**Polynomial Recurrence Relations**

The Legendre polynomials satisfy the following three-term recurrence relation [50],

\[ P_0(x) = 1 \]  
\[ P_1(x) = xP_0(x) = x \]  
\[ n \geq 0 \]  

\[ (n + 1)P_{n+1}(x) = (2n + 1)xP_n(x) - nP_{n-1}(x), \quad n \geq 0 \]  

**Derivative Recurrence Relations**

The recurrence relations for the derivatives are [50],

\[ (1 - x^2)P'_n(x) = -nP_n(x) + nP_{n-1}(x), \quad n \geq 0 \]  
\[ (2n + 1)P_n(x) = P'_{n+1}(x) - P'_n(x), \quad n \geq 0 \]  
\[ (1 - x^2)P'_n(x) = -n^2 + n \frac{P_{n+1}(x) - P_{n-1}(x)}{2n + 1}, \quad n \geq 0 \]  

We can prove (3.187) and (3.188) from (3.186) and (3.185) as follows,

\[ (1 - x^2)P'_n(x) = -nP_n(x) + nP_{n-1}(x) \]
\[ = (n + 1)xP_n(x) - (n + 1)P_{n+1}(x) \]  

Therefore,

\[ (1 - x^2)P'_{n+1}(x) = -(n + 1)xP_{n+1}(x) + (n + 1)P_n(x) \]
\[ = -x^2(2n + 1)P_n(x) + nxP_{n-1}(x) + (n + 1)P_n(x) \]  

And also,

\[ (1 - x^2)P'_{n-1}(x) = nxP_{n-1}(x) - nP_n(x) \]  

Subtracting the above two equations, we have,

\[ P'_{n+1}(x) - P'_{n-1}(x) = (2n + 1)P_n(x) \]  

57
For (3.188), we can derive it as follows,

\[
(1 - x^2)P'_n(x) = -nP_n(x) + nP'_{n-1}(x)
\]

\[
= -\frac{n^2 + n}{2n + 1} P_{n+1}(x) - \frac{n^2}{2n + 1} P_{n-1}(x) + nP_{n-1}(x)
\]

\[
= -\frac{n^2 + n}{2n + 1} [P_{n+1}(x) - P_{n-1}(x)]
\]

(3.193)

**Symmetries and Special Values**

From the three-term recurrence relation of the Legendre polynomials and derivatives, it is easy to see the symmetry property of \(P_n(x)\) and \(P'_n(x)\) as follows,

\[
P_n(-x) = (-1)^n P_n(x), \quad P'_n(-x) = (-1)^{n+1} P'_n(x), \quad n \geq 0
\]

(3.194)

and the values of the Legendre polynomials at special points are as follows,

\[
P_n(-1) = (-1)^n, \quad P_{2m+1}(0) = 0, \quad P_n(1) = 1, \quad n \geq 0, \quad m \geq 0
\]

(3.195)

and for \(n \geq 0, \ m \geq 0\), the derivatives at these special points are as follows,

\[
P'_n(-1) = (-1)^{n+1} \frac{n(n+1)}{2}, \quad P'_{2m}(0) = 0, \quad P'_n(1) = \frac{n(n+1)}{2}
\]

(3.196)

**The Eigenvalue Method**

From the three-term recurrence relation of the Legendre polynomials, it is also easy to see the coefficient of \(x^n\) in \(P_n(x)\) is \(\frac{(2n)!}{2^n (n!)^2}\), \(n \geq 0\), that is,

\[
\pi_n(x) = \frac{2^n (n!)^2}{(2n)!} P_n(x), \quad n \geq 0
\]

(3.197)

then, the three-term recurrence can be converted to the monic recurrence as follows,

\[
\pi_0(x) = 1
\]

(3.198)

\[
\pi_1(x) = x \pi_0(x) = x
\]

(3.199)

\[
\vdots
\]

\[
\pi_{n+1}(x) = x \pi_n(x) - \frac{n^2}{4n^2 - 1} \pi_{n-1}(x), \quad n \geq 0
\]

(3.200)
The general formula of the monic three-term recurrence is as follows,

\[ \pi_{n+1}(x) = (x - a_n)\pi_n(x) - b_n\pi_{n-1}(x), \quad n \geq 0 \]  
(3.201)

hence, we have the \( a_n \) and \( b_n \) as follows,

\[ a_n = 0, \quad n \geq 0, \quad \text{and} \quad b_n = \frac{n^2}{4n^2 - 1}, \quad n \geq 1 \]  
(3.202)

while the first moment is,

\[ u_0 = \langle \pi_0, \pi_0 \rangle = \langle P_0, P_0 \rangle = 2 \]  
(3.203)

Then, the grid points \( \{x_i\} \) and the associated weights \( \{w_i\} \) for the Gauss-Legendre quadrature can be computed by implementing Theorem 3.3.19.

**The Analytical Formulas**

In Gauss-Legendre quadrature, the grid points will be the roots of \( P_{n+1}(x), n \geq 0 \). For the weights, we can solve them from (3.58) in Corollary 3.3.14, for \( n \geq 0 \),

\[ w_i = \frac{\langle \pi_n, \pi_n \rangle}{\pi'_{n+1}(x_i)\pi_n(x_i)} = \frac{2n + 1}{n + 1} \frac{\langle P_n, P_n \rangle}{P'_{n+1}(x_i)P_n(x_i)} = \frac{2}{n + 1} \frac{1}{P'_{n+1}(x_i)P_n(x_i)} = \frac{2}{(1 - x_i^2)[P'_{n+1}(x_i)]^2} = \frac{2(1 - x_i^2)}{[(n + 1)P_n(x_i)]^2}, \quad 0 \leq i \leq n \]  
(3.204)

Some sample Legendre polynomials \( P_n(x) \) are depicted in the following Figure 3.3.
3.5 Radau and Lobatto Quadratures

Gauss-type quadrature rules can be extended to the cases of preassigned nodes, which are required to be included in some applications. This is particularly true for imposing boundary conditions for solving PDEs numerically [55].

For the regular Gauss quadrature, all the grid points are strictly inside the interval of \((a, b)\). For Gauss-Radau quadrature, one of the two end points will be included as a grid point, i.e., either \(x_0 = a\) or \(x_n = b\). If the grid points include both of the two end points, i.e., \(x_0 = a\) and \(x_n = b\), it becomes Gauss-Lobatto quadrature.

Gauss-Radau quadrature including the end point of \(a\) has the form, as follow,

\[
\int_a^b w(x)f(x)dx = \sum_{i=0}^n w_i g(x_i) = w_0 f(a) + \sum_{i=1}^n w_i f(x_i)
\] (3.206)

or has the following formula if including the end point of \(b\) as follows,
\[
\int_a^b w(x)f(x)dx = \sum_{i=0}^n w_i g(x_i) = \sum_{i=0}^{n-1} w_i f(x_i) + w_n f(b)
\]  
(3.207)

which is exact for any polynomial of degree up to \(2n\) by craftily choosing the \(n+1\) weights and the rest of \(n\) grid points besides the one known grid point.

Likewise, Gauss-Lobatto quadrature, which includes both of the end points \(a\) and \(b\), has the integration equation as follows,

\[
\int_a^b w(x)f(x)dx = \sum_{i=0}^n w_i g(x_i) = w_0 f(a) + \sum_{i=1}^{n-1} w_i f(x_i) + w_n f(b)
\]  
(3.208)

which is exact for any polynomial of degree up to \(2n-1\) with the \(n+1\) appropriate weights and the rest of \(n-1\) associated grid points besides the two known grid points.

The \textit{Gauss-Radau-Laguerre quadrature} is using Laguerre polynomials with preassigned node of left end point 0 and the \textit{Gauss-Lobatto-Legendre quadrature} uses Legendre polynomials with preassigned nodes of both ends -1 and 1.

\subsection*{3.5.1 Proof of Gauss-Radau Quadrature}

Without loss of generality, we will work on the case of preassigned node of \(a\) only.

\textbf{Theorem 3.5.1.} No matter how we choose the rest of grid points and the associated weights, The Gauss-Radau quadrature in (3.206) can not hold exact for all the polynomials of degree more than \(2n\), i.e., the Gauss-Radau quadrature is only possible to be precise for polynomials of degree up to \(2n\).

\textit{Proof.} Let \(g(x)\) be a polynomial of degree \(2n+1\) as follows,

\[
g(x) = (x-a) \prod_{j=1}^n (x-x_j)^2
\]  
(3.209)

where \(x_1, \ldots, x_n\) are chosen grid points besides \(a\).
Then, since \( g(x) \) is a nonnegative polynomial in \([a, b]\),

\[
0 < \int_a^b w(x)g(x)dx \neq \sum_{i=0}^n w_i g(x_i) = 0 \tag{3.210}
\]
hence, (3.206) holds exactly for any polynomial of degree no more than \(2n\).

**Lemma 3.5.2.** The \((n + 1)\)th polynomial \(\rho_{n+1}(x)\), \(n \geq 0\),

\[
\rho_{n+1}(x) = \begin{vmatrix} p_{n+1}(x) & p_n(x) \\ p_{n+1}(a) & p_n(a) \end{vmatrix} = p_{n+1}(x)p_n(a) - p_{n+1}(a)p_n(x) \tag{3.211}
\]

has \(n + 1\) distinct real roots in \([a, b]\). The first root is \(x_0 = a\) and the other \(n\) roots are located in the interior of the support interval between \(a\) and \(b\), that is, \((a, b)\).

**Proof.** All the roots of \(p_n(x)\) reside inside of \((a, b)\), therefore \(p_n(a) \neq 0\) and \(\rho_{n+1}(x)\) is indeed an \((n + 1)\)th degree polynomial. \(x_0 = a\) apparently is one root of \(\rho_{n+1}(x)\).

Let \(x_1, \ldots, x_m\) be the other \(m\) distinct roots in \((a, b)\) and \(\rho_{n+1}(x)\) changes signs at those points. As \(\rho_{n+1}(x)\) is an \((n + 1)\)th degree polynomial, \(m\) only can be anywhere from zero to a maximum of \(n\), i.e., \(0 \leq m \leq n\). However, if \(m < n\), by orthogonality,

\[
\int_a^b w(x)\rho_{n+1}(x)(x - x_1)(x - x_2) \cdots (x - x_m)dx = 0 \tag{3.212}
\]

which is impossible to be zero from Corollary 3.3.1 since \(x > a\) in \((a, b)\) and so the combined polynomial \(\rho_{n+1}(x) \prod_{i=0}^m(x - x_i)\) in the integrand never changes signs in \((a, b)\). Therefore, \(m\) must be equal to \(n\), and consequently \(\rho_{n+1}(x)\) has the rest of \(n\) distinct real roots inside of \((a, b)\) besides the first root, i.e., \(x_0 = a\).

**Theorem 3.5.3.** Let the grid points, \(x_0, x_1, \ldots, x_n\), be the distinct real roots in \([a, b]\) of the \((n + 1)\)th polynomial \(\rho_{n+1}(x)\) as follows,

\[
\rho_{n+1}(x) = \begin{vmatrix} p_{n+1}(x) & p_n(x) \\ p_{n+1}(a) & p_n(a) \end{vmatrix} = p_{n+1}(x)p_n(a) - p_{n+1}(a)p_n(x) \tag{3.213}
\]
and let \( \ell_i(x) \) the Lagrange interpolation polynomial based on those grid points,

\[
\ell_i(x) = \prod_{j=0, j \neq i}^{n} \frac{x - x_j}{x_i - x_j} = \frac{\rho_{n+1}(x)}{(x - x_i)} \rho'_{n+1}(x), \quad i = 0, 1, \cdots, n
\]  

(3.214)

and the weight \( w_i \) is determined as follows,

\[
w_i = \int_a^b w(x)\ell_i(x)dx
\]

(3.215)

then, the Gauss-Radau quadrature holds exactly as follows,

\[
\int_a^b w(x)f(x)dx = w_0 f(a) + \sum_{i=1}^{n} w_i f(x_i)
\]

(3.216)

for any polynomial \( f(x) \) of degree up to \( 2n \) and the weights \( \{w_i\} \) are positive.

**Proof.** Suppose \( f(x) \) is a polynomial of degree \( 2n \) or less, which is divided by the polynomial \( \rho_{n+1}(x) \) as follows,

\[
f(x) = \rho_{n+1}(x)q(x) + r(x)
\]

(3.217)

where \( q(x) \) is a polynomial of degree \( n - 1 \) or less and \( r(x) \) is a polynomial of degree \( n \) or less, therefore, we have,

\[
f(x_i) = \rho_{n+1}(x_i)q(x_i) + r(x_i) = r(x_i), \quad i = 0, 1, \cdots, n
\]

(3.218)

and \( r(x) \) can be exactly interpolated by the Lagrange polynomials,

\[
r(x) = \sum_{i=0}^{n} r(x_i)\ell_i(x) = \sum_{i=0}^{n} f(x_i)\ell_i(x)
\]

(3.219)

then, because of orthogonality and the above properties,

\[
\int_a^b w(x)f(x)dx = \int_a^b w(x)\rho_{n+1}(x)q(x)dx + \int_a^b w(x)r(x)dx
\]
\[ = 0 + \int_a^b w(x) \sum_{i=0}^{n} f(x_i) \ell_i(x) \, dx \]

\[ = \sum_{i=0}^{n} f(x_i) \int_a^b w(x) \ell_i(x) \, dx \]

\[ = \sum_{i=0}^{n} w_i f(x_i) \quad (3.220) \]

To prove the weight \( w_i \) is positive, let \( h_i(x) \) be as follows,

\[
 h_0(x) = \prod_{\substack{j=1 \atop j \neq i}}^{n} (x - x_j)^2, \quad h_i(x) = (x - a) \prod_{\substack{j=1 \atop j \neq i}}^{n} (x - x_j)^2, \quad i = 1, \ldots, n \quad (3.221)
\]

then, since \( h_i(x) \) is a nonnegative polynomial of degree less than \( 2n \),

\[
 0 < \int_a^b w(x) h_i(x) \, dx = \sum_{k=0}^{n} w_k h_i(x_k) = w_i h_i(x_i) \quad (3.222)
\]

the roots \( \{x_i\} \) are real and distinct, so \( h_i(x_i) > 0 \), therefore \( w_i \) is indeed positive. \( \square \)

**Theorem 3.5.4** (the eigenvalue method). If \( a_n, \ n \geq 0 \) and \( b_n, \ n \geq 1 \) are known in the monic three-term recurrence relation for the orthogonal polynomials, then the weights \( w_i \) and the grid points \( x_i, \ 0 \leq i \leq n \), for the Gauss-Radau quadrature can be obtained from the eigenvalue decomposition of the modified symmetric tridiagonal Jacobi matrix \( J_{n+1}^* \). The eigenvalue decomposition of \( J_{n+1}^* \) is as follows,

\[
 J_{n+1}^* = \begin{bmatrix}
  a_0 & \sqrt{b_1} \\
 \sqrt{b_1} & a_1 & \sqrt{b_2} \\
 & \ddots & \ddots & \ddots \\
 & & \sqrt{b_{a-1}} & a_{a-1} & \sqrt{b_n} \\
 & & & \sqrt{b_n} & a_n^* 
\end{bmatrix} = VAV^T \quad (3.223)
\]

where the modified \( a_n^* \) is

\[ a_n^* = a_n + \frac{\pi_{n+1}(a)}{\pi_n(a)} = a - b_n \frac{\pi_{n-1}(a)}{\pi_n(a)}, \quad n \geq 0 \quad (3.224) \]
and $\Lambda$ is the diagonal matrix of eigenvalues $\{\lambda_i, 0 \leq i \leq n\}$, that is,

$$
\Lambda = \begin{bmatrix}
\lambda_0 \\
\lambda_1 \\
\vdots \\
\lambda_n
\end{bmatrix}, \quad \text{where } \lambda_0 < \lambda_1 < \cdots < \lambda_n
$$

(3.225)

and $V$ is the orthogonal matrix of eigenvectors, i.e., $V^TV = I$. Each column vector $v_i$ of $V$ is the normalized eigenvector respect to the corresponding eigenvalue $\lambda_i$, i.e.,

$$
V = \begin{bmatrix} v_0 & v_1 & \cdots & v_n \end{bmatrix}, \quad \text{where } v_i^T v_j = \delta_{ij}, \quad 0 \leq i, j \leq n
$$

(3.226)

Then, $x_i = \lambda_i$, and $w_i = u_0[v_i]_0^2$, where $u_0$ is the first moment of the orthogonal polynomial and $[v_i]_0$ is the first element of the orthonormal eigenvector $v_i$.

**Proof.** From the definition of the orthonormal polynomial $\varphi_n(x)$, we have,

$$
\varphi_n(x) = \frac{\pi_n(x)}{\parallel\pi_n\parallel}, \quad n \geq 0, \quad \text{and } \varphi_{-1}(x) = 0
$$

(3.227)

and from the definition of three-term recurrence, we have,

$$
b_n = \frac{\langle \pi_n, \pi_n \rangle}{\langle \pi_{n-1}, \pi_{n-1} \rangle} = \frac{\parallel\pi_n\parallel^2}{\parallel\pi_{n-1}\parallel^2}, \quad n \geq 1
$$

(3.228)

therefore, with the monic three-term recurrence, the modified $a_n^*$ is as follows,

$$
a_n^* = a_n + \frac{\pi_{n+1}(a)}{\pi_n(a)} = a_n \frac{\pi_n(a) + \pi_{n+1}(a)}{\pi_n(a)}
$$

$$
= a_n \frac{\pi_n(a) - b_n \pi_{n-1}(a)}{\pi_n(a)} = a_n \frac{\pi_{n-1}(a)}{\pi_n(a)}
$$

$$
= a_n + \frac{\parallel\pi_{n+1}\parallel}{\parallel\pi_n\parallel} \frac{\varphi_{n+1}(a)}{\varphi_n(a)}
$$

$$
= a_n + \sqrt{b_{n+1}} \frac{\varphi_{n+1}(a)}{\varphi_n(a)}, \quad n \geq 0
$$

(3.229)
Also from the monic three-term recurrence, we have the following equation,

\[ x \varphi_k(x) = \sqrt{b_{k+1}} \varphi_{k+1}(x) + a_k \varphi_k(x) + \sqrt{b_k} \varphi_{k-1}(x), \quad k \geq 0 \]  
(3.230)

and when \( k = n \), from the definition of \( a_n^* \), we have,

\begin{align*}
x \varphi_n(x) &= \sqrt{b_{n+1}} \varphi_{n+1}(x) + a_n \varphi_n(x) + \sqrt{b_n} \varphi_{n-1}(x) \\
&= \sqrt{b_{n+1}} \varphi_{n+1}(x) + (a_n - a_n^*) \varphi_n(x) + a_n^* \varphi_n(x) + \sqrt{b_n} \varphi_{n-1}(x) \\
&= \sqrt{b_{n+1}} \varphi_{n+1}(x) - \sqrt{b_{n+1}} \frac{\varphi_{n+1}(a)}{\varphi_n(a)} \varphi_n(x) + a_n^* \varphi_n(x) + \sqrt{b_n} \varphi_{n-1}(x) \\
&= \sqrt{b_{n+1}} \varphi_n^*(x) + a_n^* \varphi_n(x) + \sqrt{b_n} \varphi_{n-1}(x) \\
&= \sqrt{b_{n+1}} \varphi_n^*(x) + a_n^* \varphi_n(x) + \sqrt{b_n} \varphi_{n-1}(x) \\
&= \sqrt{b_{n+1}} \varphi_n^*(x) + a_n^* \varphi_n(x) + \sqrt{b_n} \varphi_{n-1}(x) \\
\end{align*}

where,

\[ \varphi_n^*(x) = \frac{1}{\varphi_n(a)} [\varphi_{n+1}(x) \varphi_n(a) - \varphi_n(a) \varphi_{n+1}(x)] \]

\[ = \frac{1}{p_n(a) p_{n+1}(a)} [p_{n+1}(x) p_n(a) - p_n(a) p_{n+1}(x)] \]

\[ = \frac{1}{p_n(a) p_{n+1}(a)} \rho_{n+1}(x) \]  
(3.232)

then we have the following system of \((n + 1)\) equations,

\[ x \begin{bmatrix}
\varphi_0(x) \\
\varphi_1(x) \\
\vdots \\
\varphi_{n-1}(x) \\
\varphi_n(x)
\end{bmatrix} = \begin{bmatrix}
a_0 & \sqrt{b_1} \\
\sqrt{b_1} & a_1 & \sqrt{b_2} \\
& \ddots & \ddots & \ddots \\
& & \ddots & a_{n-1} & \sqrt{b_n} \\
& & & \sqrt{b_n} & a_n^*
\end{bmatrix} \begin{bmatrix}
\varphi_0(x) \\
\varphi_1(x) \\
\vdots \\
\varphi_{n-1}(x) \\
\varphi_n(x)
\end{bmatrix} + \begin{bmatrix}
0 \\
0 \\
\vdots \\
0 \\
\sqrt{b_{n+1}} \varphi_n^*(x)
\end{bmatrix} \]

that is,

\[ x \Psi(x) = J_{n+1}^* \Psi(x) + \Upsilon^*(x) \]  
(3.233)
where,

\[ \Psi(x) = \begin{bmatrix} \varphi_0(x) \\ \varphi_1(x) \\ \vdots \\ \varphi_n(x) \end{bmatrix}, \quad \text{and} \quad \Phi^*(x) = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ \sqrt{b_{n+1}} \varphi^*_{n+1}(x) \end{bmatrix} \]

let \( x = x_i \), the root of \( \rho_{n+1}(x) \), which is also the root of \( \varphi^*_{n+1}(x) \), then,

\[ x_i \Psi(x_i) = J^*_{n+1} \Psi(x_i) \] (3.234)

therefore, \( x_i \) is an eigenvalue of \( J^*_{n+1} \) with eigenvector \( \Psi(x_i) \). Hence, eigenvalues of \( J^*_{n+1} \) are indeed roots of \( \rho_{n+1}(x) \), i.e., the grid points of the Gauss-Radau quadrature.

Since Gauss-Radau quadrature is exactly for polynomials of degree up to \( 2n \), the dual orthogonality of Lemma 3.3.17 still holds with new grid points and weights,

\[ \Psi(x_i)^T \Psi(x_i) = \sum_{k=0}^{n} \frac{p_k(x_i)p_k(x_i)}{\|p_k\|^2} = \sum_{k=0}^{n} \frac{p_k(x_i)p_k(x_i)}{\langle p_k, p_k \rangle} = \frac{1}{w_i} \] (3.235)

therefore, the orthonormal eigenvector \( v_1 = \pm \sqrt{w_i} \Psi(x_i) \), then

\[ w_i = \frac{[v_1]_0^2}{\varphi^2(x_i)} = \frac{\|\pi_0\|^2}{\pi_0^2} [v_1]_0^2 = \langle \pi_0, \pi_0 \rangle [v_1]_0^2 = u_0[v_1]_0^2 \] (3.236)

where \([v_1]_0\) is the first element of the orthonormal vector \( v_1 \).

\[ \square \]

3.5.2 Gauss-Radau-Laguerre Quadrature

Gauss-Radau-Laguerre quadrature includes the left-end point of zero as one grid point, which is useful for solving problems with semi-infinite intervals and enforcing boundary conditions at zero. It formulates as follows,

\[ \int_0^\infty e^{-x} f(x) dx = \sum_{i=0}^{n} w_i g(x_i) = w_0 f(0) + \sum_{i=1}^{n} w_i f(x_i) \] (3.237)
The grid points are the roots of *Gauss-Radau-Laguerre polynomial* \( \rho_{n+1}(x) \),

\[
\rho_{n+1}(x) = L_{n+1}(x)L_n(0) - L_{n+1}(0)L_n(x)
= L_{n+1}(x) - L_n(x)
= \frac{1}{n+1} x L'_{n+1}(x)
\]

and \( x L'_{n+1}(x) \) (3.238)

and the Lagrange polynomials \( \ell_i(x) \) based on those *interpolation points* are,

\[
\ell_i(x) = \frac{\rho_{n+1}(x)}{(x-x_i)\rho'_{n+1}(x_i)}
= \frac{L_{n+1}(x) - L_n(x)}{(x-x_i)[L'_{n+1}(x_i) - L'_{n}(x_i)]}
= \frac{-L_{n+1}(x) - L_n(x)}{(x-x_i)L_n(x_i)}, \quad i = 0, 1, \ldots, n
\]

and the associated weights on the grid points are derived as follows,

\[
w_i = \int_a^b w(x)\ell_i(x)dx
= \int_a^b w(x)\left(\frac{L_{n+1}(x) - L_n(x)}{(x-x_i)L_n(x_i)}\right)dx
= \frac{-1}{L^2_n(x_i)} \int_a^b w(x)\frac{L_{n+1}(x)L_n(x_i) - L_n(x)L_{n+1}(x_i)}{x-x_i}dx
= \frac{-1}{L^2_n(x_i)} \int_a^b w(x)\frac{L_{n+1}(x)L_n(x_i) - L_n(x)L_{n+1}(x_i)}{x-x_i}dx
= \frac{1}{L^2_n(x_i)n!(n+1)!} \int_a^b w(x)\frac{\pi_{n+1}(x)\pi_n(x_i) - \pi_n(x)\pi_{n+1}(x_i)}{x-x_i}dx
= \frac{\langle \pi_n, \pi_n \rangle}{L^2_n(x_i)n!(n+1)!} = \frac{n!n!\langle L_n, L_n \rangle}{L^2_n(x_i)n!(n+1)!}
= \frac{1}{(n+1)L^2_n(x_i)}, \quad i = 0, 1, \ldots, n
\]

(3.242)
The modified \(a_n^*\) in the eigenvalue method solving the grid points and weights is,

\[
a_n^* = a_n + \frac{\pi_{n+1}(0)}{\pi_n(0)} = 2n + 1 - (n + 1) = n, \quad n \geq 0
\]  

(3.243)

with \(a_n, b_n\) and the first moment \(u_0\) as follows,

\[
a_n = 2n + 1, \quad n \geq 0, \quad b_n = n^2, \quad n \geq 1, \quad u_0 = 1
\]  

(3.244)

### 3.5.3 Stable Gauss-Radau-Laguerre Quadrature

For the Gauss-Radau-Laguerre quadrature, similar to the Gauss-Laguerre quadrature, the weights decay very rapidly as the order \(n\) increases. To stabilized numerical calculations, we have the modified Laguerre polynomials as introduced before,

\[
\mathcal{L}_n(x) = e^{-x/2}L_n(x), \quad n \geq 0
\]  

(3.245)

By letting \(g(x) = e^{-x}f(x)\), we can rewrite the Gauss-Radau-Laguerre quadrature,

\[
\int_0^\infty e^{-x}f(x)dx = \sum_{i=0}^{n} w_i f(x_i) = w_0 f(0) + \sum_{i=1}^{n} w_i f(x_i)
\]  

(3.246)

as the following modified integration equation with the modified weights \(\{\omega_i\}\),

\[
\int_0^\infty g(x)dx = \sum_{i=0}^{n} w_i e^{x_i}g(x_i) = \sum_{i=0}^{n} \omega_i g(x_i) = \omega_0 g(0) + \sum_{i=1}^{n} \omega_i g(x_i)
\]  

(3.247)

where the grid points \(\{x_i\}\) are still the roots of \(\rho_{n+1}(x)\), and the stabilized weights \(\{\omega_i\}\) can be computed as follows,

\[
\omega_i = w_i e^{x_i} = \frac{e^{x_i}}{(n+1)\mathcal{L}^2_n(x_i)} = \frac{1}{(n+1)\mathcal{L}^2_n(x_i)}, \quad 0 \leq i \leq n
\]  

(3.248)
3.5.4 Proof of Gauss-Lobatto Quadrature

**Theorem 3.5.5.** No matter how we choose the rest of grid points and the associated weights, the Gauss-Lobatto quadrature in (3.208) can not hold exact for all the polynomials of degree more than $2n - 1$, i.e., the Gauss-Lobatto quadrature is only possible to be precise for polynomials of degree up to $2n - 1$.

**Proof.** Let $g(x)$ be a polynomial of degree $2n$ as follows,

$$g(x) = (x - a)(b - x) \prod_{j=1}^{n-1} (x - x_j)^2 \quad (3.249)$$

where $x_1, ..., x_{n-1}$ are chosen grid points besides $a$ and $b$.

Then, since $g(x)$ is a nonnegative polynomial in $[a, b]$,

$$0 < \int_a^b w(x)g(x)dx \neq \sum_{i=0}^n w_i g(x_i) = 0 \quad (3.250)$$

hence, (3.208) holds exactly for any polynomial of degree no more than $2n - 1$. \qed

**Lemma 3.5.6.** The $(n + 1)^{th}$ polynomial $\rho_{n+1}(x)$, $n \geq 1$,

$$\rho_{n+1}(x) = \begin{vmatrix} p_{n+1}(x) & p_n(x) & p_{n-1}(x) \\ p_{n+1}(a) & p_n(a) & p_{n-1}(a) \\ p_{n+1}(b) & p_n(b) & p_{n-1}(b) \end{vmatrix} \quad (3.251)$$

has $n + 1$ distinct real roots in $[a, b]$. The first root is $x_0 = a$, the last root is $x_n = b$ and the other $n - 1$ roots are located in the interior of the support interval, $(a, b)$.

**Proof.** From Theorem 3.3.4,

$$\begin{vmatrix} p_n(a) & p_{n-1}(a) \\ p_n(b) & p_{n-1}(b) \end{vmatrix} > 0 \quad (3.252)$$
therefore, $\rho_{n+1}(x)$ is indeed an $(n + 1)^{th}$ degree polynomial. Moreover, $x_0 = a$ and $x_n = b$ are two roots of $\rho_{n+1}(x)$ apparently.

Let $x_1, \ldots, x_{m-1}$ be the other $m - 1$ distinct roots in $(a, b)$ and $\rho_{n+1}(x)$ changes signs at those points. Since $\rho_{n+1}(x)$ is an $(n + 1)^{th}$ degree polynomial, $m$ only can be anywhere from zero to a maximum of $n$, i.e., $0 \leq m \leq n$. However, if $m < n$, by orthogonality of the orthogonal polynomials,

$$
\int_a^b w(x)\rho_{n+1}(x)(x - x_1)(x - x_2) \cdots (x - x_{m-1})dx = 0 \quad (3.253)
$$

which is impossible to be zero from Corollary 3.3.1 since $a < x < b$ in $(a, b)$ and so the combined polynomial $\rho_{n+1}(x) \prod_{i=0}^{m-1} (x - x_i)$ in the integrand never changes signs in $(a, b)$. Therefore, $m$ must be equal to $n$, and consequently $\rho_{n+1}(x)$ has the rest of $n - 1$ distinct real roots inside of $(a, b)$ besides the two roots at the end points. \hfill \Box

**Theorem 3.5.7.** Let the grid points, $x_0, x_1, \ldots, x_n$, be the distinct real roots in $[a, b]$ of the $(n + 1)^{th}$ polynomial $\rho_{n+1}(x)$ as follows,

$$
\rho_{n+1}(x) = \begin{vmatrix}
  p_{n+1}(x) & p_n(x) & p_{n-1}(x) \\
  p_{n+1}(a) & p_n(a) & p_{n-1}(a) \\
  p_{n+1}(b) & p_n(b) & p_{n-1}(b)
\end{vmatrix} \quad (3.254)
$$

and let $\ell_i(x)$ the Lagrange interpolation polynomial based on those grid points,

$$
\ell_i(x) = \prod_{\begin{subarray}{c}
j = 0 \\
j \neq i
\end{subarray}}^{n} \frac{x - x_j}{x_i - x_j}, \quad i = 0, 1, \ldots, n \quad (3.255)
$$

and the weight $w_i$ is determined as follows,

$$
w_i = \int_a^b w(x)\ell_i(x)dx \quad (3.256)
$$

then, the Gauss-Lobatto quadrature holds exactly as follows,

$$
\int_a^b w(x)f(x)dx = w_0f(a) + \sum_{i=1}^{n-1} w_i f(x_i) + w_n f(b) \quad (3.257)
$$
for any polynomial \( f(x) \) of degree up to \( 2n - 1 \) and the weights \( \{w_i\} \) are positive.

**Proof.** Suppose \( f(x) \) is a polynomial of degree \( 2n - 1 \) or less, which is divided by the polynomial \( \rho_{n+1}(x) \) as follows,

\[
f(x) = \rho_{n+1}(x)q(x) + r(x)
\]

where \( q(x) \) is a polynomial of degree \( n - 2 \) or less and \( r(x) \) is a polynomial of degree \( n \) or less, therefore, we have,

\[
f(x_i) = \rho_{n+1}(x_i)q(x_i) + r(x_i) = r(x_i), \quad i = 0, 1, \ldots, n
\]

and \( r(x) \) can be exactly interpolated by the Lagrange polynomials,

\[
r(x) = \sum_{i=0}^{n} r(x_i)\ell_i(x) = \sum_{i=0}^{n} f(x_i)\ell_i(x)
\]

then, because of orthogonality and the above properties,

\[
\int_{a}^{b} w(x)f(x)dx = \int_{a}^{b} w(x)\rho_{n+1}(x)q(x)dx + \int_{a}^{b} w(x)r(x)dx
\]

\[
= 0 + \int_{a}^{b} w(x)r(x)dx
\]

\[
= \int_{a}^{b} w(x) \sum_{i=0}^{n} f(x_i)\ell_i(x)dx
\]

\[
= \sum_{i=0}^{n} f(x_i) \int_{a}^{b} w(x)\ell_i(x)dx
\]

\[
= \sum_{i=0}^{n} w_i f(x_i)
\]

To prove the weight \( w_i \) is positive, let \( h_i(x) \) be as follows,
\[ h_0(x) = (b - x) \prod_{j=1}^{n-1} (x - x_j)^2 \]

\[ h_i(x) = (x - a)(b - x) \prod_{j=1 \atop j \neq i}^{n-1} (x - x_j)^2, \quad i = 1, \ldots, n - 1 \]  

(3.262)

\[ h_n(x) = (x - a) \prod_{j=1}^{n-1} (x - x_j)^2 \]

then, since \( h_i(x) \) is a nonnegative polynomial of degree no more than \( 2n - 1 \),

\[ 0 < \int_a^b w(x) h_i(x) \, dx = \sum_{k=0}^n w_k h_i(x_k) = w_i h_i(x_i) \]  

(3.263)

the roots \( \{x_i\} \) are real and distinct, so \( h_i(x_i) > 0 \), therefore \( w_i \) is indeed positive.  \( \square \)

**Theorem 3.5.8** (the eigenvalue method). If \( a_n, \ n \geq 0 \) and \( b_n, \ n \geq 1 \) are known in the monic three-term recurrence relation for the orthogonal polynomials, then the weights \( w_i \) and the grid points \( x_i, \ 0 \leq i \leq n \), for the Gauss-Lobatto quadrature can be obtained from the eigenvalue decomposition of the modified symmetric tridiagonal Jacobi matrix \( J_{n+1}^* \). The eigenvalue decomposition of \( J_{n+1}^* \) is as follows,

\[
J_{n+1}^* = \begin{bmatrix}
    a_0 & \sqrt{b_1} & & \\
    \sqrt{b_1} & a_1 & \sqrt{b_2} & \\
    & \sqrt{b_2} & \ddots & \ddots \\
    & & \ddots & a_{n-1} & \sqrt{b_n} \\
    & & & \sqrt{b_n} & a_n
\end{bmatrix} = V \Lambda V^T
\]

(3.264)

where the modified \( a_n^* \) and \( b_n^* \) are the solution of the following linear equation,

\[
\begin{bmatrix}
    \pi_n(a) & \pi_{n-1}(a) \\
    \pi_n(b) & \pi_{n-1}(b)
\end{bmatrix}
\begin{bmatrix}
    a_n^* \\
    b_n^*
\end{bmatrix} = \begin{bmatrix}
    \pi_n(a) & \pi_{n-1}(a) \\
    \pi_n(b) & \pi_{n-1}(b)
\end{bmatrix}
\begin{bmatrix}
    a_n \\
    b_n
\end{bmatrix} + \begin{bmatrix}
    \pi_{n+1}(a) \\
    \pi_{n+1}(b)
\end{bmatrix} = \begin{bmatrix}
    a \pi_n(a) \\
    b \pi_n(b)
\end{bmatrix}
\]

(3.265)
and \( \Lambda \) is the diagonal matrix of eigenvalues \( \{ \lambda_i, \ 0 \leq i \leq n \} \), that is,

\[
\Lambda = \begin{bmatrix}
\lambda_0 \\
\lambda_1 \\
\vdots \\
\lambda_n
\end{bmatrix}, \quad \text{where} \quad \lambda_0 < \lambda_1 < \cdots < \lambda_n \tag{3.266}
\]

and \( V \) is the orthogonal matrix of eigenvectors, i.e., \( V^T V = I \). Each column vector \( v_i \) of \( V \) is the normalized eigenvector respect to the corresponding eigenvalue \( \lambda_i \), i.e.,

\[
V = \begin{bmatrix}
v_0 \\
v_1 \\
\vdots \\
v_n
\end{bmatrix}, \quad \text{where} \quad v_i^T v_j = \delta_{ij}, \quad 0 \leq i, j \leq n \tag{3.267}
\]

Then, \( x_i = \lambda_i \), and \( w_i = u_0 [v_i]_0^2 \), where \( u_0 \) is the first moment of the orthogonal polynomial and \( [v_i]_0 \) is the first element of the orthonormal eigenvector \( v_i \).

**Proof.** From the three-term recurrence of the monic orthogonal polynomials,

\[
\begin{bmatrix}
\pi_n(a) & \pi_{n-1}(a) \\
\pi_n(b) & \pi_{n-1}(b)
\end{bmatrix}
\begin{bmatrix}
a_n \\
b_n
\end{bmatrix} + \begin{bmatrix}
\pi_{n+1}(a) \\
\pi_{n+1}(b)
\end{bmatrix} = \begin{bmatrix}
a \pi_n(a) \\
b \pi_n(b)
\end{bmatrix}, \quad n \geq 1 \tag{3.268}
\]

and also the modified \( a_n^* \) and \( b_n^* \) satisfy the following system equation,

\[
\begin{bmatrix}
\pi_n(a) & \pi_{n-1}(a) \\
\pi_n(b) & \pi_{n-1}(b)
\end{bmatrix}
\begin{bmatrix}
a_n - a_n^* \\
b_n - b_n^*
\end{bmatrix} + \begin{bmatrix}
\pi_{n+1}(a) \\
\pi_{n+1}(b)
\end{bmatrix} = \begin{bmatrix}
0 \\
0
\end{bmatrix}, \quad n \geq 1 \tag{3.269}
\]

From the definition of the orthonormal polynomial \( \varphi_n(x) \), we have,

\[
\varphi_n(x) = \frac{\pi_n(x)}{\| \pi_n \|}, \quad n \geq 0, \quad \text{and} \quad \varphi_{-1}(x) = 0 \tag{3.270}
\]

and from the definition of three-term recurrence, we have,

\[
b_n = \frac{\langle \pi_n, \pi_n \rangle}{\langle \pi_{n-1}, \pi_{n-1} \rangle} = \frac{\| \pi_n \|^2}{\| \pi_{n-1} \|^2}, \quad n \geq 1 \tag{3.271}
\]

74
along with the three-term recurrence, we have the following equation,

\[ x\varphi_k(x) = \sqrt{b_{k+1}}\varphi_{k+1}(x) + a_k\varphi_k(x) + \sqrt{b_k}\varphi_{k-1}(x), \quad k \geq 0 \quad (3.272) \]

when \( k = n - 1 \), we have,

\[
x\varphi_{n-1}(x) = \sqrt{b_n}\varphi_n(x) + a_{n-1}\varphi_{n-1}(x) + \sqrt{b_{n-1}}\varphi_{n-2}(x)
\]

\[
= \sqrt{b_n}\frac{\sqrt{b_n}}{\sqrt{b_n^*}}\varphi_n(x) + a_{n-1}\varphi_{n-1}(x) + \sqrt{b_{n-1}}\varphi_{n-2}(x)
\]

\[
= \sqrt{b_n^*}\varphi_n^*(x) + a_{n-1}\varphi_{n-1}(x) + \sqrt{b_{n-1}}\varphi_{n-2}(x) \quad (3.273)
\]

where \( \varphi_n^*(x) = \frac{\sqrt{b_n}}{\sqrt{b_n^*}}\varphi_n(x) \), and when \( k = n \),

\[
x\varphi_n(x) = \sqrt{b_{n+1}}\varphi_{n+1}(x) + a_n\varphi_n(x) + \sqrt{b_n}\varphi_{n-1}(x) \quad (3.274)
\]

from the definition of \( \varphi_n^*(x) \), \( a_n^* \) and \( b_n^* \), we have,

\[
x\varphi_n^*(x) = \frac{\sqrt{b_n}}{\sqrt{b_n^*}}x\varphi_n(x) = \frac{\sqrt{b_n}}{\sqrt{b_n^*}} \left[ \sqrt{b_{n+1}}\varphi_{n+1}(x) + a_n\varphi_n(x) + \sqrt{b_n}\varphi_{n-1}(x) \right]
\]

\[
= \sqrt{b_n^*}b_{n+1}\varphi_{n+1}(x) + \sqrt{b_n^*}a_n\varphi_n(x) + \sqrt{b_n^*}b_n\varphi_{n-1}(x)
\]

\[
= \sqrt{b_n^*}b_{n+1}\varphi_{n+1}(x) + (a_n - a_n^*)\varphi_n^*(x) + a_n^*\varphi_n^*(x)
\]

\[
+ \left( \frac{b_n}{\sqrt{b_n^*}} - \sqrt{b_n^*} \right) \varphi_{n-1}(x) + \sqrt{b_n^*}\varphi_{n-1}(x)
\]

\[
= \sqrt{b_n^*}b_{n+1}\varphi_{n+1}(x) + a_n^*\varphi_n^*(x) + \sqrt{b_n^*}\varphi_{n-1}(x) \quad (3.275)
\]

where,
\[
\sqrt{b_{n+1}^*} \varphi_{n+1}^*(x) = \frac{\sqrt{b_n b_{n+1}}}{\sqrt{b_n^*}} \varphi_{n+1}(x) + (a_n - a_n^*) \varphi_n^*(x) + \left( \frac{b_n}{\sqrt{b_n^*}} - \sqrt{b_n^*} \right) \varphi_{n-1}(x)
\]

\[
= \frac{\sqrt{b_n b_{n+1}}}{\sqrt{b_n^*}} \varphi_{n+1}(x) + (a_n - a_n^*) \frac{\sqrt{b_n}}{\sqrt{b_n^*}} \varphi_n(x) + \left( \frac{b_n}{\sqrt{b_n^*}} - \sqrt{b_n^*} \right) \varphi_{n-1}(x)
\]

\[
= \frac{1}{\sqrt{b_n^*}} \left[ \frac{\pi_{n+1}(x)}{\parallel \pi_n \parallel} + (a_n - a_n^*) \frac{\pi_n(x)}{\parallel \pi_n - 1 \parallel} + (b_n - b_n^*) \frac{\pi_{n-1}(x)}{\parallel \pi_n - 1 \parallel} \right]
\]

\[
= \frac{\pi_{n+1}(x) + (a_n - a_n^*) \pi_n(x) + (b_n - b_n^*) \pi_{n-1}(x)}{\sqrt{b_n^*} \parallel \pi_{n-1} \parallel}
\]  

(3.276)

and,

\[
\sqrt{b_{n+1}^*} = \frac{\sqrt{b_n}}{\sqrt{b_n^*}} \sqrt{b_{n+1}}
\]  

(3.277)

\[
\varphi_{n+1}^*(x) = \frac{\pi_{n+1}(x) + (a_n - a_n^*) \pi_n(x) + (b_n - b_n^*) \pi_{n-1}(x)}{\parallel \pi_{n+1} \parallel}
\]  

(3.278)

then we have the following system of \((n + 1)\) equations,

\[
\begin{bmatrix}
\varphi_0(x) \\
\varphi_1(x) \\
\vdots \\
\varphi_{n-1}(x) \\
\varphi_n^*(x)
\end{bmatrix} =
\begin{bmatrix}
a_0 & \sqrt{b_1} \\
\sqrt{b_1} & a_1 & \sqrt{b_2} \\
\sqrt{b_2} & \ddots & \ddots \\
\vdots & \ddots & a_{n-1} & \sqrt{b_n^*} \\
\sqrt{b_n^*} & a_n^* & \sqrt{b_n} & a_n^*
\end{bmatrix}
\begin{bmatrix}
\varphi_0(x) \\
\varphi_1(x) \\
\vdots \\
\varphi_{n-1}(x) \\
\varphi_n^*(x)
\end{bmatrix} +
\begin{bmatrix}
0 \\
0 \\
\vdots \\
\vdots \\
\sqrt{b_{n+1}^*} \varphi_{n+1}^*(x)
\end{bmatrix}
\]

that is,

\[
x \Psi^*(x) = J_{n+1}^* \Psi^*(x) + \Upsilon^*(x)
\]  

(3.279)

where,

\[
\Psi^*(x) =
\begin{bmatrix}
\varphi_0(x) \\
\varphi_1(x) \\
\vdots \\
\varphi_{n-1}(x) \\
\varphi_n^*(x)
\end{bmatrix}, \quad \text{and} \quad \Upsilon^*(x) =
\begin{bmatrix}
0 \\
0 \\
\vdots \\
0 \\
\sqrt{b_{n+1}^*} \varphi_{n+1}^*(x)
\end{bmatrix}
\]

76
Let \( x = x_i, 0 \leq i \leq n \), the root of \( \varphi_{n+1}^*(x) \), which satisfies,

\[
\pi_{n+1}(x) + (a_n - a_n^*)\pi_n(x) + (b_n - b_n^*)\pi_{n-1}(x) = 0 \quad (3.280)
\]

alternatively, combing with (3.269), we have the system equation,

\[
\begin{bmatrix}
\pi_{n+1}(x) & \pi_n(x) & \pi_{n-1}(x) \\
\pi_{n+1}(a) & \pi_n(a) & \pi_{n-1}(a) \\
\pi_{n+1}(b) & \pi_n(b) & \pi_{n-1}(b)
\end{bmatrix}
\begin{bmatrix}
1 \\
a_n - a_n^* \\
b_n - b_n^*
\end{bmatrix}
= 0 \quad (3.281)
\]

therefore, the \( n + 1 \) roots of \( \varphi_{n+1}^*(x) \) have the following determinant equal to 0,

\[
\begin{vmatrix}
\pi_{n+1}(x) & \pi_n(x) & \pi_{n-1}(x) \\
\pi_{n+1}(a) & \pi_n(a) & \pi_{n-1}(a) \\
\pi_{n+1}(b) & \pi_n(b) & \pi_{n-1}(b)
\end{vmatrix} = 0 \quad (3.282)
\]

which is proportional to \( \rho_{n+1}(x) \),

\[
\rho_{n+1}(x) = \begin{vmatrix}
p_{n+1}(x) & p_n(x) & p_{n-1}(x) \\
p_{n+1}(a) & p_n(a) & p_{n-1}(a) \\
p_{n+1}(b) & p_n(b) & p_{n-1}(b)
\end{vmatrix} \quad (3.283)
\]

therefore, the roots \( \{x_i\} \) are also the roots of \( \rho_{n+1}(x) \), and

\[
x_i\Psi^*(x_i) = J_{n+1}^*\Psi^*(x_i) \quad (3.284)
\]

so \( x_i \) is an eigenvalue of \( J_{n+1}^* \) with eigenvector \( \Psi^*(x_i) \). Hence, eigenvalues of \( J_{n+1}^* \)
are indeed roots of \( \rho_{n+1}(x) \), i.e., the grid points of the Gauss-Lobatto quadrature.

Similar to Lemma 3.3.17, the following equation can be proved,

\[
\Psi^*(x_i)^T\Psi^*(x_i) = \sum_{k=0}^{n-1} \frac{p_k(x_i)p_k(x_i)}{\langle p_k, p_k \rangle} + \frac{b_n}{b_n^*} \frac{p_n(x_i)p_n(x_i)}{\langle p_n, p_n \rangle} = \frac{1}{w_i} \quad (3.285)
\]

therefore, the orthonormal eigenvector \( \mathbf{v}_1 = \pm \sqrt{w_i}\Psi^*(x_i) \), then

\[
w_i = \frac{[\mathbf{v}_1]^2}{\varphi_0^2(x_i)} = \left[ \frac{\pi_0}{\|\pi_0\|} \right]^2 = \langle \pi_0, \pi_0 \rangle [\mathbf{v}_1]^2 = u_0[\mathbf{v}_1]^2 \quad (3.286)
\]
where $[\mathbf{v}_1]_0$ is the first element of the orthonormal vector $\mathbf{v}_i$. \hfill \Box

**Lemma 3.5.9** (dual orthogonality for Lobatto quadrature). For $0 \leq i, j \leq n$, and $x_0, x_1, \ldots, x_n$ are the distinct roots of $\rho_{n+1}(x)$, we have the following equation,

$$
\sum_{k=0}^{n-1} \frac{p_k(x_i)p_k(x_j)}{\langle p_k, p_k \rangle} + \frac{b_n p_n(x_i)p_n(x_j)}{b_n^n \langle p_n, p_n \rangle} = \frac{\delta_{ij}}{\sqrt{w_i w_j}} \quad (3.287)
$$

**Proof.** From (3.280), the roots of $\rho_{n+1}(x)$ follow the following equation,

$$
\pi_{n+1}(x) + (a_n - a_n^*)\pi_n(x) + (b_n - b_n^*)\pi_{n-1}(x) = 0 \quad (3.288)
$$

alternatively, using the monic three-term recurrence, it is equivalent to,

$$(x - a_n^*)\pi_n(x) - b_n^*\pi_{n-1}(x) = 0 \quad (3.289)$$

Then, let $Q_{ij}$, $0 \leq i, j \leq n$, be the entries of the matrix $Q$, and define,

$$
Q_{ij} = \begin{cases} \\
\frac{\sqrt{w_i p_j(x_i)}}{\|p_j\|} & \text{if } j \neq n \\
\frac{\sqrt{b_n \sqrt{w_i p_j(x_i)}}}{\sqrt{b_n^*} \|p_j\|} & \text{if } j = n 
\end{cases} \quad (3.290)
$$

then, for $0 \leq i, j < n$, the polynomial $p_i(x)p_j(x)$ has degree less than $2n - 1$, and for $i = n$ or $j = n$, and $i \neq j$, the polynomial $p_i(x)p_j(x)$ has degree no more than $2n - 1$, the original Gauss quadrature still hold exactly, and therefore we have,
\[
\begin{align*}
[Q^T Q]_{i,j} &= \sum_{k=0}^{n} Q_{ki}Q_{kj} \\
&= \sum_{k=0}^{n} \frac{\sqrt{w_k p_i(x_k)}}{\|p_i\|} \frac{\sqrt{w_k p_j(x_k)}}{\|p_j\|} \frac{b_n}{\sqrt{b_n^*}} \\
&= \sum_{k=0}^{n} w_k p_i(x_k)p_j(x_k) \frac{b_n}{\|p_i\| \cdot \|p_j\|} \frac{1}{\sqrt{b_n^*}} \\
&= 0 = \delta_{ij}, \quad i \neq j, \ i = n \text{ or } j = n \quad (3.292)
\end{align*}
\]

finally, for \(i = j = n\), the polynomial \(p_i(x)p_j(x)\) has degree of \(2n\), the original Gauss quadrature doesn’t hold exactly, but the following equation holds precisely,

\[
\begin{align*}
[Q^T Q]_{i,j} &= [Q^T Q]_{n,n} = \sum_{k=0}^{n} Q_{kn}Q_{kn} \\
&= \frac{b_n}{b_n^*} \sum_{k=0}^{n} \frac{\sqrt{w_k p_n(x_k)}}{\|p_n\|} \frac{\sqrt{w_k p_n(x_k)}}{\|p_n\|} \\
&= \frac{b_n}{b_n^*} \sum_{k=0}^{n} w_k p_n(x_k) p_n(x_k) \left\langle p_n, p_n \right\rangle \\
&= \frac{b_n}{b_n^*} \left\langle \pi_n, \pi_n \right\rangle \\
&= \frac{b_n}{b_n^*} \sum_{k=0}^{n} w_k \pi_n(x_k) \pi_n(x_k) \left\langle \pi_n, \pi_n \right\rangle \\
&= \frac{b_n}{b_n^*} \left\langle \pi_n, \pi_n \right\rangle \\
&= 1 = \delta_{ij}, \quad i = j = n \quad (3.293)
\end{align*}
\]

therefore, \(Q^T Q = I\), on the other hand, \(QQ^T = I\) too, then,
\[
[QQ^T]_{i,j} = \sum_{k=0}^{n} Q_{ik} Q_{jk}
\]
\[
= \sum_{k=0}^{n-1} \frac{\sqrt{w_i} p_k(x_i) \sqrt{w_j} p_k(x_j)}{||p_k||} + \frac{b_n}{\|p_n\|} \frac{\sqrt{w_i} p_n(x_i) \sqrt{w_j} p_n(x_j)}{\|p_n\|}
\]
\[
= \sum_{k=0}^{n-1} \frac{\sqrt{w_i} w_j p_k(x_i) p_k(x_j)}{\langle p_k, p_k \rangle} + \frac{b_n}{\|p_n\|} \frac{\sqrt{w_i} w_j p_n(x_i) p_n(x_j)}{\langle p_n, p_n \rangle} = \delta_{ij} \quad (3.294)
\]

Therefore,
\[
\sum_{k=0}^{n-1} \frac{p_k(x_i) p_k(x_j)}{\langle p_k, p_k \rangle} + \frac{b_n}{\|p_n\|} \frac{p_n(x_i) p_n(x_j)}{\langle p_n, p_n \rangle} = \frac{\delta_{ij}}{\sqrt{w_i w_j}} \tag{3.295}
\]

### 3.5.5 Gauss-Lobatto-Legendre Quadrature

Gauss-Lobatto-Legendre quadrature includes both end points as grid points, which is useful for solving problems with closed intervals and enforcing boundary conditions at both ends. It formulates as follows,
\[
\int_{-1}^{1} f(x) dx = \sum_{i=0}^{n} w_i g(x_i) = w_0 f(-1) + \sum_{i=1}^{n-1} w_i f(x_i) + w_n f(1) \tag{3.295}
\]

The grid points are the roots of Gauss-Lobatto-Legendre polynomial \( \rho_{n+1}(x) \),
\[
\rho_{n+1}(x) = \begin{vmatrix}
P_{n+1}(x) & P_n(x) & P_{n-1}(x) \\
P_{n+1}(-1) & P_n(-1) & P_{n-1}(-1) \\
P_{n+1}(1) & P_n(1) & P_{n-1}(1)
\end{vmatrix} \quad (3.296)
\]

That is,
\[
\rho_{n+1}(x) = (-1)^n \begin{vmatrix}
P_{n+1}(x) & P_n(x) & P_{n-1}(x) \\
-1 & 1 & -1 \\
1 & 1 & 1
\end{vmatrix} \tag{3.297}
\]
therefore, from the properties of Legendre polynomials,

\[ \rho_{n+1}(x) = 2(-1)^n \left[ P_{n+1}(x) - P_{n-1}(x) \right] \]  
(3.298)

\[ = (-1)^{n+1} \frac{4n + 2}{n^2 + n} (1 - x^2) P'_n(x), \quad n \geq 1 \]  
(3.299)

\[ \propto P_{n+1}(x) - P_{n-1}(x) \]  
(3.300)

\[ \propto (1 - x^2) P'_n(x) \]  
(3.301)

For simplicity, we can redefine GLL polynomial \( \rho(x) = P_{n+1}(x) - P_{n-1}(x) \). Then, the Lagrange interpolation polynomials \( \{\ell_i(x)\} \) based on those grid points are,

\[ \ell_i(x) = \frac{\rho_{n+1}(x)}{(x - x_i)\rho'_{n+1}(x_i)} \]

\[ = \frac{P_{n+1}(x) - P_{n-1}(x)}{(x - x_i)\left[P'_{n+1}(x_i) - P'_{n-1}(x_i)\right]} \]

\[ = \frac{P_{n+1}(x) - P_{n-1}(x)}{(2n + 1)(x - x_i)P_n(x_i)} \quad i = 0, 1, \cdots, n \]  
(3.302)

\[ = -\frac{(1 - x^2) P'_n(x)}{n(n + 1)(x - x_i)P_n(x_i)} \quad i = 0, 1, \cdots, n \]  
(3.303)

and the associated weights on the grid points are derived as follows,

\[ w_i = \int_a^b w(x)\ell_i(x)dx = \int_a^b w(x)\frac{P_{n+1}(x) - P_{n-1}(x)}{(2n + 1)(x - x_i)P_n(x_i)}dx \]

\[ = \frac{1}{(2n + 1)P_n^2(x_i)} \int_a^b w(x)\frac{P_{n+1}(x)P_n(x_i) - P_{n-1}(x)P_n(x_i)}{x - x_i}dx \]  
(3.304)

and the integral in the above equation,

\[ \int_a^b w(x)\frac{P_{n+1}(x)P_n(x_i) - P_{n-1}(x)P_n(x_i)}{x - x_i}dx \]

\[ = \int_a^b w(x)\frac{P_{n+1}(x)P_n(x_i) - P_n(x)P_{n+1}(x_i) + P_n(x)P_{n+1}(x_i) - P_{n-1}(x)P_n(x_i)}{x - x_i}dx \]
\[
\int_a^b w(x) \frac{P_{n+1}(x)P_n(x_i) - P_n(x)P_{n+1}(x_i) + P_n(x)P_{n-1}(x_i) - P_{n-1}(x)P_n(x_i)}{x - x_i} \, dx
\]

\[
= \frac{(2n + 2)!}{2^{n+1}[(n + 1)!]^2} \frac{(2n)!}{2^{n}(n!)^2} \langle \pi_n, \pi_n \rangle + \frac{(2n)!}{2^n(n!)^2} \frac{(2n - 2)!}{2^{n-1}[(n - 1)!]^2} \langle \pi_{n-1}, \pi_{n-1} \rangle
\]

\[
= \frac{(2n+2)(2n+1)}{2(n+1)^2} \langle P_n, P_n \rangle + \frac{2n(2n-1)}{2n^2} \langle P_{n-1}, P_{n-1} \rangle
\]

\[
= \frac{2n+1}{n+1} \frac{2}{2n+1} + \frac{2n-1}{n} \frac{2}{2n-1} = \frac{4n+2}{n(n+1)}
\]

Therefore, the weights \( \{w_i\} \) are as follows,

\[
w_i = \frac{1}{(2n+1)P_n^2(x_i) n(n+1)} = \frac{2}{n(n+1)P_n^2(x_i)}, \quad i = 0, 1, \ldots, n
\]

The modified \( a^*_n \) and \( b^*_n \) in the eigenvalue method solving the grid points and weights are the solution of the following linear equation,

\[
\begin{bmatrix}
\pi_n(-1) & \pi_{n-1}(-1) \\
\pi_n(1) & \pi_{n-1}(1)
\end{bmatrix}
\begin{bmatrix}
a^*_n \\
b^*_n
\end{bmatrix}
= \begin{bmatrix}
-\pi_n(-1) \\
\pi_n(1)
\end{bmatrix}
\]

(3.307)

That is,

\[
\begin{bmatrix}
(-1)^n \pi_n(1) & (-1)^{n-1} \pi_{n-1}(1) \\
\pi_n(1) & \pi_{n-1}(1)
\end{bmatrix}
\begin{bmatrix}
a^*_n \\
b^*_n
\end{bmatrix}
= \begin{bmatrix}
-(-1)^n \pi_n(1) \\
\pi_n(1)
\end{bmatrix}
\]

(3.308)

Which can be simplified as,

\[
\begin{bmatrix}
1 & -\pi_{n-1}(1)/\pi_n(1) \\
1 & \pi_{n-1}(1)/\pi_n(1)
\end{bmatrix}
\begin{bmatrix}
a^*_n \\
b^*_n
\end{bmatrix}
= \begin{bmatrix}
-1 \\
1
\end{bmatrix}
\]

(3.309)

Therefore,

\[
a^*_n = 0, \quad b^*_n = \frac{\pi_n(1)}{\pi_{n-1}(1)} = \frac{2n^2}{2n(2n-1)} = \frac{n}{2n-1}, \quad n \geq 1
\]

(3.310)

With \( a_n, b_n \) and the first moment \( u_0 \) as follows,

\[
a_n = 0, \quad n \geq 0, \quad b_n = \frac{n^2}{4n^2 - 1}, \quad n \geq 1, \quad u_0 = 2
\]

(3.311)
3.6 Orthogonal Polynomial Basis Expansion

3.6.1 Gauss-Hermite Basis Function

We have the Gauss-Hermite quadrature as follows,

\[ \int_{-\infty}^{\infty} e^{-x^2} f(x) dx = \sum_{i=0}^{n} w_i f(x_i) \]  

(3.312)

where \( \{x_i\} \) are the roots of the Hermite polynomial \( H_{n+1}(x) \) or the monic Hermite polynomial \( \pi_{n+1}(x) \) and the corresponding weights are,

\[ w_i = \frac{\langle \pi_n, \pi_n \rangle}{\pi_{n+1}(x_i) \pi_n(x_i)} = \frac{2^n n! \sqrt{\pi}}{(n+1)H_n^2(x_i)}, \quad i = 0, 1, \cdots, n \]  

(3.313)

We define the Gauss-Hermite basis function, \( \phi_i(x) \), as the Lagrange interpolation polynomial in the Gauss-Hermite quadrature,

\[ \phi_i(x) = \ell_i(x) = \frac{\pi_{n+1}(x)}{(x - x_i)\pi_n(x_i)}, \quad i = 0, 1, \cdots, n \]  

(3.314)

and from the property of the Lagrange interpolation polynomial, we have,

\[ \phi_j(x_i) = \delta_{ij} \]  

(3.315)

Therefore, any smooth function \( f(x) \) can be approximated by the following interpolation with the above basis functions. The approximation will be exact for any polynomial with order \( n \) or less.

\[ f(x) \approx \sum_{i=0}^{n} \phi_i(x_i) f(x_i) \]  

(3.316)

Gauss-Hermite basis functions can be expressed by Hermite functions as follows,
\[
\phi_i(x) = \frac{\pi_{n+1}(x)}{(x-x_i)\pi'_{n+1}(x_i)} \\
= \frac{H_{n+1}(x)}{(x-x_i)H'_{n+1}(x_i)} \\
= \frac{H_{n+1}(x)}{2(n+1)(x-x_i)H_n(x_i)}
\]

(3.317)

(3.318)

The derivatives of the Gauss-Hermite basis functions at the grid points are,

\[
D_{ij} = \left. \frac{\partial \phi_j(x)}{\partial x} \right|_{x = x_i}
\]

(3.319)

To calculate \(D_{ij}\), first we rearrange the Gauss-Hermite basis function as follows,

\[
(x - x_j)\phi_j(x) = \frac{H_{n+1}(x)}{H'_{n+1}(x_j)}
\]

(3.320)

and take the derivative on both sides, we have,

\[
\phi_j(x) + (x - x_j)\phi'_j(x) = \frac{H'_{n+1}(x)}{H'_{n+1}(x_j)} = \frac{H_n(x)}{H_n(x_j)}
\]

(3.321)

When \(x_i \neq x_j\), we have,

\[
(x_i - x_j)\phi'_j(x_i) = \frac{H_n(x_i)}{H_n(x_j)}
\]

(3.322)

therefore,

\[
\phi'_j(x_i) = \frac{H_n(x_i)}{(x_i - x_j)H_n(x_j)}, \quad i \neq j
\]

(3.323)

When \(x_i = x_j\), take the derivative of (3.321), we have,

\[
\phi'_j(x) + \phi'_j(x) + (x - x_j)\phi''_j(x) = \frac{H'_n(x)}{H_n(x_j)}
\]

(3.324)

therefore,
\[ \phi'_j(x_i) = \frac{H'_n(x_i)}{2H_n(x_j)} = \frac{2x_iH_n(x_i)}{2H_n(x_j)} = x_i, \quad i = j \] (3.325)

Hence, the derivatives of the Gauss-Hermite basis functions at the grid points for \( 0 \leq i \leq n \) and \( 0 \leq j \leq n \), can be summarized as follows,

\[ D_{ij} = \frac{\partial \phi_j(x_i)}{\partial x} = \begin{cases} \frac{H_n(x_i)}{(x_i - x_j)H_n(x_j)} & \text{if } i \neq j \\ x_i & \text{if } i = j \end{cases} \] (3.326)

The second derivatives of the Gauss-Hermite basis functions at grid points are,

\[ B_{ij} = \phi''_j(x_i) = \frac{\partial^2 \phi_j(x)}{\partial x^2} \bigg|_{x = x_i} \] (3.327)

When \( x_i \neq x_j \), from (3.324) we have

\[ 2\phi'_j(x_i) + (x_i - x_j)\phi''_j(x_i) = \frac{H'_n(x_i)}{H_n(x_j)} \] (3.328)

therefore,

\[ (x_i - x_j)\phi''_j(x_i) = \frac{H'_n(x_i)}{H_n(x_j)} - 2\phi'_j(x_i) \]

\[ = \frac{H'_n(x_i)}{H_n(x_j)} - \frac{2H_n(x_i)}{(x_i - x_j)H_n(x_j)} \]

\[ = \frac{(x_i - x_j)H'_n(x_i) - 2H_n(x_i)}{(x_i - x_j)H_n(x_j)} \]

\[ = \frac{(x_i - x_j)2x_iH_n(x_i) - 2H_n(x_i)}{(x_i - x_j)H_n(x_j)} \]

\[ = \frac{2(x_i^2 - x_ix_j - 1)H_n(x_i)}{(x_i - x_j)H_n(x_j)} \] (3.329)

85
hence,
\[ \phi''_j(x_i) = \frac{2(x_i^2 - x_ix_j - 1)H_n(x_i)}{(x_i - x_j)^2H_n(x_j)}, \quad i \neq j \] (3.330)

When \( x_i = x_j \), take the derivative of (3.324), we have,
\[ 2\phi''_j(x) + \phi''_j(x) + (x - x_j)\phi'''_j(x) = \frac{H''_n(x)}{H_n(x_j)} \] (3.331)

therefore,
\[ \phi''_j(x_i) = \frac{H''_n(x_i)}{3H_n(x_j)} \]
\[ = \frac{2x_iH'_n(x_i) - 2nH_n(x_i)}{3H_n(x_j)} \]
\[ = \frac{4x_i^2H_n(x_i) - 2nH_n(x_i)}{3H_n(x_j)} \]
\[ = \frac{4x_i^2 - 2n}{3}, \quad i = j \] (3.332)

Hence, the second derivatives of the Gauss-Hermite basis functions at the grid points for \( 0 \leq i \leq n \) and \( 0 \leq j \leq n \), can be summarized as follows,
\[
B_{ij} = \phi''_j(x_i) = \frac{\partial^2 \phi_j(x_i)}{\partial x^2} = \begin{cases} 
  \frac{2(x_i^2 - x_ix_j - 1)H_n(x_i)}{(x_i - x_j)^2H_n(x_j)} & \text{if } i \neq j \\
  \frac{4x_i^2 - 2n}{3} & \text{if } i = j 
\end{cases} \] (3.333)

3.6.2 Stable Gauss-Hermite Basis Function

The modified Hermite polynomials are as follows,
\[ \mathcal{H}_n(x) = \frac{1}{\sqrt{2^n n!}}e^{-x^2/2}H_n(x), \quad n \geq 0 \] (3.334)
Therefore, the modified Gauss-Hermite basis functions are,

\[ \psi_i(x) = \phi_i(x) \frac{e^{-x^2/2}}{e^{-x_i^2/2}} \quad (3.335) \]

\[ \psi_j(x) \] at the grid points still hold the following property,

\[ \psi_j(x_i) = \delta_{ij} \quad (3.336) \]

Hence, any smooth function \( g(x) \) with the property of quadratic exponential decay can be approximated by the following interpolation with the above modified basis functions. The approximation will be exact if \( f(x) = g(x)e^{-x^2/2} \) and \( g(x) \) is any polynomial with order \( n \) or less.

\[ f(x) \approx \sum_{i=0}^{n} \psi_i(x)f(x_i) \quad (3.337) \]

When \( g(x) \) is any polynomial with order \( n \) or less,

\[ g(x) = \sum_{i=0}^{n} \phi_i(x)g(x_i) \quad (3.338) \]

then, let the above equation times \( e^{-x^2/2} \) on both sides,

\[ g(x)e^{-x^2/2} = \sum_{i=0}^{n} \phi_i(x)e^{-x^2/2}g(x_i) \quad (3.339) \]

we have,

\[ g(x)e^{-x^2/2} = \sum_{i=0}^{n} \phi_i(x) \frac{e^{-x^2/2}}{e^{-x_i^2/2}}g(x_i)e^{-x_i^2/2} \quad (3.340) \]

therefore,

\[ f(x) = \sum_{i=0}^{n} \psi_i(x)f(x_i) \quad (3.341) \]
The analytical formula for the modified Gauss-Hermite basis functions is,

\[
\psi_i(x) = \phi_i(x) \frac{e^{-x^2/2}}{e^{-x_i^2/2}} = \frac{H_{n+1}(x)e^{-x^2/2}}{2(n + 1)(x - x_i)H_n(x_i)e^{-x_i^2/2}} = \frac{H_{n+1}(x)}{\sqrt{2(n + 1)(x - x_i)H_n(x_i)}}
\] (3.342)

The modified Gauss-Hermite basis functions with order of \( n = 2 \) are shown in the following Figure 3.4. The interpolation points are the roots of the related Hermite polynomial \( H_3(x) \) or the modified polynomial \( \mathcal{H}_3(x) \).

![Figure 3.4: Modified Gauss-Hermite basis function \( \psi_i(x), \, i = 0, 1, 2 \)](image)

Then, the derivatives of the modified Gauss-Hermite basis functions are,

\[
\psi'_i(x) = \left[ \phi'_i(x) - x\phi_j(x) \right] e^{-x^2/2} e^{-x_i^2/2}
\] (3.343)

Hence, the derivatives of \( \psi_j(x) \) at the grid points are,
\[ D_{ij} = \frac{\partial^2 \psi_j(x)}{\partial x^2} \bigg|_{x = x_i} \]
\[ = D_{ij} \frac{e^{-x_i^2/2}}{e^{-x_j^2/2}} - x_i \delta_{ij} \]
\[ = \begin{cases} 
H_n(x_i) / (x_i - x_j) H_n(x_j) & \text{if } i \neq j \\
0 & \text{if } i = j 
\end{cases} \] (3.344)

And the second derivatives of the modified Gauss-Hermite basis functions are,

\[ \psi_j''(x) = \left[ \phi_j''(x) - 2x \phi_j'(x) + (x^2 - 1) \phi_j(x) \right] \frac{e^{-x^2/2}}{e^{-x_j^2/2}} \] (3.345)

Hence, the second derivatives of \( \psi_j(x) \) at the grid points are,

\[ \psi_j''(x_i) = \left[ \phi_j''(x_i) - 2x_i D_{ij} + (x_i^2 - 1) \delta_{ij} \right] \frac{e^{-x_i^2/2}}{e^{-x_j^2/2}} \] (3.346)

When \( x_i \neq x_j \),

\[ \psi_j''(x_i) = \left[ 2(x_i^2 - x_i x_j - 1) H_n(x_i) / (x_i - x_j)^2 H_n(x_j) - 2x_i H_n(x_i) / (x_i - x_j) H_n(x_j) \right] \frac{e^{-x_i^2/2}}{e^{-x_j^2/2}} \]
\[ = \frac{-2H_n(x_i)}{(x_i - x_j)^2 H_n(x_j)} \frac{e^{-x_i^2/2}}{e^{-x_j^2/2}} \]
\[ = \frac{-2H_n(x_i)}{(x_i - x_j)^2 H_n(x_j)} \] (3.347)

When \( x_i = x_j \),

\[ \psi_j''(x_i) = \left[ \phi_j''(x_i) - 2x_i D_{ij} + (x_i^2 - 1) \delta_{ij} \right] \frac{e^{-x_i^2/2}}{e^{-x_j^2/2}} \]
\[ = \frac{4x_i^2 - 2n}{3} - 2x_i^2 + (x_i^2 - 1) \]
\[ = \frac{x_i^2 - 2n - 3}{3} \] (3.348)
Hence, the second derivatives of the modified Gauss-Hermite basis functions at the grid points for $0 \leq i \leq n$ and $0 \leq j \leq n$, can be summarized as follows,

$$B_{ij} = \psi_j''(x_i) = \begin{cases} -\frac{2\mathcal{H}_n(x_i)}{(x_i - x_j)^2\mathcal{H}_n(x_j)} & \text{if } i \neq j \\ \frac{x_i^2 - 2n - 3}{3} & \text{if } i = j \end{cases} \quad (3.349)$$

### 3.6.3 Gauss-Radau-Laguerre Basis Function

We have the Gauss-Radau-Laguerre quadrature as follows,

$$\int_0^\infty e^{-x} f(x) dx = \sum_{i=0}^{n} w_i f(x_i) = w_0 f(0) + \sum_{i=1}^{n} w_i f(x_i) \quad (3.350)$$

where $\{x_i\}$ are the roots of $L_{n+1}(x) - L_n(x)$ or $xL_{n+1}'(x)$ and the corresponding weights associated with those grid points are,

$$w_i = \frac{1}{(n + 1)L_n^2(x_i)}, \quad i = 0, 1, \cdots, n \quad (3.351)$$

We define the Gauss-Radau-Laguerre basis function, $\phi_i(x)$, as the Lagrange interpolation polynomial in the Gauss-Radau-Laguerre quadrature,

$$\phi_i(x) \equiv \ell_i(x) = -\frac{L_{n+1}(x) - L_n(x)}{(x - x_i)L_n(x_i)}, \quad (3.352)$$

$$= -\frac{xL_{n+1}'(x_i)}{(n + 1)(x - x_i)L_n(x_i)!}, \quad i = 0, 1, \cdots, n \quad (3.353)$$

and from the property of the Lagrange interpolation polynomial, we have,

$$\phi_j(x_i) = \delta_{ij} \quad (3.354)$$

Therefore, any smooth function $f(x)$ can be approximated by the following interpolation with the above basis functions. The approximation will be exact for any polynomial with order $n$ or less.
\[
f(x) \approx \sum_{i=0}^{n} \phi_i(x) f(x_i)
\]  

(3.355)

The derivatives of the Gauss-Radau-Laguerre basis functions at grid points are,

\[
D_{ij} = \left. \frac{\partial \phi_j(x)}{\partial x} \right|_{x = x_i}
\]  

(3.356)

To calculate \( D_{ij} \), first we rearrange Gauss-Radau-Laguerre basis function as follows,

\[
(x - x_j) \phi_j(x) = \frac{L_n(x) - L_{n+1}(x)}{L_n(x_j)}
\]  

(3.357)

and take the derivative on both sides, we have,

\[
\phi_j(x) + (x - x_j) \phi'_j(x) = \frac{L'_n(x) - L'_{n+1}(x)}{L_n(x_j)} = \frac{L_n(x)}{L_n(x_j)}
\]  

(3.358)

When \( x_i \neq x_j \), we have,

\[
(x_i - x_j) \phi'_j(x_i) = \frac{L_n(x_i)}{L_n(x_j)}
\]  

(3.359)

therefore,

\[
\phi'_j(x_i) = \frac{L_n(x_i)}{(x_i - x_j)L_n(x_j)}, \quad i \neq j
\]  

(3.360)

When \( x_i = x_j \), take the derivative of (3.358), we have,

\[
\phi'_j(x) + \phi'_j(x) + (x - x_j) \phi''_j(x) = \frac{L'_n(x)}{L_n(x_j)}
\]  

(3.361)

and when \( x_i \neq 0 \), \( L'_{n+1}(x_i) = 0 \), and from (3.156), we have,

\[
L'_n(x_i) = L'_n(x_i) - L'_{n+1}(x_i) = L_n(x_i)
\]  

(3.362)

therefore, when \( x_i = x_j \neq 0 \), we have,

\[
\phi'_j(x_i) = \frac{L'_n(x_i)}{2L_n(x_j)} = \frac{L_n(x_i)}{2L_n(x_j)} = \frac{1}{2}, \quad i = j \neq 0
\]  

(3.363)
and when \( x_i = x_j = 0 \),

\[
\phi'_j(x_i) = \frac{L'_n(0)}{2L_n(0)} = -\frac{n}{2}, \quad i = j = 0
\]  

(3.364)

Hence, the derivatives of the Gauss-Radau-Laguerre basis functions at the grid points for \( 0 \leq i \leq n \) and \( 0 \leq j \leq n \), can be summarized as follows,

\[
D_{ij} = \frac{\partial \phi_j(x_i)}{\partial x} = \begin{cases} 
\frac{L_n(x_i)}{(x_i - x_j)L_n(x_j)} & \text{if } i \neq j \\
\frac{1}{2} & \text{if } i = j \neq 0 \\
-\frac{n}{2} & \text{if } i = j = 0 
\end{cases}
\]  

(3.365)

Second derivatives of Gauss-Radau-Laguerre basis functions at grid points are,

\[
B_{ij} = \phi''_j(x_i) = \left. \frac{\partial^2 \phi_j(x)}{\partial x^2} \right|_{x = x_i}
\]  

(3.366)

When \( x_i \neq x_j \), from (3.361) we have

\[
2\phi'_j(x_i) + (x_i - x_j)\phi''_j(x_i) = \frac{L'_n(x_i)}{L_n(x_j)}
\]  

(3.367)

therefore, when \( x_i \neq x_j \) and \( x_i \neq 0 \), we have,

\[
(x_i - x_j)\phi''_j(x_i) = \frac{L'_n(x_i)}{L_n(x_j)} - 2\phi'_j(x_i)
\]

\[
= \frac{L_n(x_i)}{L_n(x_j)} - \frac{2L_n(x_i)}{(x_i - x_j)L_n(x_j)}
\]

\[
= \frac{(x_i - x_j)L_n(x_i) - 2L_n(x_i)}{(x_i - x_j)L_n(x_j)}
\]

\[
= \frac{(x_i - x_j - 2)L_n(x_i)}{(x_i - x_j)^2L_n(x_j)}
\]  

(3.368)
hence,

$$\phi_j''(x_i) = \frac{(x_i - x_j - 2)L_n(x_i)}{(x_i - x_j)^2 L_n(x_j)}, \quad i \neq j, \ i \neq 0 \quad (3.369)$$

and when $x_i \neq x_j$ and $x_i = 0$, we have,

$$(x_i - x_j)\phi_j''(x_i) = \frac{L'_n(0)}{L_n(x_j)} - 2\phi'_j(0)$$

$$= \frac{-n}{L_n(x_j)} - \frac{2L_n(0)}{(x_i - x_j)L_n(x_j)}$$

$$= \frac{-n(x_i - x_j) - 2}{(x_i - x_j)L_n(x_j)} = \frac{nx_j - 2}{-x_jL_n(x_j)} \quad (3.370)$$

hence,

$$\phi_j''(x_i) = \frac{nx_j - 2}{x_j^2 L_n(x_j)}, \quad i \neq j, \ i = 0 \quad (3.371)$$

When $x_i = x_j$, take the derivative of (3.361), we have,

$$2\phi_j''(x) + \phi_j''(x) + (x - x_j)\phi_j'''(x) = \frac{L''_n(x)}{L_n(x_j)} \quad (3.372)$$

and so the second derivatives when $x_i = x_j$ are,

$$\phi_j''(x_i) = \frac{L''_n(x_j)}{3L_n(x_j)} \quad (3.373)$$

therefore, when $x_i = x_j \neq 0$, we have,

$$\phi_j''(x_i) = \frac{x_iL''_n(x_i)}{3x_iL_n(x_j)}$$

$$= -\frac{(1 - x_i)L'_n(x_i) + nL_n(x_i)}{3x_iL_n(x_j)}$$

$$= -\frac{(1 - x_i)L_n(x_i) + nL_n(x_i)}{3x_iL_n(x_j)}$$

$$= \frac{x_i - 1 - n}{3x_i}, \quad i = j \neq 0 \quad (3.374)$$
and when \( x_i = x_j = 0 \), we have,

\[
\phi''(x_i) = \frac{L''_n(0)}{3} = -\frac{\sum_{k=0}^{n-1} L'_k(0)}{3} = \frac{\sum_{k=0}^{n-1} k}{3} = \frac{(n-1)n}{6}, \quad i = j = 0
\] (3.375)

Hence, the second derivatives of the Gauss-Radau-Laguerre basis functions at the grid points for \( 0 \leq i \leq n \) and \( 0 \leq j \leq n \), can be summarized as follows,

\[
B_{ij} = \phi''(x_i) = \frac{\partial^2 \phi_j(x_i)}{\partial x^2} = \begin{cases} 
\frac{(x_i - x_j - 2)L_n(x_i)}{(x_i - x_j)^2 L_n(x_j)} & \text{if } i \neq j, i \neq 0 \\
\frac{nx_j - 2}{x^2_j L_n(x_j)} & \text{if } i \neq j, i = 0 \\
\frac{x_i - 1 - n}{3x_i} & \text{if } i = j \neq 0 \\
\frac{(n-1)n}{6} & \text{if } i = j = 0
\end{cases}
\] (3.376)

### 3.6.4 Stable Gauss-Radau-Laguerre Basis Function

The modified Laguerre polynomials are as follows,

\[
\mathcal{L}_n(x) = e^{-x/2}L_n(x), \quad n \geq 0
\] (3.377)

Therefore, the modified Gauss-Radau-Laguerre basis functions are,

\[
\psi_i(x) = \phi_i(x) \frac{e^{-x/2}}{e^{-x_i/2}}
\] (3.378)

\( \psi_j(x) \) at the grid points still hold the following property,

\[
\psi_j(x_i) = \delta_{ij}
\] (3.379)
Hence, any smooth function \( f(x) \) with the property of exponential decay at positive infinity can be approximated by the following interpolation with the above modified basis functions. The approximation will be exact if \( f(x) = g(x)e^{-x/2} \) and \( g(x) \) is any polynomial with order \( n \) or less.

\[
\begin{align*}
f(x) & \approx \sum_{i=0}^{n} \psi_i(x) f(x_i) \\
& (3.380)
\end{align*}
\]

When \( g(x) \) is any polynomial with order \( n \) or less,

\[
\begin{align*}
g(x) & = \sum_{i=0}^{n} \phi_i(x) g(x_i) \\
& (3.381)
\end{align*}
\]

then, let the above equation times \( e^{-x/2} \) on both sides,

\[
\begin{align*}
g(x)e^{-x/2} & = \sum_{i=0}^{n} \phi_i(x)e^{-x/2} g(x_i) \\
& (3.382)
\end{align*}
\]

we have,

\[
\begin{align*}
g(x)e^{-x/2} & = \sum_{i=0}^{n} \phi_i(x) \frac{e^{-x/2}}{e^{-x_i/2}} g(x_i)e^{-x_i/2} \\
& (3.383)
\end{align*}
\]

therefore,

\[
\begin{align*}
f(x) & = \sum_{i=0}^{n} \psi_i(x) f(x_i) \\
& (3.384)
\end{align*}
\]

The analytical formula for modified Gauss-Radau-Laguerre basis functions is,

\[
\begin{align*}
\psi_i(x) & = \phi_i(x) \frac{e^{-x/2}}{e^{-x_i/2}} \\
& = - \frac{L_{n+1}(x) - L_n(x)}{(x-x_i)L_n(x_i)} \cdot \frac{e^{-x/2}}{e^{-x_i/2}} \\
& = \frac{\mathcal{L}_n(x) - \mathcal{L}_{n+1}(x)}{(x-x_i)L_n(x_i)} \\
& (3.385)
\end{align*}
\]
The modified Gauss-Radau-Laguerre basis functions with order of $n = 2$ are shown in Figure 3.5. The interpolation points are the roots of the related Gauss-Radau-Laguerre polynomial $\rho(x) = L_3(x) - L_2(x)$.

![Figure 3.5: Modified Gauss-Radau-Laguerre basis function $\psi_i(x)$, $i = 0, 1, 2$](image)

Then, the derivatives of the modified Gauss-Hermite basis functions are,

$$
\psi'_j(x) = \left[ \phi'_j(x) - \frac{1}{2} \phi_j(x) \right] \frac{e^{-x/2}}{e^{-x_j/2}}
$$

hence, the derivatives of $\psi_j(x)$ at the grid points are,

$$
D_{ij} = \frac{\partial \psi_j(x)}{\partial x} \bigg|_{x = x_i} = D_{ij} \frac{e^{-x_i/2}}{e^{-x_j/2}} - \frac{1}{2} \delta_{ij}
$$

$$
\begin{align*}
D_{ij} &= \begin{cases} 
\mathcal{L}_n(x_i) \\
\frac{L_n(x_i)}{(x_i - x_j)L_n(x_j)} & \text{if } i \neq j \\
0 & \text{if } i = j 
eq 0 \\
-\frac{n + 1}{2} & \text{if } i = j = 0
\end{cases}
\end{align*}
$$
The second derivatives of the modified Gauss-Radau-Laguerre basis functions are,

\[
\psi''_j(x) = \left[ \phi''_j(x) - \phi'_j(x) + \frac{1}{4} \phi_j(x) \right] \frac{e^{-x/2}}{e^{-x_j/2}}
\] (3.388)

hence, the second derivatives of \(\psi_j(x)\) at the grid points are,

\[
\psi''_j(x_i) = \left[ \phi''_j(x_i) - D_{ij} + \frac{1}{4} \delta_{ij} \right] \frac{e^{-x_i/2}}{e^{-x_j/2}}
\] (3.389)

When \(x_i \neq x_j\) and \(x_i \neq 0\),

\[
\psi''_j(x_i) = \left[ \frac{(x_i - x_j - 2)L_n(x_i)}{(x_i - x_j)^2L_n(x_j)} - \frac{L_n(x_i)}{(x_i - x_j)L_n(x_j)} \right] \frac{e^{-x_i/2}}{e^{-x_j/2}}
\]

\[
= - \frac{2L_n(x_i)}{(x_i - x_j)^2L_n(x_j)} e^{-x_i/2}
\]

\[
= - \frac{2L_n(x_i)}{(x_i - x_j)^2L_n(x_j)}, \quad i \neq j, \quad i \neq 0
\] (3.390)

When \(x_i \neq x_j\) and \(x_i = 0\),

\[
\psi''_j(x_i) = \left[ \frac{nx_j - 2}{x_j^2L_n(x_j)} + \frac{1}{x_jL_n(x_j)} \right] \frac{1}{e^{-x_j/2}}
\]

\[
= \left( \frac{n + 1}{x_j^2L_n(x_j)} \right) \cdot \frac{1}{e^{-x_j/2}}
\]

\[
= \left( \frac{n + 1}{x_j^2L_n(x_j)} \right), \quad i \neq j, \quad i = 0
\] (3.391)

When \(x_i = x_j \neq 0\),

\[
\psi''_j(x_i) = \frac{x_i - 1 - n}{3x_i} - \frac{1}{2} + \frac{1}{4} = \frac{x_i - 4n - 4}{12x_i}, \quad i = j \neq 0
\] (3.392)

When \(x_i = x_j = 0\),

\[
\psi''_j(x_i) = \frac{n(n - 1)}{6} + \frac{n}{2} + \frac{1}{4} = \frac{2n^2 + 4n + 3}{12}, \quad i = j = 0
\] (3.393)

97
Therefore, the second derivatives of the modified Gauss-Radau-Laguerre basis functions at the grid points for $0 \leq i \leq n$ and $0 \leq j \leq n$, can be summarized as follows,

$$B_{ij} = \psi_j''(x_i) = \begin{cases} 
-\frac{2L_n(x_i)}{(x_i-x_j)^2L_n(x_j)} & \text{if } i \neq j, i \neq 0 \\
-\frac{(n+1)x_j - 2}{x_j^2L_n(x_j)} & \text{if } i \neq j, i = 0 \\
\frac{x_i - 4n - 4}{12x_i} & \text{if } i = j \neq 0 \\
\frac{2n^2 + 4n + 3}{12} & \text{if } i = j = 0
\end{cases} \quad (3.394)$$

3.6.5 Gauss-Lobatto-Legendre Basis Function

We have the Gauss-Lobatto-Legendre quadrature as follows,

$$\int_{-1}^{1} f(x)dx = \sum_{i=0}^{n} w_ig(x_i) = w_0f(-1) + \sum_{i=1}^{n-1} w_if(x_i) + w_nf(1) \quad (3.395)$$

where $\{x_i\}$ are the roots of $P_{n+1}(x) - P_{n-1}(x)$ or $(1-x^2)P'_n(x)$ and the corresponding weights associated with those grid points are,

$$w_i = \frac{2}{n(n+1)P_n^2(x_i)}, \quad i = 0, 1, \cdots, n \quad (3.396)$$

We define the Gauss-Lobatto-Legendre basis function, $\phi_i(x)$, as the Lagrange interpolation polynomial in the Gauss-Lobatto-Legendre quadrature,

$$\phi_i(x) \equiv \ell_i(x) = \frac{P_{n+1}(x) - P_{n-1}(x)}{(2n+1)(x-x_i)P_n(x_i)}, \quad i = 0, 1, \cdots, n \quad (3.397)$$

$$= -\frac{(1-x^2)P'_n(x)}{n(n+1)(x-x_i)P_n(x_i)}, \quad i = 0, 1, \cdots, n \quad (3.398)$$

and from the property of the Lagrange interpolation polynomial, we have,

$$\phi_j(x_i) = \delta_{ij} \quad (3.399)$$
Therefore, any locally smooth functions on $[-1, 1]$ can be approximated by the following interpolation by the above basis functions. The approximation will be exact for any polynomial with order $n$ or less.

$$f(x) \approx \sum_{i=0}^{n} \phi_i(x)f(x_i) \quad (3.400)$$

The Gauss-Lobatto-Legendre basis functions with order of $n = 3$ are shown in Figure 3.6. The interpolation points are the roots of the related Gauss-Lobatto-Legendre polynomial $\rho(x) = P_4(x) - P_2(x)$.

![Figure 3.6: Gauss-Lobatto-Legendre basis function $\phi_i(x)$, $i = 0, \ldots, 3$](image)

Then, the derivatives of the Gauss-Lobatto-Legendre basis functions at grid points are,

$$D_{ij} = \left. \frac{\partial \phi_j(x)}{\partial x} \right|_{x = x_i} \quad (3.401)$$

To calculate $D_{ij}$, first we rearrange Gauss-Lobatto-Legendre basis function as follows,

$$(x - x_j)\phi_j(x) = \frac{P_{n+1}(x) - P_{n-1}(x)}{(2n + 1)P_n(x_j)} \quad (3.402)$$
and take the derivative on both sides, we have,

\[ \phi_j(x) + (x - x_j)\phi'_j(x) = \frac{P'_{n+1}(x) - P'_{n-1}(x)}{(2n + 1)P_n(x)} = \frac{P_n(x)}{P_n(x)} \]  

(3.403)

When \( x_i \neq x_j \), we have,

\[ (x_i - x_j)\phi'_j(x_i) = \frac{P_n(x_i)}{P_n(x_j)} \]  

(3.404)

therefore,

\[ \phi'_j(x_i) = \frac{P_n(x_i)}{(x_i - x_j)P_n(x_j)}, \quad i \neq j \]  

(3.405)

When \( x_i = x_j \), take the derivative of (3.403), we have,

\[ \phi'_j(x) + \phi'_j(x) + (x - x_j)\phi''_j(x) = \frac{P'_n(x)}{P_n(x)} \]  

(3.406)

therefore,

\[ \phi'_j(x_i) = \frac{P'_n(x_i)}{2P_n(x_j)}, \quad i = j \]  

(3.407)

and when \( x_i = x_j \neq \pm 1 \), we have \( P'_n(x_i) = 0 \), then

\[ \phi'_j(x_i) = \frac{P_n(x_i)}{2P_n(x_j)} = 0, \quad i = j \neq 0, i = j \neq n \]  

(3.408)

and when \( x_i = x_j = -1 \),

\[ \phi'_j(x_i) = \frac{P'_n(-1)}{2P_n(-1)} = \frac{(-1)^{n+1}n(n + 1)/2}{2(-1)^n} = -\frac{n(n + 1)}{4}, \quad i = j = 0 \]  

(3.409)

and when \( x_i = x_j = 1 \),

\[ \phi'_j(x_i) = \frac{P'_n(1)}{2P_n(1)} = \frac{n(n + 1)/2}{4} = \frac{n(n + 1)}{4}, \quad i = j = n \]  

(3.410)
Hence, the derivatives of the Gauss-Lobatto-Legendre basis functions at the grid points for $0 \leq i \leq n$ and $0 \leq j \leq n$, can be summarized as follows,

$$D_{ij} = \frac{\partial\phi_j(x_i)}{\partial x} = \begin{cases} \frac{P_n(x_i)}{(x_i - x_j)P_n(x_j)} & \text{if } i \neq j \\ 0 & \text{if } i = j \neq 0, n \\ -\frac{n(n+1)}{4} & \text{if } i = j = 0 \\ \frac{n(n+1)}{4} & \text{if } i = j = n \end{cases} \quad (3.411)$$

Second derivatives of Gauss-Lobatto-Legendre basis functions at grid points are,

$$B_{ij} = \phi''_j(x_i) = \left. \frac{\partial^2 \phi_j(x)}{\partial x^2} \right|_{x = x_i} \quad (3.412)$$

When $x_i \neq x_j$, from $(3.406)$ we have

$$2\phi'_j(x_i) + (x_i - x_j)\phi''_j(x_i) = \frac{P'_n(x_i)}{P_n(x_j)} \quad (3.413)$$

therefore, when $x_i \neq x_j$ and $x_i \neq \pm 1$,

$$2\phi'_j(x_i) + (x_i - x_j)\phi''_j(x_i) = 0 \quad (3.414)$$

hence,

$$\phi''_j(x_i) = -\frac{2}{x_i - x_j} \cdot \frac{P_n(x_i)}{(x_i - x_j)P_n(x_j)} = -\frac{2P_n(x_i)}{(x_i - x_j)^2P_n(x_j)}, \quad i \neq j, i \neq 0, i \neq n \quad (3.415)$$

and when $x_i \neq x_j$ and $x_i = -1$,

$$2\phi'_j(x_i) + (x_i - x_j)\phi''_j(x_i) = \frac{(-1)^{n+1}n(n+1)/2}{P_n(x_j)} \quad (3.416)$$

101
hence,

\[
\phi_j''(x_i) = \frac{(-1)^{n+1}n(n+1)/2}{(x_i - x_j)P_n(x_j)} - \frac{2(-1)^n}{(x_i - x_j)^2P_n(x_j)}
\]

\[
= (-1)^{n+1}n(n+1)(x_i - x_j) + 4 \frac{n(n+1)/2}{2(x_i - x_j)^2P_n(x_j)}, \quad i \neq j, \quad i = 0 \tag{3.417}
\]

and when \(x_i \neq x_j\) and \(x_i = 1,\)

\[
2\phi_j'(x_i) + (x_i - x_j)\phi_j''(x_i) = \frac{n(n+1)/2}{P_n(x_j)} \tag{3.418}
\]

hence,

\[
\phi_j''(x_i) = \frac{n(n+1)/2}{(x_i - x_j)P_n(x_j)} - \frac{2}{(x_i - x_j)^2P_n(x_j)}
\]

\[
= \frac{n(n+1)(x_i - x_j) - 4}{2(x_i - x_j)^2P_n(x_j)}, \quad i \neq j, \quad i = n \tag{3.419}
\]

When \(x_i = x_j,\) take the derivative of (3.406), we have,

\[
2\phi_j''(x) + \phi_j''(x) + (x - x_j)\phi_j'''(x) = \frac{P_n''(x)}{P_n(x_j)} \tag{3.420}
\]

therefore,

\[
\phi_j''(x_i) = \frac{P_n''(x)}{3P_n(x_j)}, \quad i = j \tag{3.421}
\]

and when \(x_i = x_j \neq \pm 1,\)

\[
\phi_j''(x_i) = \frac{P_n''(x)}{3P_n(x_j)}
\]

\[
= \frac{(1 - x_i^2)P_n''(x_i)}{3(1 - x_i^2)P_n(x_j)}
\]

\[
= -\frac{n(n+1)P_n(x_i)}{3(1 - x_i^2)P_n(x_j)}
\]

\[
= -\frac{n(n+1)}{3(1 - x_i^2)}, \quad i = j \neq 0, \quad i = j \neq n \tag{3.422}
\]
and when $x_i = x_j = -1,$

$$\phi_j''(x_i) = \frac{P_n''(-1)}{3P_n(-1)}$$

$$= \frac{1}{3(-1)^n} \frac{(n-1)n(n+1)(n+2)}{8}$$

$$= \frac{(n-1)n(n+1)(n+2)}{24}, \quad i = j = 0 \quad (3.423)$$

and when $x_i = x_j = 1,$

$$\phi_j''(x_i) = \frac{P_n''(1)}{3P_n(1)}$$

$$= \frac{1}{3} \frac{(n-1)n(n+1)(n+2)}{8}$$

$$= \frac{(n-1)n(n+1)(n+2)}{24}, \quad i = j = n \quad (3.424)$$

Hence, the second derivatives of the Gauss-Lobatto-Legendre basis functions at the grid points for $0 \leq i \leq n$ and $0 \leq j \leq n,$ can be summarized as follows,

$$B_{ij} = \phi_j''(x_i) = \begin{cases} 
-\frac{2P_n(x_i)}{(x_i - x_j)^2P_n(x_j)} & \text{if } i \neq j, i \neq 0, n \\
(-1)^{n+1} \frac{n(n+1)(x_i - x_j) + 4}{2(x_i - x_j)^2P_n(x_j)} & \text{if } i \neq j, i = 0 \\
\frac{n(n+1)(x_i - x_j) - 4}{2(x_i - x_j)^2P_n(x_j)} & \text{if } i \neq j, i = n \quad (3.425) \\
-\frac{n(n+1)}{3(1-x_i^2)} & \text{if } i = j \neq 0, n \\
\frac{(n-1)n(n+1)(n+2)}{24} & \text{if } i = j = 0, n
\end{cases}$$
3.7 Other Related Topics for Gauss Quadratures

3.7.1 The Newton-Raphson Method

Some algebraic expressions in Gauss quadratures exhibit excessive sensitivities to even the smallest perturbations in the grid point locations especially for high-order quadratures [56]. We combine the eigenvalue method and the Newton-Raphson method to look for the exact locations for the grid points as close as possible. Generally, the accurate initial guesses are not easy find for the grid points in Gauss quadrature, which are needed for the Newton-Raphson method to work properly. Therefore, the grid points from the eigenvalue method will serve as the starting points for the Newton-Raphson method for further refinement.

**Stable Gauss-Hermite Quadrature**

In stable Gauss-Hermite quadrature, the grid points will be the roots of the Hermite function \(H_{n+1}(x)\) or modified Hermite function \(\mathcal{H}_{n+1}(x)\). In the Newton-Raphson method, if \(x_k\) is the \(k\)th guess of the root, then the next guess \(x_{k+1}\) will be,

\[
x_{k+1} = x_k - \frac{H_{n+1}(x_k)}{H'_n(x_k)}
= x_k - \frac{H_{n+1}(x_k)}{2(n + 1)H_n(x_k)}
= x_k - \frac{\mathcal{H}_{n+1}(x_k)}{\sqrt{2(n + 1)\mathcal{H}_n(x_k)}}
\]

(3.426)

**Stable Gauss-Radau-Laguerre Quadrature**

In stable Gauss-Radau-Laguerre quadrature, the grid points will be the roots of the Gauss-Radau-Laguerre polynomial \(\rho(x) = L_{n+1}(x) - L_n(x)\). In the Newton-Raphson method, if \(x_k\) is the \(k\)th guess of the root, then the next guess \(x_{k+1}\) will be,
\[ x_{k+1} = x_k - \frac{\rho_{n+1}(x_k)}{\rho'_{n+1}(x_k)} \]
\[ = x_k - \frac{L_{n+1}(x_k) - L_n(x_k)}{L'_{n+1}(x_k) - L'_n(x_k)} \]
\[ = x_k + \frac{L_{n+1}(x_k) - L_n(x_k)}{L_n(x_k)} \]
\[ = x_k + \frac{\mathcal{L}_{n+1}(x_k) - \mathcal{L}_n(x_k)}{\mathcal{L}_n(x_k)} \quad (3.427) \]

**Gauss-Lobatto-Legendre Quadrature**

In Gauss-Lobatto-Legendre quadrature, the grid points will be the roots of the Gauss-Lobatto-Legendre polynomial \( \rho(x) = P_{n+1}(x) - P_{n-1}(x) \). In the Newton-Raphson method, if \( x_k \) is the \( k \)th guess of the root, then the next guess \( x_{k+1} \) will be,

\[ x_{k+1} = x_k - \frac{\rho_{n+1}(x_k)}{\rho'_{n+1}(x_k)} \]
\[ = x_k - \frac{P_{n+1}(x_k) - P_{n-1}(x_k)}{P'_{n+1}(x_k) - P'_{n-1}(x_k)} \]
\[ = x_k - \frac{P_{n+1}(x_k) - P_{n-1}(x_k)}{(2n+1)P_n(x_k)} \quad (3.428) \]

### 3.7.2 Scaling Factors and Domain Transformations

In stable Gauss-Hermite quadrature, usually the integrand function has quadratic exponential decay property and so is negligible outside of a finite region of support, \([-M, M]\), where \( M > 0 \). When \( n \) is large, \( M \) is usually smaller than \( x_n \), the largest grid point associated with \( \mathcal{H}_{n+1}(x) \). For example, when \( n = 63 \), \( x_0 = -10.5261 \), and \( x_{63} = 10.5261 \), while the ordinal function only has meaningful values between \([-5, 5]\), about half number of grid points outside of the region will be wasted and hence it
introduces poor resolution. In order to efficiently utilize those points and increase computational accuracy, a scaling factor, \( \alpha \), was introduced in [57] as follows,

\[
\alpha = x_n/M
\] (3.429)

Let \( f(\xi) \) be the original function,\(^2\) then,

\[
\xi = x/\alpha
\] (3.430)

where \( x \) is the coordinate in the stable Gauss-Hermite quadrature.

Therefore, the basis expansion can be written as,

\[
f(\xi) \approx \sum_{i=0}^{n} \psi_i(x)f(\xi_i) = \sum_{i=0}^{n} \psi_i(\alpha\xi)f(x_i/\alpha)
\] (3.431)

where \( \xi_i \) is the mapped grid point corresponding to the original grid point \( x_i \).

The first derivative, \( f'(\xi) \), can be expanded by the same basis functions as follows,

\[
f'(\xi) \approx \sum_{i=0}^{n} \psi_i(x)f'(\xi_i) = \sum_{i=0}^{n} \psi_i(\alpha\xi)f'(x_i/\alpha)
\] (3.432)

with the first derivatives at the grid points,

\[
f'(\xi_i) = f'(x_i/\alpha) \approx \alpha \sum_{j=0}^{n} \psi_j(x_i)f(x_j/\alpha)
\] (3.433)

Similarly, the second derivative, \( f''(\xi) \), can be expanded as follows,

\[
f''(\xi) \approx \sum_{i=0}^{n} \psi_i(x)f''(\xi_i) = \sum_{i=0}^{n} \psi_i(\alpha\xi)f''(x_i/\alpha)
\] (3.434)

with the second derivatives at the grid points,

\[
f''(\xi_i) = f''(x_i/\alpha) \approx \alpha^2 \sum_{j=0}^{n} \psi_j(x_i)f(x_j/\alpha)
\] (3.435)

\(^2\) Here, we will use \( f(\xi) \) instead of \( f(x) \) to avoid notation confusion.
Likewise, for the stable Gauss-Radau-Laguerre quadrature, $f(\xi)$ usually only has a finite region of interest, $[0, M]$, where $M > 0$. When $n$ is large, $M$ is usually smaller than $x_n$, the largest grid point associated with $\rho(x) = \mathcal{L}_{n+1}(x) - \mathcal{L}_n(x)$. For example, when $n = 63$, $x_{63} = 232.8680$, while $f(\xi)$ probably only has meaningful values up to a maximum point much less than $x_{63}$. Therefore, the exact same scaling technique can be applied for the stable GRL quadrature related numerical methods.

On the other hand, the Gauss-Lobatto-Legendre quadratures has a definite domain of $[-1, 1]$. A region of $[a, b]$, which $f(\xi)$ is defined on, has to be transformed to $[-1, 1]$ before applying GLL related methods. The mapping from the original domain $[a, b]$ to the reference domain $[-1, 1]$ is as follows,

$$x = 2 \frac{\xi - a}{b - a} - 1 \quad (3.436)$$

where,

$$\xi \in [a, b], \quad x \in [-1, 1] \quad (3.437)$$

The Jacobian $J$ of the transformation is,

$$J = \frac{d\xi}{dx} = \frac{b - a}{2} \quad (3.438)$$

therefore, $\xi$ in the original domain can be rewritten as,

$$\xi = J(x + 1) + a \quad (3.439)$$

Similarly, any infinite domains and closed semi-infinite domains can be linearly transformed (shifted and scaled) into the reference domain $(-\infty, \infty)$ and $[0, \infty)$. The scaling for the stable Gauss-Hermite and stable Gauss-Radau-Laguerre quadrature is actually a particular kind of domain transformation,

$$J = \frac{1}{\alpha}, \quad \xi = Jx \quad (3.440)$$

107
The computations for first and second derivatives of the function \( f(\xi) \) using basis function expansion for all the three quadratures can be generalized as one framework with the Jacobian of transformation in the following section.

### 3.7.3 Computations of First and Second Derivatives

Suppose after a linear transformation, \( \xi \) in an original domain will be expressed with \( x \) in a reference domain, the Jacobian \( J \) and a constant \( c \), as follows,

\[
\xi = Jx + c \tag{3.441}
\]

Then, the approximation of \( f(\xi) \) by quadrature basis function expansion is,

\[
f(\xi) \approx \sum_{i=0}^{n} \phi_i(x) f(\xi_i) = \sum_{i=0}^{n} \phi_i \left( \frac{\xi - c}{J} \right) f(Jx_i + c) \tag{3.442}
\]

where \( \xi_i \) is the mapped grid point corresponding to the original grid point \( x_i \) and \( \{\phi_i(x), i = 0, \ldots, n\} \) is the set of basis functions for expansion.

The first derivative, \( f'(\xi) \), can be expanded by the same basis functions as follows,

\[
f'(\xi) \approx \sum_{i=0}^{n} \phi_i(x) f'(\xi_i) = \sum_{i=0}^{n} \phi_i \left( \frac{\xi - c}{J} \right) f'(Jx_i + c) \tag{3.443}
\]

with the first derivatives at the grid points,

\[
f'(\xi_i) = f'(Jx_i + c) \approx \frac{1}{J} \sum_{j=0}^{n} \phi_j'(x_i) f(Jx_j + c) \tag{3.444}
\]

Similarly, the second derivative, \( f''(\xi) \), can be expanded as follows,

\[
f''(\xi) \approx \sum_{i=0}^{n} \phi_i(x) f''(\xi_i) = \sum_{i=0}^{n} \phi_i \left( \frac{\xi - c}{J} \right) f''(Jx_i + c) \tag{3.445}
\]

with the second derivatives at the grid points,

\[
f''(\xi_i) = f''(Jx_i + c) \approx \frac{1}{J^2} \sum_{j=0}^{n} \phi_j''(x_i) f(Jx_j + c) \tag{3.446}
\]
3.8 Numerical Results for Gauss Quadrature Method

For the numerical experiments, we will use the following function $f(x)$,

$$f(x) = e^{-x^2/2} \cos(8x)$$  \hspace{1cm} (3.447)

The analytical first derivative of $f(x)$ is easy to derived as follows,

$$f'(x) = e^{-x^2/2} [-x \cos(8x) - 8 \sin(8x)]$$  \hspace{1cm} (3.448)

Similarly, the analytical second derivative of $f(x)$ can be written as,

$$f''(x) = e^{-x^2/2} [(x^2 - 65) \cos(8x) + 16x \sin(8x)]$$  \hspace{1cm} (3.449)

*Stable Gauss-Hermite Quadrature*

Using stable Gauss-Hermite quadrature, the support region is set wide as $[-8, 8]$, i.e., $M = 8$, and let the basis function order $n = 63$. Using (3.431), the interpolated values are plotted with true values in Figure 3.7. The $L_2$-Error between the analytical values and approximated values is $1.6387 \times 10^{-12}$.

![Figure 3.7: $f(x)$ and its approximates expanded by stable Gauss-Hermite bases.](image-url)
Using (3.432), the first derivative values approximated by the basis functions with the analytical values are illustrated in Figure 3.8, with the $L_2$-Error = $2.1882 \times 10^{-12}$.

![Figure 3.8: $f'(x)$ and its approximated values using stable Gauss-Hermite bases.](image)

Using (3.434), the second derivatives approximated by the basis functions with the analytical values are presented in Figure 3.9, with the $L_2$-Error = $5.4741 \times 10^{-12}$.

![Figure 3.9: $f''(x)$ and its approximated values using stable Gauss-Hermite bases.](image)
The precise locations of grid points are crucial for subsequent quadrature related computations for high accuracy, which is particularly true for high-order cases [56]. With the combination of the eigenvalue method and the Newton-Raphson method, the grid points are indeed more accurate and the errors of basis function expansions are reduced for high-order basis functions. Numerical results for comparisons between the combination method and the eigenvalue method alone using stable Gauss-Hermite quadrature are shown in Figure 3.10 – 3.12.

Figure 3.10 shows the comparison of $L_2$-Errors of the interpolation of $f(x)$ with respect to the stable Gauss-Hermite basis function order $n$ for the eigenvalue method and the combination method. Likewise, Figure 3.11 and Figure 3.12 illustrate the errors for the stable Gauss-Hermite basis function expansions of the first derivative $f'(x)$ and the second derivative $f''(x)$.

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure3_10.png}
\caption{Comparison between the eigenvalue method alone and the combination method for the function $f(x)$ using stable Gauss-Hermite basis expansion.}
\end{figure}
Figure 3.11: Comparison between the eigenvalue method and combination method for the first derivative $f'(x)$ using stable Gauss-Hermite basis function expansion.

Figure 3.12: Comparison between the eigenvalue method and combination method for the second derivative $f''(x)$ using stable Gauss-Hermite basis function expansion.
**Stable Gauss-Radau-Laguerre Quadrature**

The support region is set wide as $[0, 8]$, i.e., $M = 8$, and let the basis function order $n = 95$. The interpolated values are plotted with true values in Figure 3.13. The $L_2$-Error between the analytical values and approximated values is $2.0455 \times 10^{-13}$.

![Figure 3.13: $f(x)$ and its approximates expanded by stable GRL basis functions.](image)

The first derivative values approximated by the basis functions with the analytical values are illustrated in Figure 3.14, with the $L_2$-Error $= 1.0002 \times 10^{-11}$.

![Figure 3.14: $f'(x)$ and its approximated values using stable GRL basis functions.](image)
The second derivatives approximated by the basis functions with the analytical values are presented in Figure 3.15, with the $L_2$-Error = $1.4018 \times 10^{-9}$.

![Figure 3.15: $f''(x)$ and its approximated values using stable GRL basis functions.](image)

Figure 3.15: $f''(x)$ and its approximated values using stable GRL basis functions.

Figure 3.16 shows the $L_2$-Errors of the interpolation of $f(x)$ with respect to the GRL basis function order $n$ for the eigenvalue method and the combination method.

![Figure 3.16: Comparison between the eigenvalue method alone and the combination method for the function $f(x)$ using stable GRL basis function expansion.](image)

Figure 3.16: Comparison between the eigenvalue method alone and the combination method for the function $f(x)$ using stable GRL basis function expansion.
Gauss-Lobatto-Legendre Quadrature

Suppose \( f(x) \) is defined on \([-4, 4]\), and let the basis order \( n = 77 \). The interpolated values are plotted with true values in Figure 3.17. The \( L_2 \)-Error is \( 4.1908 \times 10^{-14} \).

![Figure 3.17: \( f(x) \) and its approximates expanded by GLL basis functions.](image)

The first derivative values approximated by the basis functions with the analytical values are illustrated in Figure 3.18, with the \( L_2 \)-Error = \( 1.3819 \times 10^{-13} \).

![Figure 3.18: \( f'(x) \) and its approximated values using GLL basis functions.](image)
The second derivatives approximated by the basis functions with the analytical values are presented in Figure 3.19, with the $L_2$-Error = $5.5524 \times 10^{-12}$.

Figure 3.19: $f''(x)$ and its approximated values using GLL basis functions.

Figure 3.20 shows the $L_2$-Errors of the interpolation of $f(x)$ with respect to the GLL basis function order $n$ for the eigenvalue method and the combination method.

Figure 3.20: Comparison between the eigenvalue method alone and the combination method for the function $f(x)$ using Gauss-Lobatto-Legendre basis expansion.
4

Introduction to Spectral Element Method

4.1 Historical Backgrounds

Spectral element method (SEM) is one of four major numerical techniques for the solutions of differential equations, which was pioneered by Anthony T. Patera in 1984 in [58]. Those major methodologies are roughly in the order of following decades, though the origins of each method can be further traced back [28],

1950s: finite difference methods
1960s: finite element methods
1970s: spectral methods
1980s: spectral element methods

Spectral methods were originally developed by meteorologists in the 1950s [59]. But it was not until the 1970s, spectral methods were used in practice for problems in fluid dynamics and meteorology by the advent of transform methods [60]. Classic spectral methods for smooth functions were well researched in both theories and algorithms by the late 1980s. Extending spectral methods to non-smooth complex geometries is one major research area in numerical methods for the past decades [61].
4.2 Modern Approaches

Due to the complexity of certain partial differential equations (PDEs) and their initial and boundary conditions, finding a fast and accurate solution of those complicated partial differential equations presents a paramount challenge. The relatively new partial differential equation solving technique, the spectral element method (SEM), combines the accuracy of the conventional spectral method and the geometric flexibility of the finite element method [25, 26].

The spectral element method has an excellent error decay property, so-called “exponential convergence”. When the accuracy of the solution is crucial, the spectral element method (SEM) provides a viable alternative to traditional methods like the finite difference method (FDM) and the finite element method (FEM), where the error converges algebraically in general. The spectral element method often achieves ten digits of accuracy where the finite difference method or finite element method would probably get much less [28]. At the same accuracy, the spectral element method (SEM) demands less computer memory and faster than the alternative finite difference method (FDM) and the finite element method (FEM) [28].

Spectral methods, which work well for globally smooth functions, have difficulties for problems in complex geometries, like problems with coordinate singularities, problems with discontinuous coefficients or solutions [62]. On the other hand, in the spectral element method (SEM), the underlying domain is generally separated into a number of elements according to the domain properties, like continuities (but non-smooth) and discontinuities. The solution in each local element is expanded by high-order orthogonal basis functions with associated Gauss quadrature rules. The schemes are “infinite-order” accurate if the expansion functions are properly chosen [58]. The *multi-element* spectral element method (ME-SEM) is named to differentiate the *one-element* spectral element method in this dissertation.
The spectral element method (SEM) is essentially a special discretization method for the approximate solution of a partial differential equation (PDE) expressed in a weak (variational) form [62]. In the spectral element method, the Galerkin approach is usually implemented, which enjoys the nice feature of the same basis function as trial function and test function [60]. In the middle 1980s, the Galerkin approach with Gaussian quadrature came into common use, which is referred as the Galerkin with numerical integration method, or Galerkin-NI method [61]. The corresponding spectral element method with numerical integration technique is named as SEM-NI method [61]. The spectral element method with Gauss quadrature, i.e., SEM-NI method, is implemented throughout this dissertation.

4.3 Simple One-Element Example

In the spectral element method (SEM), the entire domain is usually separated into several elements according to the complexity of the original solution. Within each element, a spectral representation is expressed based on a set of high-order Lagrange interpolation polynomials (basis functions). Each element could have a different set of basis functions with a different order. A discretization based on a weak (variational) formulation is commonly implemented in the spectral element method, where the continuity of adjacent elements and structure of the discretization enforce the continuity of the solution at the internal subdomain boundaries [62]. The numerical integration (NI) technique is generally considered an inherent part of the spectral element method (SEM), i.e., SEM-NI, as introduced in the original paper [58].

A Gauss-Lobatto-Legendre (GLL) quadrature is utilized in the following simple one-element example, mutli-element cases will be further explored in the subsequent chapters. Consider the following one-dimensional problem,
\[
\begin{align*}
\begin{cases}
u''(x) = f(x), & x \in [-1, 1] \\
u(-1) = u(1) = 0
\end{cases}
\end{align*}
\] (4.1)

where \( u(x) \) and \( f(x) \) are smooth in the support of \([-1, 1]\), \( f(x) \) is known and the objective is to solve \( u(x) \) using the spectral element method (SEM). The weak (variational) formulation of the above differential equation is as follows,

\[
\int_{-1}^{1} f(x)v(x)dx = \int_{-1}^{1} u''(x)v(x)dx \\
= u'(x)v(x)\big|_{-1}^{1} - \int_{-1}^{1} u'(x)v'(x)dx
\] (4.2)

where \( v(x) \), a test function, is any smooth function defined on \([-1, 1]\) that satisfies the certain boundary conditions. It can be shown there exists a unique solution \( u(x) \) in (4.2), which is indeed the original solution in (4.1) \[60\]. If the term \( u'(x)v(x) \) vanishes at the boundaries of the support, \([-1, 1]\), then the weak form becomes,

\[
\int_{-1}^{1} u'(x)v'(x)dx = - \int_{-1}^{1} f(x)v(x)dx
\] (4.3)

To discretize the problem, a set of basis functions \( \{v_i(x), \; i = 0, 1, \ldots, n\} \) is chosen to expand the original function, which serve as trial functions in the SEM. Thus \( u(x) \) and \( f(x) \) can be expanded by the trial functions as follows,

\[
u(x) = \sum_{i=0}^{n} u_i v_i(x), \quad f(x) = \sum_{i=0}^{n} f_i v_i(x)
\] (4.4)

In Galerkin method, the test functions are the same as the trial functions, then,

\[
\int_{-1}^{1} \sum_{i=0}^{n} u_i v'_i(x)v'_j(x)dx = - \int_{-1}^{1} \sum_{i=0}^{n} f_i v_i(x)v_j(x)dx
\] (4.5)
where \( j = 0, 1, \ldots, n \). For Galerkin method with numerical integration (Galerkin-NI), \( u_i \) and \( f_i \) are usually the function values at the grid points for a particular quadrature. That is, \( v_i(x) \) is the Lagrange interpolation polynomial with the grid points as the interpolating points, and the original functions are interpolated by those interpolation polynomials (basis functions). Hence, \( u(x) \) and \( f(x) \) can be discretized at the grid points with vector forms, \( \mathbf{u}, \mathbf{f} \), as follows,

\[
\mathbf{f} = [f_0, f_1, \cdots, f_n]', \quad \mathbf{u} = [u_0, u_1, \cdots, u_n]'
\]  

(4.6)

The weak form can be rearranged as follows,

\[
\sum_{i=0}^{n} u_i \int_{-1}^{1} v'_i(x)v'_j(x)dx = -\sum_{i=0}^{n} f_i \int_{-1}^{1} v_i(x)v_j(x)dx
\]  

(4.7)

where the above two integrations from the weak formulation can be written as the forms of mass matrix, \( \mathbf{M} \), and stiffness matrix, \( \mathbf{S} \), as follows,

\[
M_{ji} = \int_{-1}^{1} v_i(x)v_j(x)dx, \quad S_{ji} = \int_{-1}^{1} v'_i(x)v'_j(x)dx
\]  

(4.8)

Then, the weak form can be expressed in the matrix form as follows,

\[
\mathbf{S}\mathbf{u} = -\mathbf{M}\mathbf{f}
\]  

(4.9)

Because of the special characteristics of mass and stiffness matrices, which are usually diagonal or sparse, there are plenty of numerical linear algebra techniques to compute \( \mathbf{u} \) and \( u(x) \) efficiently, though it is not the main research topic of this dissertation. Precisely, in the methods proposed in this dissertation, the mass matrix is diagonal and the stiffness matrix is sparse for multi-element cases.
In common practices, high-order interpolation polynomials based on orthogonal polynomials will be used as trial and test functions, i.e. the Galerkin approach. For example, when Gauss-Lobatto-Legendre basis function, $\phi(x)$, is implemented as the trial and test function, then the mass matrix becomes,

$$M_{ji} = \int_{-1}^{1} \phi_i(x)\phi_j(x)dx = \sum_{i=0}^{n} w_k\phi_i(x_k)\phi_j(x_k) = w_i\delta_{ij} \quad (4.10)$$

and the stiffness matrix becomes,

$$S_{ji} = \int_{-1}^{1} \frac{\partial \phi_i(x)}{\partial x} \frac{\partial \phi_j(x)}{\partial x} dx = \sum_{k=0}^{n} w_k \frac{\partial \phi_i(x_k)}{\partial x} \frac{\partial \phi_j(x_k)}{\partial x} = \sum_{k=0}^{n} w_k D_{ki}D_{kj} \quad (4.11)$$

The discretized $u_i$ and $f_i$ are indeed the function values at grid point $x_i$,

$$u(x_i) = \sum_{k=0}^{n} u_k \phi_k(x_i) = \sum_{k=0}^{n} u_k \delta_{ki} = u_i, \quad i = 0, 1, \cdots, n \quad (4.12)$$

$$f(x_i) = \sum_{k=0}^{n} f_k \phi_k(x_i) = \sum_{k=0}^{n} f_k \delta_{ki} = f_i, \quad i = 0, 1, \cdots, n \quad (4.13)$$

With the computed $\mathbf{u}$, the $u(x)$ at arbitrary points can be interpolated as follows,

$$u(x) = \sum_{k=0}^{n} u_k \phi_k(x) \quad (4.14)$$

the first and second derivatives of $u(x)$ can be computed as in Section 3.7.3.

As discussed, the multi-element approach will be explored in the real examples in the next chapters. The main difference between the one-element and multi-element method is that the multi-element approach needs to enforce boundary conditions between elements correctly, with either continuities or discontinuities.
4.4 Error and Convergence Analysis

In the spectral element method (SEM), the error converges algebraically with the number of elements, \( m \), and exponentially with the polynomial order, \( n \), which are called \( h\)-refinement (element size) and \( p\)-refinement (polynomial order). If the original function to solve evolves with time, the Crank-Nicolson method for time-integration is usually employed, which is the method implemented in this dissertation. In that case, theoretically, the error has a second-order convergence with the time step while other parameters are kept constant.

Heuristically, the error approximately then equals to [62],

\[
\text{Convergence Error} \approx c_1 \frac{e^{-\alpha n}}{m^\gamma} + c_2 \delta t^\beta \approx c_1 \frac{e^{-\alpha n}}{m^{n+1}} + c_2 \delta t^2 \quad (4.15)
\]

where \( c_1 \) and \( c_2 \) are some constants, \( \delta t \) is the size of the time step. In (4.15) above, the natural logarithm base \( e \) is used, but it can be other bases too. For example, the base of 10 is more frequently implemented when the logarithm error is looked into in practice. Assuming other parameters are kept constant, the variable factors will affect the convergence error individually as follows,

\[
\Delta \log (\text{Error}) \approx -\alpha \times \Delta n \quad (4.16)
\]
\[
\Delta \log (\text{Error}) \approx \beta \times \Delta \log \delta t \approx 2 \times \Delta \log \delta t \quad (4.17)
\]
\[
\Delta \log (\text{Error}) \approx -\gamma \times \Delta \log m \approx -(n + 1) \times \Delta \log m \quad (4.18)
\]

where the actual estimates of \( \alpha \), \( \beta \) and \( \gamma \) will depend on other numerical factors, actual algorithm implementations, accuracy of boundary conditions, computing environments, so they do not have to be extremely close to the theoretical values.
Normally, for the same problem and beyond the required number of elements, the load of computational work for the spectral solutions depends on the polynomial order more than the number of elements, that is, the computation cost will likely be smaller if the total number of grid points is fixed while the number of elements is increased, while the error, on the other hand, will probably be larger [62]. Therefore, the work load and the accuracy need to be balanced in practice [62]. More technical details of the convergence and error analysis and the required work are discussed in [61]. Throughtout this dissertation, only the requirement number of elements will be studied because of the exponential convergence of $p$-refinement, which is the main focus of the spectral element method (SEM) here. The lesser number of elements will be illustrated only once for the accuracy comparison for the first real example. All the errors in this dissertation are relative errors unless otherwise specified.
Spectral Element Method on Black-Scholes

In this chapter, the spectral element method (SEM) will be applied to a standard European put option. Both one-element and multi-element method will be implemented and compared. The option final payoff is continuous but not smooth. In other words, the option price is continuous at maturity, but the derivative of the option price, i.e., Delta is not continuous at expiration. Therefore, it’s natural to use the multi-element SEM (ME-SEM) and the boundary should be set at the strike price. The numerical results do show that the multi-element method significantly increases the accuracy and reduces computational time. A standard European call option can be computed direct from the put-call parity. Option Greeks like Delta, Gamma and Theta are directly computed from the spectral element method. The popular finite difference method (FDM) and finite element method (FEM) are compared with this method. The comparison results indicate the spectral element method is an alternative efficient algorithm both in accuracy and time cost. The spectral element method can be applied to options with more complicated payoffs. As an example, a condor option shows an excellent result under this technique.
5.1 One-Element SEM on Black-Scholes PDE

5.1.1 Weak Formulation

A standard European put option price will follow the following PDE,

\[
\frac{1}{2} \sigma^2 S^2(t) P_{SS} + r S(t) P_S + P_t - r P = 0
\]  

(5.1)

where \( t \geq 0, S \geq 0, r \) and \( \sigma \) are constant, and the final condition is,

\[
P(S(T), T) = \max\{K - S(T), 0\}
\]  

(5.2)

Then, the weak formulation of the partial differential equation is,

\[
\int_0^\infty \left[ \frac{1}{2} \sigma^2 S^2(t) P_{SS} + r S(t) P_S + P_t - r P \right] v(S) dS = 0
\]  

(5.3)

In theory, the stock price is not prevented from going to infinity. Moreover, the integration of the entire domain has better accuracy and is easier to deal with the boundary conditions comparing with the method of domain truncation.

Using integration by parts, the integration of the diffusion term (the second derivative term) in the weak formulation (5.3) becomes,

\[
\int_0^\infty \frac{1}{2} \sigma^2 S^2(t) P_{SS} v(S) dS
\]

\[
= \frac{1}{2} \sigma^2 \left[ S^2 v(S) P_S \right]_0^\infty - \frac{1}{2} \sigma^2 \int_0^\infty P_S d \left[ S^2 v(S) \right]
\]

\[
= \frac{1}{2} \sigma^2 S^2 v(S) P_S \bigg|_0^\infty - \sigma^2 \int_0^\infty P_S v(S) dS - \frac{1}{2} \sigma^2 \int_0^\infty P_S S^2 v_S dS
\]  

(5.4)

Then, the weak formulation (5.3) can be rewritten as follows,

\[
\frac{1}{2} \sigma^2 S^2 v(S) P_S \bigg|_0^\infty + (r - \sigma^2) \int_0^\infty P_S v(S) dS
\]

\[
- \frac{1}{2} \sigma^2 \int_0^\infty P_S S^2 v_S dS + \int_0^\infty P_t v(S) dS - r \int_0^\infty P(S) v(S) dS = 0
\]  

(5.5)
For the one-element method, stable Gauss-Radau-Laguerre (GRL) basis functions will be chosen here as both trial functions and test functions. Though the domain of GRL basis functions matches exactly the domain of stock price, \([0, \infty)\), a scaling factor is needed to increase the computational accuracy. For the purpose of consistency, a Jacobian will be used instead of the scaling factor. That is, the stock price domain \([0, \infty)\) is affine mapped to the reference domain \([0, \infty)\) as follows,

\[
S(x) = Jx, \quad x \in [0, \infty), \quad S \in [0, \infty)
\]  

(5.6)

where \(J\) here can set as, \(S_{\text{max}}/x_n\), \(S_{\text{max}}\) is the stock price where the put option price is deep out-of-money, i.e., \(P \approx 0\), and \(x_n\) is the largest grid point for the \(n\)th-order stable GRL basis function. After the transformation, the weak formulation becomes,

\[
\frac{1}{2} \sigma^2 S(x)^2 v(x) \left( \frac{P_x}{J} \right) \bigg|_0^\infty + (r - \sigma^2) \int_0^\infty P(x) S(x) v(x) dx
\]

\[
- \frac{1}{2} \sigma^2 \int_0^\infty P_x S(x)^2 \frac{v_x}{J} dx + \int_0^\infty P_t v(x) J dx - r \int_0^\infty P(x) v(x) J dx = 0
\]  

(5.7)

Implemented so-called Galerkin-NI method, the put option price \(P\) is expanded by the stable GRL basis functions, which are used as test functions as well, i.e.,

\[
P(x) = \sum_{i=0}^n \phi_i(x) P(x_i)
\]  

(5.8)

\[
v(x) = \phi_j(x)
\]  

(5.9)

Applying all the test functions onto the weak formulation, there will be a set of different weak formulations, each of which is corresponding to one test function. If \(n\)th-order basis functions are used, there are \(n + 1\) test functions, hence \(n + 1\) simultaneous equations. Then the weak formulation can be rewritten as follows,

---

\(1\) For unifying notations, \(w_i, \phi_i(x), D_{ij}, B_{ij}\) will be referred as stable ones as well here.
\[
\frac{1}{2} \sigma^2 S(x)^2 \phi_j(x) \left. \frac{P_x}{J} \right|_0^\infty \\
+ (r - \sigma^2) \int_0^\infty \sum_{i=0}^n P(x_i) \frac{\partial \phi_i(x)}{\partial x} S(x) \phi_j(x) dx \\
- \frac{1}{2} \sigma^2 \int_0^\infty \sum_{i=0}^n P(x_i) \frac{\partial \phi_i(x)}{\partial x} S(x)^2 \frac{\partial \phi_j(x)}{\partial x} \frac{1}{J} dx \\
+ \int_0^\infty \sum_{i=0}^n \phi_i(x) \frac{\partial P(x_i)}{\partial t} \phi_j(x) J dx \\
- r \int_0^\infty \sum_{i=0}^n \phi_i(x) P(x_i) \phi_j(x) J dx = 0 \quad (5.10)
\]

where \( \phi_j(x) \) exponentially decays when \( x \to \infty \) and \( S(x = 0) = 0 \), therefore the first term of the above equation vanishes for each \( j \). The boundary condition term disappears as mentioned without any prior knowledge of the option price or its Greeks at the boundaries. Then the above equations can be rearranged as follows,

\[
(r - \sigma^2) \sum_{i=0}^n P(x_i) \int_0^\infty S(x) \frac{\partial \phi_i(x)}{\partial x} \phi_j(x) dx \\
- \frac{1}{2J} \sigma^2 \sum_{i=0}^n P(x_i) \int_0^\infty S(x)^2 \frac{\partial \phi_i(x)}{\partial x} \frac{\partial \phi_j(x)}{\partial x} dx \\
+ J \sum_{i=0}^n \frac{\partial P(x_i)}{\partial t} \int_0^\infty \phi_i(x) \phi_j(x) dx \\
- rJ \sum_{i=0}^n P(x_i) \int_0^\infty \phi_i(x) \phi_j(x) dx = 0 \quad (5.11)
\]

The integration terms in the above equation can be computed efficiently using the associated stable GRL quadrature. Those integrations will be very easy to compute as long as the accurate weights and spectral differential matrix are available. The equation also can be rearranged by separating the time integration term as follows,
\[ J \sum_{i=0}^{n} \frac{\partial P(x_i)}{\partial t} \int_{0}^{\infty} \phi_i(x) \phi_j(x) dx \]

\[ = -(r - \sigma^2) \sum_{i=0}^{n} P(x_i) \int_{0}^{\infty} S(x) \frac{\partial \phi_i(x)}{\partial x} \phi_j(x) dx \]

\[ + \frac{1}{2} J^2 \sigma^2 \sum_{i=0}^{n} P(x_i) \int_{0}^{\infty} S(x)^2 \frac{\partial \phi_i(x) \partial \phi_j(x)}{\partial x} dx \]

\[ + r J \sum_{i=0}^{n} P(x_i) \int_{0}^{\infty} \phi_i(x) \phi_j(x) dx \] (5.12)

For the above equation, the mass and stiffness matrices can be written as follows,

\[ M_{ji} = J \int_{0}^{\infty} \phi_i(x) \phi_j(x) dx \] (5.13)

\[ S_{ji} = -(r - \sigma^2) \int_{0}^{\infty} S(x) \frac{\partial \phi_i(x)}{\partial x} \phi_j(x) dx \]

\[ + \frac{1}{2} J^2 \sigma^2 \int_{0}^{\infty} S(x)^2 \frac{\partial \phi_i(x) \partial \phi_j(x)}{\partial x} dx \]

\[ + r J \int_{0}^{\infty} \phi_i(x) \phi_j(x) dx \] (5.14)

The integration terms can be computed using stable GRL quadrature as follows,

\[ \int_{0}^{\infty} \phi_i(x) \phi_j(x) dx = \sum_{k=0}^{n} w_k \phi_i(x_k) \phi_j(x_k) = w_i \delta_{ij} \] (5.15)

\[ \int_{0}^{\infty} S(x) \frac{\partial \phi_i(x)}{\partial x} \phi_j(x) dx = \sum_{k=0}^{n} w_k S(x_k) \frac{\partial \phi_i(x_k)}{\partial x} \phi_j(x_k) \]

\[ = w_j D_{ji} S(x_j) \] (5.16)

\[ \int_{0}^{\infty} S(x)^2 \frac{\partial \phi_i(x)}{\partial x} \frac{\partial \phi_j(x)}{\partial x} dx = \sum_{k=0}^{n} w_k S(x_k)^2 \frac{\partial \phi_i(x_k)}{\partial x} \frac{\partial \phi_j(x_k)}{\partial x} \]

\[ = \sum_{k=0}^{n} w_k D_{ki} D_{kj} S(x_k)^2 \] (5.17)
The put option price can be discretized at grid points, which are both interpolation points for the basis functions and quadrature points for the Gauss quadrature.

\[
P = \begin{bmatrix} P_0, P_1, \cdots, P_n \end{bmatrix}' = \begin{bmatrix} P(x_0), P(x_1), \cdots, P(x_n) \end{bmatrix}'
\]

Therefore, the weak formulation can be written as a matrix equation as follows,

\[
M \frac{\partial P}{\partial t} = SP
\]

where the mass matrix, \( M \), is diagonal because of the unique property of the basis functions, while the stiffness matrix here, \( S \), is generally neither diagonal nor sparse for the one-element spectral element method.

### 5.1.2 Time Discretization

The Crank-Nicolson time discretization method will be implemented here as follows,

\[
M \frac{P^{k+1} - P^k}{\delta t} = S \frac{P^{k+1} + P^k}{2}
\]

therefore, the put option price can work backwards from \( t_{k+1} \) to \( t_k \) as follows,

\[
P^k = \left( I + \frac{\delta t}{2} M^{-1} S \right)^{-1} \left( I - \frac{\delta t}{2} M^{-1} S \right) P^{k+1}
\]

where \( \delta t = t_{k+1} - t_k \), and the put option final payoff \( P(T) \) is as follows,

\[
P(T) = \max\{K - S(T), 0\}
\]

The initial put option price is \( P^0 \). As long as the put option price at the grid points are available, the put option price at any arbitrary points (arbitrary stock prices) can be interpolated (expanded) by the basis functions as follows,

\[
P(S) = P(S(x)) = P(x) = \sum_{i=0}^{n} \phi_i(x) P(x_i)
\]
5.1.3 SEM Option Greeks

Some of the option sensitivities to the underlying variables can be directly computed from the spectral element method (SEM), for example, Delta ($\Delta$), Gamma ($\Gamma$) and Theta ($\Theta$). As in the option pricing, the option Greeks at the grid points will first be computed, and the Greeks at any other points can be interpolated by the same basis functions. Those Greeks can be calculated as efficiently as the option price.

From the spectral element method (SEM), the put option price is,

$$P(S) = \sum_{i=0}^{n} \phi_i(x) P(x_i), \quad S = Jx$$

(5.24)

then, the option Delta , i.e, the first derivative with respect to the stock price is,

$$\Delta = \frac{\partial P(S)}{\partial S} = \frac{1}{J} \sum_{i=0}^{n} \frac{\partial \phi_i(x)}{\partial x} P(x_i) = \frac{1}{J} \sum_{i=0}^{n} \phi'_i(x) P(x_i)$$

(5.25)

therefore, the option Delta at the grid points can be easily derived as follows,

$$\Delta_i = \frac{1}{J} \sum_{j=0}^{n} \phi'_j(x_i) P(x_j) = \frac{1}{J} \sum_{j=0}^{n} D_{ij} P(x_j), \quad i = 0, 1, \ldots, n$$

(5.26)

Similarly, the option Gamma, the second derivative with respect to stock price is,

$$\Gamma = \frac{\partial^2 P(S)}{\partial S^2} = \frac{1}{J^2} \sum_{i=0}^{n} \frac{\partial^2 \phi_i(x)}{\partial x^2} P(x_i) = \frac{1}{J^2} \sum_{i=0}^{n} \phi''_i(x) P(x_i)$$

(5.27)

hence, the option Gamma at the grid points can be easily computed as follows,

$$\Gamma_i = \frac{1}{J^2} \sum_{j=0}^{n} \phi''_j(x_i) P(x_j) = \frac{1}{J^2} \sum_{j=0}^{n} B_{ij} P(x_j), \quad i = 0, 1, \ldots, n$$

(5.28)

The option Theta, i.e., the first derivative with respect to the passage of time,

$$\Theta = \frac{\partial P(S,t)}{\partial t} = \frac{\partial P}{\partial t}$$

(5.29)
can be directly computed from the matrix equation of the weak form as follows,

\[
\Theta = \frac{\partial P}{\partial t} = M^{-1}SP
\]  

(5.30)

without the need to refer to the original Black-Scholes partial differential equation.

5.1.4 Computation Result

The one-element spectral element method (SEM) using stable Gauss-Radau-Laguerre (GRL) quadrature and basis functions with Galerkin-NI approach is applied to a standard European put option with the following parameters,

- Strike Price: \( K = 40.0 \)
- Interest Rate: \( r = 5.0\% \)
- Stock Volatility: \( \sigma = 25\% \)
- Option Maturity: \( T = 0.5 \text{ yr} \)

For one example, the stable GRL basis functions with order of 60, i.e., the total number of grid points is 61, is implemented with a time step of about \( 10^{-5} \) year. The computed put option surface over time-to-maturity and possible traded stock price is illustrated in Figure 5.1. The comparison of the calculated put option price with the analytical price from the Black-Scholes closed-form solution is plotted in Figure 5.2. The errors for the current put option price, \( P(t = 0) \), are quiet impressive with \( L_2\)-Error = \( 7.4738 \times 10^{-5} \) and \( L_\infty\)-Error = \( 7.2058 \times 10^{-5} \).

The \( L_2\)-Errors with increasing number of total grid points are graphed in Figure 5.3, while the \( L_2\)-Errors with respect to the time step are depicted in Figure 5.4. Those results are consistent with convergence analysis in Section 4.4. There are quiet similar results for the \( L_\infty\)-Errors. The accuracy can be further improved using a multi-element method because of the non-smoothness of the final payoff.
Figure 5.1: The standard European put option prices over time-to-maturity and stock price using one-element spectral element method with stable GRL quadrature.

Figure 5.2: Comparison between analytical Black-Scholes prices and one-element spectral element method prices for the standard European put option at inception.
Figure 5.3: The $L_2$-Errors for the standard European put option prices with respect to the total number of grid points using one-element method with $\delta t \doteq 10^{-5}$ year.

Figure 5.4: The $L_2$-Errors for the standard European put option prices with respect to time step using the one-element method with the total number of grid points $= 61$. 
5.2 Multi-Element SEM on Black-Scholes PDE

5.2.1 Weak Formulation

Since the final payoffs at maturity for most options are not smooth, one-element approach has limited accuracy because of this non-smoothness. For example, a put option’s final payoff is continuous in the level of price, but discontinuous for the first derivative of the put option at strike. In order to further improve the accuracy, a multi-element approach will be implemented. In this approach, the whole domain will be divided into several subdomains (elements) and each element will be affine mapped to a reference domain. In every element, the function will be interpolated with a set of appropriate basis functions. Each subdomain does not have to be the same size and each element can use a different set of basis functions.

For a standard European put option, the price function is smooth between \([0, K]\) and \([K, \infty)\), where \(K\) is the strike, therefore theoretically only two elements are needed to achieve a higher accuracy. Not surprisingly, the whole domain will be cut into two elements according to the local smoothness, which are, \([0, K]\) and \([K, \infty)\).

As in the one-element method, Galerkin-NI method will be implemented for the multi-element approach as well, i.e., the trial function will be the same as the test function. Then, the weak formulation of the partial differential equation is,

\[
\int_{0}^{\infty} \left[ \frac{1}{2} \sigma^2 S^2 (t) P_{SS} + r S(t) P_{S} + P_t - r P \right] v(S) dS = 0 \tag{5.31}
\]

where \(v(S)\) can be considered as a global basis function consists of basis functions from each element. The structure of \(v(S)\) will be more clear later. As in one-element approach, the weak formulation can be rewritten as follows,

\[
\frac{1}{2} \sigma^2 S^2 v(S) P_{S} \bigg|_{0}^{\infty} + (r - \sigma^2) \int_{0}^{\infty} P_{S} S v(S) dS
\]

\[
- \frac{1}{2} \sigma^2 \int_{0}^{\infty} P_{S} S^2 v_S dS + \int_{0}^{\infty} P_{t} v(S) dS - r \int_{0}^{\infty} P(S) v(S) dS = 0 \tag{5.32}
\]
As discussed above, the whole domain of stock price, \([0, \infty)\), will be separated into two elements, \([0, K]\) and \([K, \infty)\). The first element will be expanded by the Gauss-Lobatto-Legendre (GLL) basis functions, and the second element will be interpolated with the stable Gauss-Radau-Laguerre (GRL) basis functions. Therefore, the first subdomain will be affine mapped to the reference domain, \([-1, 1]\) and the second one will be linearly transformed to the reference domain of \([0, \infty)\). The scaling of the second element is necessary for further accuracy improvement. The separations of the domain with the Jacobians are as follows,

\[
S^a = J^a x^a + J^a, \quad J^a = \frac{K}{2}, \quad x^a \in [-1, 1], \quad S^a \in [0, K]
\]

\[
S^b = J^b x^b + K, \quad J^b = \frac{S_{\text{max}} - K}{x^b_n}, \quad x^b \in [0, \infty), \quad S^b \in [K, \infty)
\]

where the parameters of the first element are denoted by the superscript \(a\) and the superscript \(b\) for the second element, and \(x^b_n\) is the largest grid point for the stable Gauss-Radau-Laguerre (GRL) basis function. The corresponding put option price \(P\) will be denoted as \(P^a\) and \(P^b\) in the separated two domains respectively. After the separation of domains, the weak formulation becomes,

\[
\frac{1}{2} \sigma^2 S^2 v(S) P_S \bigg|_0^\infty \\
+ (r - \sigma^2) \int_0^K P_S S v(S) dS \\
- \frac{1}{2} \sigma^2 \int_0^K P_S S^2 v_S dS + \int_0^K P_t v(S) dS - r \int_0^K P(S) v(S) dS \\
+ (r - \sigma^2) \int_K^\infty P_S S v(S) dS \\
- \frac{1}{2} \sigma^2 \int_K^\infty P_S S^2 v_S dS + \int_K^\infty P_t v(S) dS - r \int_K^\infty P(S) v(S) dS = 0
\]

(5.35)
The put prices at the two elements, $P^a$ and $P^b$, can be expanded by the GLL functions, $\phi^a(x)$, and the stable GRL basis functions, $\phi^b(x)$, respectively. The order of the two sets of the basis functions are $n^a$ and $n^b$. These expansions are,

$$P^a(S^a) = P^a(x^a) = \sum_{i=0}^{n^a} \phi^a_i(x^a)P^a(x^a_i) \quad (5.36)$$

$$P^b(S^b) = P^b(x^b) = \sum_{i=0}^{n^b} \phi^b_i(x^b)P^b(x^b_i) \quad (5.37)$$

where the last grid point of the first element is the first grid point of the second element, and the put prices at those two points are actually one put price, i.e.,

$$x^a_{n^a} = x^b_0, \quad P^a(x^a_{n^a}) = P^b(x^b_0) \quad (5.38)$$

There will be total $n^a + n^b + 1$ distinctive grid points and the put option price is continuous at the boundary of strike price $K$. Therefore, a set of $n^a + n^b + 1$ global basis functions can be constructed as follows,

$$v_0(S) = \begin{cases} 
\phi^a_0(x^a), & S \in [0, K] \\
0, & S \in [K, \infty) 
\end{cases} \quad (5.39)$$

$$v_1(S) = \begin{cases} 
\phi^a_1(x^a), & S \in [0, K] \\
0, & S \in [K, \infty) 
\end{cases} \quad (5.40)$$

$$\vdots$$

$$v_{n^a}(S) = \begin{cases} 
\phi^a_{n^a}(x^a), & S \in [0, K] \\
\phi^b_0(x^b), & S \in [K, \infty) 
\end{cases} \quad (5.41)$$

$$v_{n^a+1}(S) = \begin{cases} 
0, & S \in [0, K] \\
\phi^b_1(x^b), & S \in [K, \infty) 
\end{cases} \quad (5.42)$$

$$\vdots$$

$$v_{n^a+n^b}(S) = \begin{cases} 
0, & S \in [0, K] \\
\phi^b_{n^b}(x^b), & S \in [K, \infty) 
\end{cases} \quad (5.43)$$
where the global basis function $v_i(S)$ is a continuous function. The set of the basis function can be formulated as a basis function vector, $v(S)$, as follows,

$$v(S) = [v_0(S), v_1(S), \ldots, v_{n^a}(S), v_{n^a+1}(S), \ldots, v_{n^a+n^b}(S)]' \quad (5.44)$$

An illustrative example of a set of global basis functions is graphed in Figure 5.5. The local basis function in the first element is the GLL basis function with order $n^a = 3$ and the second one is the stable GRL basis function with order $n^b = 3$.

![Figure 5.5](image)

**Figure 5.5**: A multi-element basis function with the GLL basis function on the left with order $n^a = 3$ and the stable GRL basis function on the right with order $n^b = 3$.

The last local basis function in the first element is actually welded with the first local basis function of the second element, which enforces the continuity of the original function, i.e., the continuity of the put option price. After the construction of the global basis function, as a trial function, the put option price can expanded by the basis functions in a global formula as follows,

$$P(S) = \sum_{i=0}^{n^a+n^b} v_i(S)P(S_i) \quad (5.45)$$
where $S_i$ is the stock price corresponding to the $i^{th}$ grid point in global perspective. The global expansion is consistent with the local interpolation for each element,

$$P^a(x^a) = \sum_{i=0}^{n^a+n^b} v_i(S)P(S_i) = \sum_{i=0}^{n^a} v_i(S)P(S_i) = \sum_{i=0}^{n^a} \phi_i^a(x^a)P^a(x^a)$$ (5.46)

$$P^b(x^b) = \sum_{j=0}^{n^a+n^b} v_j(S)P(S_j) = \sum_{j=n^a}^{n^b} v_j(S)P(S_j) = \sum_{i=0}^{n^b} \phi_i^b(x^b)P^b(x^b)$$ (5.47)

Similar to the one-element approach, the same basis function $v(S)$ will be used as a test function in Galerkin-NI method. Because of this method, the exponential decay property will be preserved with relatively easy numerical integration. There are $n^a+n^b+1$ basis functions, so there will be $n^a+n^b+1$ simultaneous equations of weak formulations, each of which is correspond to one basis function. The boundary condition term in (5.35) disappears with same reasoning as in the one-element method. Then the weak formulation without the boundary condition term is as follows,

$$\left(r - \sigma^2\right) \int_0^K P_S S v(S) dS - \frac{1}{2} \sigma^2 \int_0^K P_S S^2 v_S dS$$

$$+ \int_0^K P_t v(S) dS - r \int_0^K P(S) v(S) dS +$$

$$\left(r - \sigma^2\right) \int_0^\infty P_S S v(S) dS - \frac{1}{2} \sigma^2 \int_0^\infty P_S S^2 v_S dS$$

$$+ \int_0^\infty P_t v(S) dS - r \int_0^\infty P(S) v(S) dS = 0$$ (5.48)

Recall that the affine mappings of the two subdomains are as follows,

$$S^a = J^a x^a + J^a, \quad J^a = \frac{K}{2}, \quad x^a \in [-1, 1], \quad S^a \in [0, K]$$ (5.49)

$$S^b = J^b x^b + K, \quad J^b = \frac{S_{\text{max}} - K}{x_n^b}, \quad x^b \in [0, \infty), \quad S^b \in [K, \infty)$$ (5.50)
Then, the weak formulation will be looked into separately element by element. The symbols, $E^a$ and $E^b$, will be denoted for the formulas of the first and second element respectively. In each element, the weak formulation will be conducted with local basis functions, later the weak forms will be consolidated back into a globe one.

$$E^a = (r - \sigma^2) \int_{-1}^{1} P_x S(x) \phi^a(x) dx - \frac{1}{2} \sigma^2 \int_{-1}^{1} P_x S(x)^2 J^a \phi^a_x dx$$

$$+ \int_{-1}^{1} P \phi^a(x) J^a dx - r \int_{-1}^{1} P(x) \phi^a(x) J^a dx \quad (5.51)$$

$$E^b = (r - \sigma^2) \int_{0}^{\infty} P_x S(x) \phi^b(x) dx - \frac{1}{2} \sigma^2 \int_{0}^{\infty} P_x S(x)^2 J^b \phi^b_x dx$$

$$+ \int_{0}^{\infty} P \phi^b(x) J^b dx - r \int_{0}^{\infty} P(x) \phi^b(x) J^b dx \quad (5.52)$$

where $E^a$ and $E^b$ are not simply additive to have the summation of zero. They will be combined together according to the structures of the global basis function as one defined in (5.44). Similar to the one-element case, after the put option prices at the two elements are expanded separately in its own subdomain, the weak formulations at the first element, $E^a$, and the second element, $E^b$, become,

$$E^a = (r - \sigma^2) \sum_{i=0}^{n^a} P^a(x_i^a) \int_{-1}^{1} S^a(x) \frac{\partial \phi^a_i(x)}{\partial x} \phi^a_j(x) dx$$

$$- \frac{1}{2} J^a \sigma^2 \sum_{i=0}^{n^a} P^a(x_i^a) \int_{-1}^{1} S^a(x)^2 \frac{\partial \phi^a_i(x)}{\partial x} \frac{\partial \phi^a_j(x)}{\partial x} dx$$

$$+ J^a \sum_{i=0}^{n^a} \frac{\partial P^a(x_i^a)}{\partial t} \int_{-1}^{1} \phi^a_i(x) \phi^a_j(x) dx$$

$$- r J^a \sum_{i=0}^{n^a} P^a(x_i^a) \int_{-1}^{1} \phi^a_i(x) \phi^a_j(x) dx \quad (5.53)$$
\[ E^b = (r - \sigma^2) \sum_{i=0}^{n^b} P^b(x_i^b) \int_0^{\infty} S^b(x) \frac{\partial \phi^b_i(x)}{\partial x} \phi^b_j(x) dx \]

\[ - \frac{1}{2} J^b \sigma^2 \sum_{i=0}^{n^b} P^b(x_i^b) \int_0^{\infty} S^b(x)^2 \frac{\partial \phi^b_i(x)}{\partial x} \frac{\partial \phi^b_j(x)}{\partial x} dx \]

\[ + J^b \sum_{i=0}^{n^b} \frac{\partial P^b(x_i^b)}{\partial t} \int_0^{\infty} \phi^b_i(x) \phi^b_j(x) dx \]

\[ - r J^b \sum_{i=0}^{n^b} P^b(x_i^b) \int_0^{\infty} \phi^b_i(x) \phi^b_j(x) dx \]  

(5.54)

Like those in one-element approach, the integration terms in the above formulas can be computed efficiently using the associated Gauss-type quadratures, the Gauss-Lobatto-Legendre (GLL) quadrature and the stable Gauss-Radau-Laguerre (GRL) quadrature. As long as the accurate weights and spectral differential matrix are available, those integrations can be computed easily as follows,

\[ \int \phi_i(x) \phi_j(x) dx = \sum_{k=0}^{n} w_k \phi_i(x_k) \phi_j(x_k) = w_i \delta_{ij} \]  

(5.55)

\[ \int S(x) \frac{\partial \phi_i(x)}{\partial x} \phi_j(x) dx = \sum_{k=0}^{n} w_k S(x_k) \frac{\partial \phi_i(x_k)}{\partial x} \phi_j(x_k) \]

\[ = w_j D_{ji} S(x_j) \]  

(5.56)

\[ \int S(x)^2 \frac{\partial \phi_i(x)}{\partial x} \frac{\partial \phi_j(x)}{\partial x} dx = \sum_{k=0}^{n} w_k S(x_k)^2 \frac{\partial \phi_i(x_k)}{\partial x} \frac{\partial \phi_j(x_k)}{\partial x} \]

\[ = \sum_{k=0}^{n} w_k D_{ki} D_{kj} S(x_k)^2 \]  

(5.57)

The above equations are good for the calculations in all elements. The nature of the above quadratures is easy for related equations to be written in the matrix forms.
From the weak formulas of $E^a$ and $E^b$, the local mass matrices, $M^a$ and $M^b$, and the local stiffness matrices, $S^a$ and $S^b$, for the two elements are defined as follows,

\[ M^a_{ji} = J^a \int_{-1}^{1} \phi^a_i(x) \phi^a_j(x) dx \] (5.58)

\[ S^a_{ji} = -(r - \sigma^2) \int_{-1}^{1} S^a(x) \frac{\partial \phi^a_i(x)}{\partial x} \phi^a_j(x) dx \]
\[ + \frac{1}{2J^a} \sigma^2 \int_{-1}^{1} S^a(x)^2 \frac{\partial \phi^a_i(x)}{\partial x} \frac{\partial \phi^a_j(x)}{\partial x} dx \]
\[ + rJ^a \int_{-1}^{1} \phi^a_i(x) \phi^a_j(x) dx \] (5.59)

\[ M^b_{ji} = J^b \int_{0}^{\infty} \phi^b_i(x) \phi^b_j(x) dx \] (5.60)

\[ S^b_{ji} = -(r - \sigma^2) \int_{0}^{\infty} S^b(x) \frac{\partial \phi^b_i(x)}{\partial x} \phi^b_j(x) dx \]
\[ + \frac{1}{2J^b} \sigma^2 \int_{0}^{\infty} S^b(x)^2 \frac{\partial \phi^b_i(x)}{\partial x} \frac{\partial \phi^b_j(x)}{\partial x} dx \]
\[ + rJ^b \int_{0}^{\infty} \phi^b_i(x) \phi^b_j(x) dx \] (5.61)

where the local mass matrices, $M^a$ and $M^b$, are diagonal because of the unique properties of the basis functions, while the stiffness matrices, $S^a$ and $S^b$, are generally neither diagonal nor sparse, though the global stiffness matrix is indeed sparse.

Therefore, the weak formulas of $E^a$ and $E^b$ for its own $j^{th}$ local test function, $\phi^a_j$ and $\phi^b_j$, denoted by, $E^a_j$ and $E^b_j$, can be written as follows,

\[ E^a_j = \sum_{i=0}^{n_a} \frac{\partial P^a(x^a_i)}{\partial t} M^a_{ji} - \sum_{i=0}^{n_a} P^a(x^a_i) S^a_{ji} \] (5.62)

\[ E^b_j = \sum_{i=0}^{n_b} \frac{\partial P^b(x^b_i)}{\partial t} M^b_{ji} - \sum_{i=0}^{n_b} P^b(x^b_i) S^b_{ji} \] (5.63)
The two parts of $E^a$ and $E^b$ are not simply additive because of the structure of the global basis function as defined in (5.44). $E^a$ and $E^b$ can only be formulated together after careful constructions of the global mass matrix, $M$, and the global stiff matrix, $S$, and the proper discretization of the put option price in the entire domain. The global mass matrix, $M$, is constructed as follows,

$$
M = \begin{bmatrix}
M_{00}^a & \cdots & M_{0n_a}^a \\
\vdots & \ddots & \vdots \\
M_{n_a0}^a & \cdots & M_{n_an_a}^a + M_{00}^b & \cdots & M_{0n_b}^b \\
\vdots & \ddots & \vdots \\
M_{n_b0}^b & \cdots & M_{n_bn_b}^b
\end{bmatrix}
$$

(5.64)

where $M^a$ and $M^b$ are diagonal, $M$ is diagonal as well. So $M$ can be written as,

$$
M = \begin{bmatrix}
M_{00}^a \\
\vdots \\
M_{n_an_a}^a + M_{00}^b \\
\vdots \\
M_{n_bn_b}^b
\end{bmatrix}
$$

(5.65)

Similarly, the global stiffness matrix, $S$, can be constructed as the following,

$$
S = \begin{bmatrix}
S_{00}^a & \cdots & S_{0n_a}^a \\
\vdots & \ddots & \vdots \\
S_{n_an_a}^a + S_{00}^b & \cdots & S_{0n_b}^b \\
\vdots & \ddots & \vdots \\
S_{n_bn_b}^b
\end{bmatrix}
$$

(5.66)

Unlike the one-element method, the global stiffness matrix, $S$, is indeed sparse though generally not diagonal for the multi-element spectral element Galerkin-NI method.
The put option price can be discretized at the global grid points, which are the interpolation points of the global basis function, i.e., the combination of interpolation points of the two basis function sets. The discretized put price can be written as,

\[ P = [P_0, P_1, \ldots, P_n^a, P_{n^a+1}, \ldots, P_{n^a+n^b}]' \]

(5.67)

While there are total \( n^a + n^b + 1 \) global basis functions as defined in (5.39)-(5.43). The set of the global basis functions can be written in a vector form, \( \mathbf{v} \), as follows,

\[ \mathbf{v} = [v_0, v_1, \ldots, v_n^a, v_{n^a+1}, \ldots, v_{n^a+n^b}]' \]

(5.68)

Recall that the last local basis function in the first element is welded with the first local basis function in the second element and it forms one global basis function to enforce the continuity at the boundary. The global basis function is as follows,

\[ v_{n^a}(S) = \begin{cases} 
\phi_{n^a}(x^a), & S \in [0, K] \\
\phi_{0}^b(x^b), & S \in [K, \infty) 
\end{cases} \]

(5.69)
While the weak formulation for the two-element method under \( v(S) \) is as follows,

\[
(r - \sigma^2) \int_0^K P_S v(S) dS - \frac{1}{2} \sigma^2 \int_0^K P_S S^2 v_S dS + \int_0^K P_t v(S) dS - r \int_0^K P(S) v(S) dS = 0
\]

\[
(r - \sigma^2) \int_K^\infty P_S v(S) dS - \frac{1}{2} \sigma^2 \int_K^\infty P_S S^2 v_S dS + \int_K^\infty P_t v(S) dS - r \int_K^\infty P(S) v(S) dS = 0
\]  \hspace{1cm} (5.70)

Therefore, after the discretization, the weak formulation can be written as,

\[
\begin{bmatrix}
M_{00}^a & \cdots & M_{0n^a}^a \\
\vdots & \ddots & \vdots \\
M_{n^a0}^a & \cdots & M_{n^a n^a}^a + M_{00}^b & \cdots & M_{0n^b}^b \\
\vdots & \ddots & \vdots \\
M_{n^b0}^b & \cdots & M_{n^b n^b}^b
\end{bmatrix}
\begin{bmatrix}
\frac{\partial P_0}{\partial t} \\
\vdots \\
\frac{\partial P_{n^a+n^b}}{\partial t}
\end{bmatrix}
\]

\[
\begin{bmatrix}
S_{00}^a & \cdots & S_{0n^a}^a \\
\vdots & \ddots & \vdots \\
S_{n^a0}^a & \cdots & S_{n^a n^a}^a + S_{00}^b & \cdots & S_{0n^b}^b \\
\vdots & \ddots & \vdots \\
S_{n^b0}^b & \cdots & S_{n^b n^b}^b
\end{bmatrix}
\begin{bmatrix}
P_0 \\
\vdots \\
P_{n^a+n^b}
\end{bmatrix}
\]

\hspace{1cm} (5.71)

The “middle” row of the above matrix form is consistent with and is the consequence of the construction of the “middle” global basis function, \( v_{n^a}(S) \). Therefore, the weak formulation in a simpler matrix form is as follows,

\[
M \frac{\partial P}{\partial t} = SP
\]  \hspace{1cm} (5.72)
5.2.2 Time Discretization

The Crank-Nicolson time discretization method will be implemented here as well,

\[
M \frac{P^{k+1} - P^k}{\delta t} = S \frac{P^{k+1} + P^k}{2}
\]  

(5.73)

therefore, the put option price can work backwards from \( t_{k+1} \) to \( t_k \) as follows,

\[
P^k = \left( I + \frac{\delta t}{2} M^{-1} S \right)^{-1} \left( I - \frac{\delta t}{2} M^{-1} S \right) P^{k+1}
\]  

(5.74)

where \( \delta t = t_{k+1} - t_k \), and the put option final payoff \( P(T) \) is as follows,

\[
P(T) = \max\{K - S(T), 0\}
\]  

(5.75)

The initial put option price is \( P^0 \). As long as the put option price at the global grid points are available, the put option price at any arbitrary points (arbitrary stock prices) can be interpolated (expanded) by the global basis functions. Or equivalently, the put option price at each element can be interpolated with the local basis functions with the put option prices at local grid points, as follows,

\[
P^a(S^a) = P^a(S^a(x^a)) = P^a(x^a) = \sum_{i=0}^{n^a} \phi_i^a(x^a) P^a(x_i^a)
\]  

(5.76)

\[
P^b(S^b) = P^b(S^b(x^b)) = P^b(x^b) = \sum_{i=0}^{n^b} \phi_i^b(x^b) P^b(x_i^b)
\]  

(5.77)

5.2.3 SEM Option Greeks

As in the one-element approach, the option sensitivities, Delta (\( \Delta \)), Gamma (\( \Gamma \)) and Theta (\( \Theta \)) can be directly computed from the multi-element spectral element method. The algorithms are quiet identical with the one-element method. The option Delta and Gamma at the local grid points will first be computed, then the Greeks at any
other local points can be interpolated by the local basis functions in each element. The Delta and Gamma at the strike price can either be calculated from the left (first) element or the right (second) element, which will be exponentially close, i.e.,

\[ \Delta_{n_a}^a = \frac{1}{J_a} \sum_{j=0}^{n_a} D_{n_a j}^a P^a(x_j^a) = \frac{1}{J_b} \sum_{j=0}^{n_b} D_{0 j}^b P^b(x_j^b) = \Delta_0^b \]  
\[ \Gamma_{n_a}^a = \frac{1}{J_a^2} \sum_{j=0}^{n_a} B_{n_a j}^a P^a(x_j^a) = \frac{1}{J_b^2} \sum_{j=0}^{n_b} B_{0 j}^b P^b(x_j^b) = \Gamma_0^b \]  

While the Theta at the global grid points can directly be computed from the global weak matrix formulation, the Theta at any other points can be interpolated locally by using the corresponding local basis functions.

5.2.4 Results for Put Prices

The multi-element spectral element method (ME-SEM) using the Gauss-Lobatto-Legendre (GLL) and stable Gauss-Radau-Laguerre (GRL) quadrature and basis functions with Galerkin-NI approach is applied to a standard European put option with the following parameters, which are identical to the one-element method,

- Strike Price: \( K = 40.0 \)
- Interest Rate: \( r = 5.0\% \)
- Stock Volatility: \( \sigma = 25\% \)
- Option Maturity: \( T = 0.5 \text{ yr} \)

Because there is only one non-smooth point in the final payoff, only two elements will be sufficient for implementing the SEM on the put option. The first element uses the GLL quadrature and its basis functions, and the second uses the stable GRL quadrature and its basis functions. The error decays exponentially with the number of orders of the basis functions (\( p \)-refinement) and algebraically with the number of
elements ($h$-refinement), therefore it generally will be best only using the necessary number of elements, which will be two for the put option here.

In the following illustration, for the first element, the order of the GLL basis functions is set to 40, i.e., the total number of grid points is 41 for the first element, and for the second element, the order of the stable GRL basis functions is set to 20, i.e., the total number of grid points is 21 for the second element. Therefore there will be total $41 + 21 - 1 = 61$ grid points, which is the same number of total grid points in the one-element example. The same time step of about $10^{-5}$ year is implemented for the two-element SEM here. The computed put option surface over time-to-maturity and possible traded stock price is illustrated in Figure 5.6. The comparison of the calculated put option price with the analytical price from the Black-Scholes closed-form solution is plotted in Figure 5.7. The errors for the current put option price are significantly reduced with $L_2$-Error $= 2.9831 \times 10^{-12}$ and $L_{\infty}$-Error $= 4.6662 \times 10^{-12}$ because implementing the correct number of elements.

The $L_2$-Errors with increasing total number of grid points are graphed in Figure 5.8 with $\alpha = 0.2192$, which is much larger than $\alpha = 0.0732$ in the one-element spectral element approach in Figure 5.3. It means the two-element spectral element approach indeed speeds up the convergence significantly when eliminating the non-smoothness by using the multi-element spectral element method (ME-SEM). Because of the nature of the put option, which decays rapidly with increasing stock price, the order number of the GLL basis functions in the first element is set roughly twice the order number of the stable GRL basis functions in the second element to further increase the convergence rate with the increasing total number of grid points. While the $L_2$-Errors with respect to the time step are depicted in Figure 5.9. The estimate slope of convergence, $\beta = 1.9952$, is very close to the theoretical value of 2.00, which means a second-order convergence as described in Section 4.4. There are quiet similar results of $L_{\infty}$-Errors for both Figure 5.8 and Figure 5.9.
**Figure 5.6**: The standard European put option prices over time-to-maturity and stock price using two-element spectral element method with GLL and stable GRL.

**Figure 5.7**: Comparison between analytical Black-Scholes prices and two-element spectral element method prices for the standard European put option at inception.
Figure 5.8: The $L_2$-Errors for the standard European put option prices with respect to the total number of grid points using two-element method with $\delta t = 10^{-5}$ year.

Figure 5.9: The $L_2$-Errors for the standard European put option prices with respect to time step using the two-element method with the total number of grid points = 61.
5.2.5 Results for Put Greeks

In the numerical results for the Greeks, the same option parameters will be used for computing the put option Greeks, specifically Delta (\(\Delta\)), Gamma (\(\Gamma\)) and Theta (\(\Theta\)). The order of the Gauss-Lobatto-Legendre (GLL) basis functions of the first element is set to 40, i.e., the total number of grid points is 41 for the first element, and the order of the stable Gauss-Radau-Laguerre (GRL) basis functions of the second element is set to 20, i.e., the total number of grid points is 21 for the second element. Therefore there will be total 61 grid points. Same order number construction will be used for all the Greeks. The time step is set to about \(10^{-5}\) year.

The comparison of the calculated put option Delta with the analytical Delta from the Black-Scholes closed-form solution is plotted in Figure 5.10. The errors for the current put option Delta, \(\Delta(t = 0)\), are very impressive with \(L_2\)-Error = \(2.6017 \times 10^{-10}\) and \(L_\infty\)-Error = \(8.1334 \times 10^{-10}\). Because the nature of the spectral element method (SEM), the derivatives (the Greeks) of the original solution (\(P\)) will be less accurate than the function itself, which is especially true for the second derivative, Gamma (\(\Gamma\)), though all are accurate enough for practical uses. As discussed theoretically, the Deltas at the boundary between the two element are indeed exponentially close with a relative error of \(6.2970 \times 10^{-12}\).

With the same allocation mechanism of order numbers for the two elements as in computing the price itself, the \(L_2\)-Errors for the Delta (\(\Delta\)) with increasing total number of grid points are graphed in Figure 5.11 with \(\alpha = 0.1799\), which is slightly slower than \(\alpha = 0.2192\) for the put price in Figure 5.8, as expected. While the \(L_2\)-Errors with respect to the time step are depicted in Figure 5.12. The estimate slope of error convergence, \(\beta = 1.9825\), is very close to the theoretical value of 2.00, which means a second-order convergence as described in Section 4.4. There are quiet similar results of \(L_\infty\)-Errors for both Figure 5.11 and Figure 5.12.
Figure 5.10: Comparison between analytical Black-Scholes Deltas and two-element spectral element method Deltas for the standard European put option at inception.

Figure 5.11: The $L_2$-Errors for the standard European put Deltas with respect to the total number of grid points using the two-element method with $\delta t = 10^{-5}$ year.
The comparison of the calculated put option Gamma with the analytical Gamma is plotted in Figure 5.13. The errors for the current put option Gamma, $\Gamma(t = 0)$, are very impressive with $L_2$-Error = $1.0029 \times 10^{-8}$ and $L_\infty$-Error = $3.0474 \times 10^{-8}$. As discussed theoretically, the Gammas at the boundary between the two elements are indeed exponentially close with a relative error of $6.8571 \times 10^{-10}$.

With the same allocation mechanism of order numbers for the two elements as before, the $L_2$-Errors for the Gamma ($\Gamma$) with increasing total number of grid points are graphed in Figure 5.14 with $\alpha = 0.1569$, which is slightly slower than $\alpha = 0.1799$ for the put option Delta in Figure 5.11, as expected. While the $L_2$-Errors with respect to the time step are depicted in Figure 5.15. The estimate slope of error convergence, $\beta = 1.9718$, is very close to the theoretical value of 2.00, which means a second-order convergence as described in Section 4.4. There are quiet similar results of $L_\infty$-Errors for both Figure 5.14 and Figure 5.15.
Figure 5.13: Comparison between the analytical Black-Scholes Gammas and from the two-element spectral element method for the European put option at inception.

Figure 5.14: The $L_2$-Errors for the standard European put Gammas with respect to the total number of grid points using two-element method with $\delta t = 10^{-5}$ year.
Figure 5.15: The $L_2$-Errors for the standard European put Gammas with respect to time step using two-element method with the total number of grid points = 61.

The errors of Gamma ($\Gamma$) are actually significantly smaller for the stock prices at a reasonable region of interest. For example, if the penny stock is excluded when computing the errors, that is, only stock price no less than $1 is counted, the accuracy of Gamma ($\Gamma$) is improved by about the order of 2, as shown in Figure 5.16. For instance, the $L_2$-Error $= 1.0029 \times 10^{-8}$ for the total number of grid points of 61 in Figure 5.14, while $L_2$-Error $= 6.2220 \times 10^{-10}$ for the same total number of grid points in Figure 5.16 by just excluding the stock prices less than one. The convergence rate is also increased, from $\alpha = 0.1569$ in Figure 5.14 to $\alpha = 0.1870$ in Figure 5.16.

While the $L_2$-Errors with respect to the time step are significantly decreased too about the order of 2, as depicted in Figure 5.17. The estimate slope of error convergence, $\beta = 1.9995$, is extremely close to the theoretical value of 2.00, which means a second-order convergence as described in Section 4.4. There are quiet similar results of $L_\infty$-Errors for both Figure 5.16 and Figure 5.17.
Figure 5.16: The $L_2$-Errors for the put Gammas of stock prices no less than $\$1$ with respect to total number of grid points using two-element with $\delta t \approx 10^{-5}$ year.

Figure 5.17: The $L_2$-Errors for the put Gammas of stock prices no less than $\$1$ with respect to time step using two-element with total number of grid points = 61.
The comparison of the calculated put option Theta with the analytical Theta is plotted in Figure 5.18. The errors for the current put option Theta, $\Theta(t = 0)$, are very impressive with $L_2$-Error $= 1.4417 \times 10^{-10}$ and $L_8$-Error $= 3.2930 \times 10^{-10}$. Because the Thetas at all the grid points are directly computed from the matrix weak formulation of the Black-Scholes PDE for the multi-element spectral element method, there are no two distinct values at the boundary between two elements.

With the same allocation mechanism of order numbers for the two elements as before, the $L_2$-Errors for the Theta ($\Theta$) with increasing total number of grid points are graphed in Figure 5.19 with $\alpha = 0.1929$. While the $L_2$-Errors with respect to the time step are depicted in Figure 5.20. The estimate slope of error convergence, $\beta = 1.9990$, is very close to the theoretical value of 2.00, which means a second-order convergence as described in Section 4.4. There are quiet similar results of $L_8$-Errors for both Figure 5.19 and Figure 5.20.

![Figure 5.18](image-url)  
**Figure 5.18:** Comparison between analytical Black-Scholes Thetas and two-element spectral element method Thetas for the standard European put option at inception.
Figure 5.19: The $L_2$-Errors for the standard European put Thetas with respect to the total number of grid points using the two-element method with $\delta t \doteq 10^{-5}$ year.

Figure 5.20: The $L_2$-Errors for the standard European put Thetas with respect to time step using the two-element method with the total number of grid points $= 61$. 

158
5.3 Comparison with Finite Difference Method

In this section, the comparison between the finite difference method (FDM) and spectral element method (SEM) for the same European put option will be presented, both of which use the Crank-Nicolson method. In the SEM, only the knowledge of one Dirichlet boundary condition, i.e., the put price goes to zero when the stock price goes to infinity, is used only for choosing the suitable basis functions, i.e. the stable Gauss-Radau-Laguerre (GRL) basis functions for the second element, it is not actually used in the equations. The boundary term in (5.35) is as follows,

\[
\frac{1}{2} \sigma^2 S^2 v(S) P_s \bigg|_0 ^\infty
\]

This term will be zero when \( S = 0 \) and zero too when \( S \to \infty \), because the basis function \( v(S) \) decays exponentially with \( S \). Therefore, no Neumann condition of \( P_S \) is needed for the computation here, although it will go to zero too. On contrary, the finite difference method heavily uses the following two Dirichlet boundary conditions for stock price at 0 and at a reasonable large price \( S_{\text{max}} \),

\[
P(0, t) = Ke^{-r(T-t)}, \quad \forall \ 0 \leq t \leq T
\]

\[
P(S_{\text{max}}, t) \approx 0, \quad \text{for large} \ S_{\text{max}}
\]

The comparison of \( L^2 \)-Errors for the European put price with respect to the total number of grid points using the finite difference method (FDM) and spectral element method (SEM) with \( \delta t = 10^{-5} \) year is illustrated in Figure 5.21. As can seen, the errors in the SEM converge much faster than in the FDM, one is exponentially and the other is algebraically. While the comparison of computational time for the put option with respect to the total number of grid points using the finite difference method (FDM) and spectral element method (SEM) with \( \delta t = 10^{-5} \) year is graphed in Figure 5.22. The FDM costs much more time than the SEM with lower accuracy. Similar figures with more grid points are depicted in Figure 5.23 and 5.24.
Figure 5.21: Comparison of $L_2$-Errors for the standard European put price with respect to the total number of grid points using FDM and SEM with $\delta t \doteq 10^{-5}$ year.

Figure 5.22: Comparison of computational time for the standard put option with respect to the total number of grid points using FDM and SEM with $\delta t \doteq 10^{-5}$ year.
Figure 5.23: Comparison of $L_2$-Errors for the standard European put price with respect to the total number of grid points using FDM and SEM with $\delta t = 10^{-5}$ year.

Figure 5.24: Comparison of computational time for the standard put option with respect to the total number of grid points using FDM and SEM with $\delta t = 10^{-5}$ year.
5.4 SEM on Black-Scholes PDE with Dividends

If the underlying stock pays dividends with continuously compounding yield of \( q \), then the stock price follows the generalized geometric Brownian motion as follows,

\[
\frac{dS(t)}{S(t)} = (\mu - q)dt + \sigma dW(t)
\]  

(5.83)

The corresponding partial differential equation for the put option becomes,

\[
\frac{1}{2} \sigma^2 S^2(t)P_{SS} + (r - q)S(t)P_S + P_t - rP = 0
\]  

(5.84)

where \( r - q \) is the cost of carry of the stock, with a final condition,

\[
P(S, T) = \max\{K - S(T), 0\}
\]  

(5.85)

The closed-form solutions for the call and put options are,

\[
P(S, t) = e^{-(r-q)T-t} \mathcal{N}(-d_2) - S(t)e^{-(r-q)T-t} \mathcal{N}(-d_1)
\]  

(5.86)

where,

\[
d_1 = \frac{\ln \left( \frac{S(t)}{K} \right) + (r - q + \frac{1}{2}\sigma^2)(T - t)}{\sigma \sqrt{T - t}}
\]  

(5.87)

\[
d_2 = d_1 - \sigma \sqrt{T - t}
\]  

(5.88)

The associated Greeks for the put option are presented in Table 2.2.

Then, the weak formulation for the partial differential equation (5.84) is

\[
\frac{1}{2} \sigma^2 S^2 v(S) P_S \bigg|_0^\infty + (r - q - \sigma^2) \int_0^\infty P_S S v(S) dS
\]

\[
- \frac{1}{2} \sigma^2 \int_0^\infty P_S S^2 v_S dS + \int_0^\infty P_t v(S) dS - r \int_0^\infty P(S) v(S) dS = 0
\]  

(5.89)

Exact same procedures can be implemented here as in the standard European put option except the cost of carry of the stock will be \( r - q \) instead of \( r \).
As a numerical example, a put option with the same parameters as in Section 5.2.4 except a non-zero dividend yield has the parameter values as follows,

\[
\begin{align*}
\text{Strike Price: } & \quad K = 40.0 \\
\text{Interest Rate: } & \quad r = 5.0\% \\
\text{Dividend Yield: } & \quad q = 2.5\% \\
\text{Stock Volatility: } & \quad \sigma = 25\% \\
\text{Option Maturity: } & \quad T = 0.5 \text{ yr}
\end{align*}
\]

As for the European put option without dividend, using 41 grid points for the first element with GLL basis functions and 21 grid points for the second element with stable GRL basis functions (total 61 grid points) with \( \delta t \approx 10^{-5} \) year in spectral element method, the numerical results are presented in the following table,

Table 5.1: Errors for the European Put Option with Dividend

<table>
<thead>
<tr>
<th>Variable</th>
<th>( L_2 )-Error</th>
<th>( L_\infty )-Error</th>
<th>Boundary Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>Put ((P))</td>
<td>(5.1012 \times 10^{-12})</td>
<td>(8.7032 \times 10^{-12})</td>
<td>Not Applicable</td>
</tr>
<tr>
<td>Delta ((\Delta))</td>
<td>(2.3269 \times 10^{-10})</td>
<td>(4.6115 \times 10^{-10})</td>
<td>(8.3920 \times 10^{-12})</td>
</tr>
<tr>
<td>Gamma ((\Gamma))</td>
<td>(2.5823 \times 10^{-8})</td>
<td>(2.4905 \times 10^{-7})</td>
<td>(7.0176 \times 10^{-10})</td>
</tr>
<tr>
<td>Theta ((\Theta))</td>
<td>(1.8995 \times 10^{-10})</td>
<td>(4.3462 \times 10^{-10})</td>
<td>Not Applicable</td>
</tr>
</tbody>
</table>

where the boundary error is the relative difference between the Greeks calculated from the first element and the second element, which are exponentially close.

The surface of the put is illustrated in Figure 5.25. The comparison with analytical value is graphed in Figure 5.26. The \( L_2 \)-Error converge rates with respect to the number of grid points and the time step are depicted in Figure 5.27 and Figure 5.28 respectively. Similar results for the Greeks are illustrated in Figure 5.29 – 5.37.
Figure 5.25: The standard European put option prices over time-to-maturity and stock price which has constant dividend using two-element spectral element method.

Figure 5.26: Comparison between analytical Black-Scholes prices and two-element spectral element method prices for the standard European put option with dividend.
Figure 5.27: The $L_2$-Errors for the standard European put option with dividend with respect to the total number of grid points using the SEM with $\delta t = 10^{-5}$ year.

Figure 5.28: The $L_2$-Errors for the standard European put option with dividend with respect to time step using the SEM with the total number of grid points = 61.
Figure 5.29: Comparison between analytical Black-Scholes Deltas and two-element spectral element method Deltas for the standard European put option with dividend.

Figure 5.30: The $L_2$-Errors for the standard European put Deltas with dividend with respect to the total number of grid points using the SEM with $\delta t = 10^{-5}$ year.
Figure 5.31: The $L_2$-Errors for the standard European put Deltas with dividend with respect to time step using the SEM with the total number of grid points = 61.

Figure 5.32: Comparison between the analytical Black-Scholes Gammas and from the two-element spectral element method for the put option with constant dividend.
Figure 5.33: The $L_2$-Errors for the standard European put Gammas with dividend with respect to the total number of grid points using the SEM with $\delta t = 10^{-5}$ year.

Figure 5.34: The $L_2$-Errors for the standard European put Gammas with dividend with respect to time step using the SEM with the total number of grid points = 61.
**Figure 5.35:** Comparison between analytical Black-Scholes Thetas and two-element spectral element method Thetas for the standard European put option with dividend.

**Figure 5.36:** The $L_2$-Errors for the standard European put Thetas with dividend with respect to the total number of grid points using the SEM with $\delta t = 10^{-5}$ year.
Figure 5.37: The $L_2$-Errors for the standard European put Thetas with dividend with respect to time step using the SEM with the total number of grid points = 61.

Comparison with Finite Difference Method

The comparison of the $L_2$-Errors for the European put option price with respect to the total number of grid points using the finite difference method (FDM) and spectral element method (SEM) with $\delta t = 10^{-5}$ year is illustrated in Figure 5.38. As can seen, the errors in the SEM converge much faster than in the FDM, one is exponentially and the other is algebraically. While the comparison of computational time for the put option with respect to the total number of grid points using the finite difference method (FDM) and the spectral element method (SEM) with $\delta t = 10^{-5}$ year is graphed in Figure 5.39. The FDM costs much more time than the SEM with lower accuracy for the same number of grid points. The computational time for the FDM also grows much faster than the SEM. For more grid points, the comparison of $L_2$-Errors of the two methods are depicted in Figure 5.40 and the comparison of computation time for the two methods are plotted in Figure 5.41.
**Figure 5.38**: Comparison of $L_2$-Errors for the standard European put option price with dividend with respect to the total number of grid points using FDM and SEM.

**Figure 5.39**: Comparison of computational time for the standard put option with constant dividend with respect to total number of grid points using FDM and SEM.
Figure 5.40: Comparison of $L_2$-Errors for the standard European put option price with dividend with respect to the total number of grid points using FDM and SEM.

Figure 5.41: Comparison of computational time for the standard put option with constant dividend with respect to total number of grid points using FDM and SEM.
5.5 SEM on A European Condor Spread Option

The European condor option is an option trading strategy to earn a limited profit when the underlying security is perceived to have little volatility. The condor option strategy can be realized by a combination of simultaneously shorting and longing of plain-vanilla European call or put options. Therefore, a condor option can be thought as a spread between regular options, which can also be named as condor spread. For example, a European condor option with the following final payoff,

![Figure 5.42: Final payoff of a European condor spread option.](image)

can be replicated by the following portfolio with standard calls,

- Long one unit of a European call option with a strike price at $30,
- Short one unit of a European call option with a strike price at $35,
- Short one unit of a European call option with a strike price at $45,
- Long one unit of a European call option with a strike price at $50.
For the sake of experiment for more than two elements, the European condor option can be solved using a five-element spectral element method since the option final payoff have four non-smooth points, which are $30, $35, $45 and $50. This European condor option has the other option parameters as follows,

\begin{align*}
\text{Interest Rate: } & \quad r = 5.0\% \\
\text{Stock Volatility: } & \quad \sigma = 25\% \\
\text{Option Maturity: } & \quad T = 0.5 \text{ yr}
\end{align*}

The stock price domain will be separated into five elements, which are $[0, 30], [30, 35], [35, 45], [45, 50]$ and $[50, \infty)$. The first four elements are approximated by the Gauss-Lobatto-Ledgenre (GLL) functions and the last element is expanded by the stable Gauss-Radau-Laguerre (GRL) basis functions because of the semi-infinite interval. For the last element, $S_{\text{max}}$ will be set as $180$, because unlike the standard put option struck at $40$, this condor option price decays much slower. For the illustrative purpose, the results will be plotted until $120$, while all the computations are based on the stock price stretched to $180$. Using 45 grid points per element with $\delta t \approx 10^{-5}$ year, the numerical results are presented in the following table,

<table>
<thead>
<tr>
<th>Variable</th>
<th>$L_2$-Error</th>
<th>$L_x$-Error</th>
<th>Boundary Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>Condor</td>
<td>$3.5029 \times 10^{-12}$</td>
<td>$4.8527 \times 10^{-12}$</td>
<td>Not Applicable</td>
</tr>
<tr>
<td>Delta ($\Delta$)</td>
<td>$3.1853 \times 10^{-11}$</td>
<td>$1.0976 \times 10^{-10}$</td>
<td>$6.0972 \times 10^{-12}$</td>
</tr>
<tr>
<td>Gamma ($\Gamma$)</td>
<td>$2.6320 \times 10^{-10}$</td>
<td>$1.6275 \times 10^{-9}$</td>
<td>$1.4486 \times 10^{-9}$</td>
</tr>
<tr>
<td>Theta ($\Theta$)</td>
<td>$9.7708 \times 10^{-11}$</td>
<td>$2.9399 \times 10^{-10}$</td>
<td>Not Applicable</td>
</tr>
</tbody>
</table>
where the boundary error is the relative difference between the Greeks calculated from the left element and the right element, which are exponentially close.

The condor option surface and the comparison of the price with the analytical value are graphed in Figure 5.43 and Figure 5.44. The $L_2$-Errors with increasing number of grid points per element are illustrated in Figure 5.45 with the per-element slope $\alpha = 0.3225$. The errors indeed converge exponentially. For simplicity, each element has a same number of grid points. It is definitely possible to allocate an optimized number of grid points for a specific element according to the behavior of the solution function. The $L_2$-Errors with respect to the time step are depicted in Figure 5.46. The estimate slope of convergence, $\beta = 1.9991$, is extremely close to the theoretical value of 2.00. There are quiet similar results for the $L_\infty$-Errors. Similar results for the Greeks, Delta, Gamma, and Theta, are illustrated in Figure 5.47 – 5.55. Comparison with the FDM is illustrated in Figure 5.56 and Figure 5.57.

![Figure 5.43](image)

**Figure 5.43**: The European condor option prices over time-to-maturity and stock price using the five-element spectral element method with 45 grid points per element.
Figure 5.44: Comparison between the analytical option prices and the spectral element method prices for the European condor option with 45 grid points per element.

Figure 5.45: The $L_2$-Errors for the European condor option with respect to the number of grid points per element using the SEM with the time step $\delta t = 10^{-5}$ year.
Figure 5.46: The $L_2$-Errors for price of the European condor spread option with respect to the time step using the SEM with number of grid points per element = 45.

Figure 5.47: Comparison between the analytical option Deltas and the spectral element method Deltas for the European condor option with 45 points per element.
Figure 5.48: The $L_2$-Errors for the European condor Delta with respect to the number of grid points per element using the SEM with the time step $\delta t \doteq 10^{-5}$ year.

Figure 5.49: The $L_2$-Errors for price of the European condor spread Delta with respect to the time step using the SEM with number of grid points per element $= 45$. 

178
Figure 5.50: Comparison between the analytical option Gammas and the spectral element method Gammas for the European condor option with 45 points per element.

Figure 5.51: The $L_2$-Errors for the European condor Gamma with respect to the number of grid points per element using the SEM with the time step $\delta t = 10^{-5}$ year.
**Figure 5.52:** The $L_2$-Errors for price of the European condor spread Gamma with respect to the time step using the SEM with number of grid points per element = 45.

**Figure 5.53:** Comparison between the analytical option Thetas and the spectral element method Thetas for the European condor option with 45 points per element.
**Figure 5.54**: The $L_2$-Errors for the European condor Theta with respect to the number of grid points per element using the SEM with the time step $\delta t \approx 10^{-5}$ year.

**Figure 5.55**: The $L_2$-Errors for price of the European condor spread Theta with respect to the time step using the SEM with number of grid points per element = 45.
Figure 5.56: Comparison of $L_2$-Errors for the European condor option price with respect to the total number of grid points using FDM and spectral element method.

Figure 5.57: Comparison of computational time for the European condor option with respect to total number of grid points using FDM and spectral element method.
6

Spectral Element on European Binary Option

6.1 Introduction and Challenges

Binary option, also named as digital option, is one kind exotic option that pays a fixed amount of financial asset or nothing at all. Mainly it has two types of option, cash-or-nothing and asset-or-nothing, where the asset generally will be the underlying security of the option. Like regular options, the binary options has two exercise styles, European and American. Depending on the bullish or bearish view, there are call and put binary options for investors.

The binary option are traditionally traded over-the-counter. During the summer of 2008, a few binary option contracts started to be exchange-traded, for example, binary options on the SPX and the VIX. Binary options have become extremely popular and very liquid since then. Binary options not only exist standalone, but also are commonly embedded in other exotic options and complex instruments, for instance, some barrier option and the range note, a structured interest rate product. Therefore, pricing methodologies of binary options can be applied to a wide variety of derivative products, especially complicated securities with binary option features.
In this chapter, European binary put option will be researched as an example using the spectral element method (SEM). The vanilla binary cash-or-nothing put option has the following final payoff at the maturity,

\[
P(S, T) = \begin{cases} 
  1 & \text{if } S(T) < K \\
  0 & \text{if } S(T) \geq K
\end{cases}
\]  

(6.1)

The final payoff diagram at the maturity looks like the following,

This vanilla European binary put option does have a closed-form solution,

\[
P(S, t) = e^{-r(T-t)}N(-d_2)
\]

(6.2)

where \(d_2\) is defined same as in the regular European put option,

\[
d_2 = \frac{\ln \frac{S(t)}{K} + (r - \frac{1}{2}\sigma^2)(T - t)}{\sigma \sqrt{T - t}}
\]

(6.3)

The European binary cash-or-nothing put option essentially is the discounted probability of the option which will be in-the-money, i.e., the stock price will be less than the strike price at the maturity.
While the Greeks for the European cash-or-nothing put option also have the closed formulas as in the Table 6.1, where Delta, Gamma and Theta are presented.

Table 6.1: Option Greeks for European Binary Put

<table>
<thead>
<tr>
<th>Name</th>
<th>Formula</th>
<th>Binary Put Greek</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delta</td>
<td>[ \Delta = \frac{\partial P}{\partial S} ]</td>
<td>[-\frac{e^{-r(T-t)}N'(d_2)}{\sigma S\sqrt{T-t}} ]</td>
</tr>
<tr>
<td>Gamma</td>
<td>[ \Gamma = \frac{\partial^2 P}{\partial S^2} ]</td>
<td>[ \frac{e^{-r(T-t)}N'(d_2)}{\sigma^2 S(T-t)} \left[ d_2 + \sigma \sqrt{T-t} \right] ]</td>
</tr>
<tr>
<td>Theta</td>
<td>[ \Theta = \frac{\partial P}{\partial t} ]</td>
<td>[ \frac{rN(-d_2)}{e^{r(T-t)}} - \frac{N'(-d_2)}{e^{r(T-t)}} \left[ \frac{d_2}{2(T-t)} - \frac{r - \frac{1}{2} \sigma^2}{\sigma \sqrt{T-t}} \right] ]</td>
</tr>
</tbody>
</table>

The challenges for pricing this kind option with discontinuous payoff are that the final payoff is not only not smooth but also not continuous. Therefore, the conventional finite difference method (FDM) creates significant numerical errors around the strike in the solutions for the binary options especially for their Greeks [63]. The FDM analysis with variable time-stepping is discussed in [63] and a finite element method (FEM) using special time-stepping in conjunction with various procedures for smoothing discontinuities is conducted in [64]. The FEM for diffusion problems with irregularities in the initial or boundary data is presented in [65].

Because of the unique property of the discontinuous payoff, it will be extremely difficult for the spectral element method to enforce the final condition at the strike price. Without correct enforcement, the convergence rate of the errors will be significantly reduced. In the following sections, two alternative methodologies are proposed and verified by numerical results with high accuracy. One is the approximation method by replicating the binary option with regular options in the limit. The other is the discontinuous payoff spectral element method (DP-SEM) by carefully enforcing the final condition without smoothing any discontinuity.
6.2 Approximations and Results

6.2.1 Approximations

The final payoff of the European binary put option can be replicated by longing some shares of a regular put option at the strike of \( K + \epsilon \) and shorting the same number of shares of a regular put option at the strike of \( K - \epsilon \) as \( \epsilon \) goes to infinitely small.

In order to have the final payoff priced at $1 as the binary put option, the number of shares of the regular put options will be set to \( 1/(2\epsilon) \), that is,

\[
P(K, T) = \frac{P_r(K + \epsilon, T) - P_r(K - \epsilon, T)}{2\epsilon}, \quad \text{as} \quad \epsilon \to 0 \tag{6.4}
\]

where \( P_r \) is denoted as the regular put option price. As \( \epsilon \) becomes smaller and smaller, the final payoff of the put option spread will approach the final payoff of the binary put option. That is, the binary put option can regarded as the limit of the put option spread as the difference between the two strikes goes to zero. This kind approximation of the binary put option final payoff is illustrated in the Figure 6.2.

![Figure 6.2: European cash-or-nothing binary put final payoff approximation.](image-url)
Therefore, the binary put option price at any time before maturity can be computed similarly as the scaled difference between two regular put options as follows,

\[ P(K,t) = \frac{P_r(K + \epsilon, t) - P_r(K - \epsilon, t)}{2\epsilon}, \quad \text{as} \quad \epsilon \to 0 \quad (6.5) \]

For the practical use, a moderate small \( \epsilon \) will be adequate for the approximation. The pricing formula can also be written as in a form of partial derivative as follows,

\[ P(K,t) = \frac{\partial P_r(K, t)}{\partial K} \quad (6.6) \]

6.2.2 Numerical Results

A European binary cash-or-nothing put option with the same parameters as in a regular European put option in Section 5.2.4 is computed using the proposed approximation method. The European binary put option parameters are as follows,

- Strike Price: \( K = 40.0 \)
- Interest Rate: \( r = 5.0\% \)
- Stock Volatility: \( \sigma = 25\% \)
- Option Maturity: \( T = 0.5 \text{ yr} \)

For illustration purpose, the final payoff will be scaled from $1 to the strike price of \( K \). Two-element spectral element method (SEM) using the Gauss-Lobatto-Legendre (GLL) and stable Gauss-Radau-Laguerre (GRL) quadrature and basis functions with Galerkin-NI approach is applied to the European binary put option. The first element uses the GLL quadrature and its basis functions, and the second uses the stable GRL quadrature and its basis functions.

As in Section 5.2.4, for the first element, the order of the GLL basis functions is set to 40, i.e., the total number of grid points is 41 for the first element, and for
the second element, the order of the stable GRL basis functions is set to 20, i.e., the total number of grid points is 21 for the second element. Therefore there will be total $41 + 21 - 1 = 61$ grid points. The time step of about $10^{-4}$ year is implemented for the two-element SEM here and the $\epsilon$ is set about $10^{-3}$.

The computed binary put option surface over time-to-maturity and possible traded stock price is illustrated in Figure 6.3. The comparison of the calculated binary put option price with the analytical price from the closed-form solution is plotted in Figure 6.4. The errors for the current binary put option price are very impressive with $L_2$-Error $= 2.3351 \times 10^{-9}$ and $L_{_{\infty}}$-Error $= 4.1242 \times 10^{-9}$.

Because of the nature of the Gauss-Lobatto-Legendre (GLL) and stable Gauss-Radau-Laguerre (GRL) quadratures as the grid points are more dense around $\pm 1$ for the GLL quadrature and around 0 for the stable GRL quadrature, the so-called Runge phenomenon, is avoided at the boundaries. Consequently, the computed European binary put option prices using the two-element spectral element method (SEM) do not create large numerical errors around the strike price at any time before the maturity as shown in the option price surf plot in Figure 6.3.

The $L_2$-Errors with increasing total number of grid points are graphed in Figure 6.5 with the converge slope $\alpha = 0.1964$. It does show that the error converges exponentially with the total number of the grid points. Because of the nature of the corresponding regular put option, which decays rapidly with increasing stock price, the order number of the GLL basis functions in the first element is set roughly twice the order number of the stable GRL basis functions in the second element to further increase the convergence rate while increasing total number of grid points. The $L_2$-Errors with respect to the time step are depicted in Figure 6.6. The estimate slope of convergence, $\beta = 1.9286$, is very close to the theoretical value of 2.00, which means a second-order convergence as described in Section 4.4. There are quiet similar results of the $L_{_{\infty}}$-Errors for both Figure 6.5 and Figure 6.6.
Figure 6.3: The standard European binary put prices over time-to-maturity and stock price using two-element spectral element method with GLL and stable GRL.

Figure 6.4: Comparison between analytical closed-form prices and two-element spectral element method prices for the standard European binary put at inception.
Figure 6.5: The $L_2$-Errors for the standard European binary put prices with respect to the total number of grid points using two-element method with $\delta t = 10^{-4}$ year.

Figure 6.6: The $L_2$-Errors for the standard European binary put prices with respect to time step using the two-element method with the total number of grid points = 61.
In the numerical results for the Greeks, the same parameters will be used for computing the binary put option Greeks. The order of the GLL basis functions of the first element is set to 40, i.e., the total number of grid points is 41 for the first element, and the order of the stable GRL basis functions of the second element is set to 20, i.e., the total number of grid points is 21 for the second element. Therefore there will be total 61 grid points. The time step is set to about $10^{-4}$ year.

The comparison of the calculated binary put option Delta with the analytical Delta from the closed-form solution is plotted in Figure 6.7. The errors for the current put option Delta, $\Delta(t = 0)$, are very impressive with $L_2$-Error $= 4.9502 \times 10^{-8}$ and $L_\infty$-Error $= 2.0806 \times 10^{-7}$. Because the nature of the spectral element method (SEM), the derivatives (the Greeks) of the original solution ($P$) will be less accurate than the function itself, though they are accurate enough for practical uses. With the same allocation mechanism of order numbers for the two elements as in computing the price itself, the $L_2$-Errors for the Delta ($\Delta$) with increasing total number of grid points are graphed in Figure 6.8 with the slope of $\alpha = 0.2193$.

The comparison of the calculated binary put option Gamma with the analytical Gamma is plotted in Figure 6.9. The errors for the current put option Gamma, $\Gamma(t = 0)$, are very impressive with $L_2$-Error $= 1.5289 \times 10^{-6}$ and $L_\infty$-Error $= 1.2423 \times 10^{-5}$. With the same allocation mechanism of order numbers for the two elements as before, the $L_2$-Errors for the Gamma ($\Gamma$) with increasing total number of grid points are graphed in Figure 6.10 with the slope of $\alpha = 0.1898$.

The comparison of the calculated binary put option Theta with the analytical Theta is plotted in Figure 6.11. The errors for the current put option Theta, $\Theta(t = 0)$, are very impressive with $L_2$-Error $= 3.9644 \times 10^{-8}$ and $L_\infty$-Error $= 3.7805 \times 10^{-8}$. With the same allocation mechanism of order numbers for the two elements as before, the $L_2$-Errors for the Theta ($\Theta$) with increasing total number of grid points are graphed in Figure 6.12 with the slope of $\alpha = 0.1711$. 
Figure 6.7: Comparison between analytical Deltas and the two-element spectral element method Deltas for the standard European binary put option at inception.

Figure 6.8: The $L_2$-Errors for the European binary put Deltas with respect to the total number of the grid points using the two-element method with $\delta t = 10^{-4}$ year.
Figure 6.9: Comparison between analytical closed-form Gammas and two-element spectral element method Gammas for the European binary put option at inception.

Figure 6.10: The $L_2$-Errors for the European binary put Gammas with respect to the total number of grid points using the two-element method with $\delta t = 10^{-3}$ year.
Figure 6.11: Comparison between analytical closed-form Thetas and two-element spectral element method Thetas for the European binary put option at the inception.

Figure 6.12: The $L_2$-Errors for the European binary put Thetas with respect to the total number of grid points using the two-element method with $\delta t = 10^{-4}$ year.
6.3 DP-SEM Method and Results

6.3.1 The Final Condition

A standard European binary put option price will follow the same Black-Scholes partial difference equation (PDE) as the standard European put option as follows,

\[ \frac{1}{2} \sigma^2 S^2(t) P_{SS} + rS(t)P_S + P_t - rP = 0 \]  

(6.7)

where \( t \geq 0, S \geq 0, r \) and \( \sigma \) are constant, and the final condition at maturity is,

\[ P(S(T), T) = \begin{cases} 1 & \text{if } S(T) < K \\ 0 & \text{if } S(T) \geq K \end{cases} \]  

(6.8)

After using the same weak formulation procedure as for the standard European put option, the weak form can be written as the same matrix equation as follows,

\[ M \frac{\partial P}{\partial t} = SP \]  

(6.9)

where the mass matrix, \( M \), is diagonal because of the unique property of the basis functions, while the stiffness matrix here, \( S \), is sparse for the multi-element method. The dynamics of (6.9) is exactly true for any time before maturity, however, there needs some special treatment when involving the time at the maturity.

Because only one discontinuous point, which is at the strike price, resides at the final payoff, the exact same constructions for the two elements as in the regular put option can be applied for the binary put. Then, the global mass matrix, \( M \), is,

\[
M = \begin{bmatrix}
M_{00}^a & \cdots & M_{0n^a}^a \\
\vdots & \ddots & \vdots \\
M_{n^an^a0}^a & \cdots & M_{n^an^a0}^a + M_{00}^b & \cdots & M_{0n^b}^b \\
\vdots & \ddots & \vdots \\
M_{n^bn^b0}^b & \cdots & M_{n^bn^b0}^b
\end{bmatrix} \]  

(6.10)
the global stiffness matrix, $S$, is constructed as follows,

$$
S = \begin{bmatrix}
S_{00} & \cdots & S_{0n^a} \\
\vdots & \ddots & \vdots \\
S_{n^a 0} & \cdots & S_{n^a n^a} + S_{00} & \cdots & S_{0n^b} \\
\vdots & \ddots & \vdots \\
S_{n^b 0} & \cdots & S_{n^b n^b}
\end{bmatrix}
$$

(6.11)

and the discretized binary put price can be written as,

$$
P = [P_0, P_1, \ldots, P_{n^a}, P_{n^a+1}, \ldots, P_{n^a+n^b}]' = 
\begin{bmatrix}
P_0 \\
P_1 \\
\vdots \\
P_{n^a} \\
P_{n^a+1} \\
\vdots \\
P_{n^a+n^b}
\end{bmatrix}
$$

(6.12)

Therefore, the discretized weak matrix formulation is,

$$
\begin{bmatrix}
M_{00}^a & \cdots & M_{0n^a}^a \\
\vdots & \ddots & \vdots \\
M_{n^a 0}^a & \cdots & M_{n^a n^a}^a + M_{00}^b & \cdots & M_{0n^b}^b \\
\vdots & \ddots & \vdots \\
M_{n^b 0}^b & \cdots & M_{n^b n^b}^b
\end{bmatrix}
\begin{bmatrix}
\frac{\partial P_0}{\partial t} \\
\vdots \\
\frac{\partial P_{n^a}}{\partial t} \\
\vdots \\
\frac{\partial P_{n^a+n^b}}{\partial t}
\end{bmatrix}
$$

(6.13)
The “middle” row of the above matrix form is consistent with and is the consequence of the construction of the “middle” global basis function.

For the binary option, the final payoff is discontinuous, but anytime before maturity, the option price is not only continuous but also smooth. Therefore, as discussed, for the time integration before the maturity, the exact same procedure of the ordinary option can be implemented for the binary option. The Crank-Nicolson time discretization method before maturity will be implemented as follows,

$$\frac{M^{k+1} - P^k}{\delta t} = S \frac{P^{k+1} + P^k}{2}$$  \hspace{1cm} (6.14)

therefore, the put binary option price can work backwards from $t_{k+1}$ to $t_k$ for any $t_{k+1}$ less than the maturity time $T$. The backward procedure is as follows,

$$P^k = \left( I + \frac{\delta t}{2} M^{-1} S \right)^{-1} \left( I - \frac{\delta t}{2} M^{-1} S \right) P^{k+1}$$  \hspace{1cm} (6.15)

where $\delta t = t_{k+1} - t_k$. The difficulty lies on how to derive $P$ immediate before $T$.

Because of the discontinuous payoff at the strike at the maturity, the Crank-Nicolson method in (6.14) needs some special treatment. $P_{l}(T)$ will be denoted as the final payoff at the immediate left to the strike price and $P_{r}(T)$ will be the final payoff at the immediate right to the strike price. For the standard European cash-or-nothing binary put option, the final payoff $P(T)$ at maturity is,

$$P(T) = \begin{cases} 
1 & \text{if } S(T) < K \\
0 & \text{if } S(T) \geq K 
\end{cases}$$  \hspace{1cm} (6.16)

then $P_{l}(T) = 1$ and $P_{r}(T) = 0$. The left element is only smooth when $P^{a}(T) = P_{l}(T)$ and the right element is only smooth when $P^{b}(T) = P_{r}(T)$, therefore, $P_{l}(T)$ will be used for the left element, and $P_{r}(T)$ will be used for the right element. In order to appropriately allocate the left and right final payoffs at the strike price, a left final payoff vector $P_{l}(T)$ and a right final payoff vector $P_{r}(T)$ are defined as follows,
and a left global mass matrix $M_\ell$ and a right global one $M_r$ are defined as follows,

$$
M_\ell = \begin{bmatrix}
M_{00}^a & \cdots & M_{0n^a}^a \\
\vdots & \ddots & \vdots \\
0 & \cdots & 0
\end{bmatrix} \quad (6.18)
$$

$$
M_r = \begin{bmatrix}
0 & \cdots & 0 \\
\vdots & \ddots & \vdots \\
M_{n^b0}^b & \cdots & M_{n^b n^b}^b
\end{bmatrix} \quad (6.19)
$$

Then the matrix multiplication, $MP$, in the weak formulation for the final time at maturity, $T$, should be formulated correctly as follows,

$$
M_\ell P_\ell + M_r P_r \quad (6.20)
$$

In this way, $P_\ell(T)$ and $P_r(T)$ will be finely assigned to the left and the right element respectively, as shown in (6.21), while the other points will be intact.
Similarly, the matrix multiplication, $SP$, in the weak formulation for the final time at maturity, $T$, should be formulated correctly as follows,

$$S_{\ell}P_{\ell} + S_{r}P_{r} = \begin{bmatrix} M_{00}^a & \cdots & M_{0n^a}^a \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 0 \end{bmatrix} \begin{bmatrix} P_0 \\ \vdots \\ P_{n^a+n^b} \end{bmatrix}$$

Similarly, the matrix multiplication, $SP$, in the weak formulation for the final time at maturity, $T$, should be formulated correctly as follows,

$$S_{\ell}P_{\ell} + S_{r}P_{r} = \begin{bmatrix} 0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 0 \end{bmatrix} \begin{bmatrix} P_0 \\ \vdots \\ P_{n^a+n^b} \end{bmatrix}$$

where the left global stiffness matrix $S_{\ell}$ and the right one $S_{r}$ are defined as follows,

$$S_{\ell} = \begin{bmatrix} S_{00}^a & \cdots & S_{0n^a}^a \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 0 \end{bmatrix}$$

and

$$S_{r} = \begin{bmatrix} 0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ S_{n^b0}^b & \cdots & S_{n^b+n^b}^b \end{bmatrix}$$

Similarly, the matrix multiplication, $SP$, in the weak formulation for the final time at maturity, $T$, should be formulated correctly as follows,
In this way, \( P_\ell(T) \) and \( P_r(T) \) will be finely assigned to the left and the right element respectively as well, while the other points will be intact too.

Therefore, the Crank-Nicolson time integration method in (6.14) for the payoffs at the final maturity, that is, \( t_{k+1} = T \), can be written as follows,

\[
\frac{M_\ell P_\ell + M_r P_r - MP^k}{\delta t} = \frac{S_\ell P_\ell + S_r P_r + SP^k}{2}
\]  

(6.25)

then, the option price, \( P^k \), which is immediately before the maturity time, \( T \), can be computed backwards from the carefully constructed “final payoff”, as follows,

\[
P^k = \left( M + \frac{\delta t}{2} S \right)^{-1} \left[ \left( M_\ell - \frac{\delta t}{2} S_\ell \right) P_\ell + \left( M_r - \frac{\delta t}{2} S_r \right) P_r \right]
\]  

(6.26)

where \( \delta t = T - t_k \). Since there is a special situation here, that is, the option price is discontinuous at the maturity, while it is smooth right before the maturity, the \( \delta t \) here should be set as small as possible to achieve the high accuracy for the option prices before maturity. For example, \( \delta t \) can set to be \( 10^{-10} \) or smaller depending on the accuracy of the computing environment. As long as, the \( P^k \) immediate before the maturity is available, the option prices further before including the initial option price can be calculated backwards normally using the regular one in (6.15).

The initial put option price is \( P^0 \). As long as the put option price at the global grid points are available, the put option price at any arbitrary points (arbitrary stock prices) can be interpolated (expanded) by the global basis functions. Or equivalently, the put option price at each element can be interpolated with the local basis functions with the put option prices at local grid points, as follows,

\[
P^a(S^a) = P^a(S^a(x^a)) = P^a(x^a) = \sum_{i=0}^{n^a} \phi^a_i(x^a) P^a(x^a_i)
\]  

(6.27)

\[
P^b(S^b) = P^b(S^b(x^b)) = P^b(x^b) = \sum_{i=0}^{n^b} \phi^b_i(x^b) P^b(x^b_i)
\]  

(6.28)
6.3.2 SEM Option Greeks

The option sensitivities, Delta (\(\Delta\)), Gamma (\(\Gamma\)) and Theta (\(\Theta\)) can be directly computed from the multi-element spectral element method. The option Delta and Gamma at the local grid points will first be computed, then the Greeks at any other local points can be interpolated by the local basis functions in each element. The Delta and Gamma at the strike price can either calculated from the left (first) element or the right (second) element, which will be exponentially close, i.e.,

\[
\Delta^a_{n_a} = \frac{1}{J_a} \sum_{j=0}^{n_a} D^a_{n_a} P^a(x^a_j) = \frac{1}{J_b} \sum_{j=0}^{n_b} D^b_{0} P^b(x^b_j) = \Delta^b_{0} \tag{6.29}
\]

\[
\Gamma^a_{n_a} = \frac{1}{J_a} \sum_{j=0}^{n_a} B^a_{n_a} P^a(x^a_j) = \frac{1}{J_b} \sum_{j=0}^{n_b} B^b_{0} P^b(x^b_j) = \Gamma^b_{0} \tag{6.30}
\]

While the Theta at the global grid points can directly computed from the global weak matrix formulation, the Theta at any other points can be interpolated locally by using the corresponding local basis functions.

6.3.3 Numerical Results

A European binary cash-or-nothing put option with the same parameters as before is computed using the proposed spectral element method (SEM), which are,

- Strike Price: \(K = 40.0\)
- Interest Rate: \(r = 5.0\%\)
- Stock Volatility: \(\sigma = 25\%\)
- Option Maturity: \(T = 0.5\) yr

For illustration purpose, the final payoff will be scaled from $1 to the strike price of \(K\). Two-element SEM using the GLL and stable GRL quadrature and their basis functions with Galerkin-NI approach is applied to the European binary put option. The first element uses the GLL quadrature and its basis functions, and the second uses the stable GRL quadrature and its basis functions.
As in Section 5.2.4, for the first element, the order of the GLL basis functions is set to 40, i.e., the total number of grid points is 41 for the first element, and for the second element, the order of the stable GRL basis functions is set to 20, i.e., the total number of grid points is 21 for the second element. There will be total \(41 + 21 - 1 = 61\) grid points. The time step of about \(10^{-5}\) year is implemented for the two-element SEM here and the \(\delta t\) is set about \(10^{-11}\) for the final payoff.

The computed binary put option surface over time-to-maturity and possible traded stock price is illustrated in Figure 6.13. The comparison of the calculated binary put option price with the analytical price from the closed-form solution is plotted in Figure 6.14. The errors for the current binary put option price, \(P(t = 0)\), are very impressive with \(L_2\)-Error = \(7.2110 \times 10^{-12}\) and \(L_\infty\)-Error = \(1.4378 \times 10^{-11}\).

Because of the nature of the Gauss-Lobatto-Legendre (GLL) and stable Gauss-Radau-Laguerre (GRL) quadratures as the grid points are more dense around \(\pm 1\) for the GLL quadrature and around 0 for the stable GRL quadrature, the Runge phenomenon, is avoided at the boundaries. Consequently, the computed binary put option prices using the two-element SEM do not show large numerical errors around the strike price at any time before the maturity as shown in Figure 6.13.

\(L_2\)-Errors with increasing total number of grid points are graphed in Figure 6.15 with the converge slope \(\alpha = 0.2106\). It does show that the error converges exponentially with the total number of the grid points. Because of the nature of the binary put option, which decays rapidly with increasing stock price, the order number of the GLL basis functions in the first element is set roughly twice the order number of the stable GRL ones in the second element to further increase the convergence rate with the increasing total number of grid points. While \(L_2\)-Errors with respect to the time step are depicted in Figure 6.16. The estimate slope of convergence, \(\beta = 1.9944\), is very close to the theoretical value of 2.00, which means a second-order convergence. There are quiet similar results of \(L_\infty\)-Errors for both Figure 6.15 and Figure 6.16.
Figure 6.13: The standard European binary put prices over time-to-maturity and stock price using two-element spectral element method with GLL and stable GRL.

Figure 6.14: Comparison between analytical closed-form prices and two-element spectral element method prices for the standard European binary put at inception.
Figure 6.15: The $L_2$-Errors for the European binary put option prices with respect to the total number of grid points using the two-element method with $\delta t \doteq 10^{-5}$ year.

Figure 6.16: The $L_2$-Errors for the European binary put option prices with respect to time step using the two-element method with the total number of grid points $= 61$. 
In the numerical results for the Greeks, the same option parameters will be used for computing the binary put option Greeks, specifically Delta ($\Delta$), Gamma ($\Gamma$) and Theta ($\Theta$). The order of the Gauss-Lobatto-Legendre (GLL) basis functions of the first element is set to 40, i.e., the total number of grid points is 41 for the first element, and the order of the stable Gauss-Radau-Laguerre (GRL) basis functions of the second element is set to 20, i.e., the total number of grid points is 21 for the second element. Therefore there will be total 61 grid points. Same order number construction will be used for all the Greeks. The time step is set to about $10^{-5}$ year and the $\delta t$ is set about $10^{-11}$ for the final payoff to increase the accuracy.

The comparison of the calculated binary put option Delta with the analytical Delta from the closed-form solution is plotted in Figure 6.17. The errors for the current put option Delta, $\Delta(t = 0)$, are very impressive with $L_2$-Error = $2.5445 \times 10^{-10}$ and $L_\infty$-Error = $8.2615 \times 10^{-10}$. Because the nature of the spectral element method (SEM), the derivatives (the Greeks) of the original solution ($P$) will be less accurate than the function itself, which is especially true for the second derivative, Gamma ($\Gamma$), though all are accurate enough for practical uses. As discussed theoretically, the Deltas at the boundary between the two element are indeed exponentially close with a relative error of $1.3464 \times 10^{-11}$.

With the same allocation mechanism of order numbers for the two elements as in computing the binary option price itself, the $L_2$-Errors for the Delta ($\Delta$) with increasing total number of grid points are graphed in Figure 6.18 with $\alpha = 0.1818$, which is slightly slower than $\alpha = 0.2106$ for the binary put option price in Figure 6.15, as expected. While the $L_2$-Errors with respect to the time step are depicted in Figure 6.19. The estimate slope of error convergence, $\beta = 1.9940$, is very close to the theoretical value of 2.00, which means a second-order convergence as described in Section 4.4. There are quiet similar results of $L_\infty$-Errors for both Figure 6.18 and Figure 6.19, which are not shown repetitively here.
Figure 6.17: Comparison between analytical Deltas and the two-element spectral element method Deltas for the standard European binary put option at the inception.

Figure 6.18: The $L_2$-Errors for the European binary put Deltas with respect to the total number of the grid points using two-element method with $\delta t = 10^{-5}$ year.
The comparison of the calculated binary put option Gamma with the analytical Gamma is plotted in Figure 6.20. The errors for the current put Gamma, $\Gamma(t = 0)$, are very impressive with $L_2$-Error $= 5.9503 \times 10^{-9}$ and $L_\infty$-Error $= 4.3415 \times 10^{-8}$. As discussed theoretically, the Gammas at the boundary between the two element are indeed exponentially close with a relative error of $1.1499 \times 10^{-8}$.

With the same allocation mechanism of order numbers for the two elements as before, the $L_2$-Errors for the Gamma ($\Gamma$) with increasing total number of grid points are graphed in Figure 6.21 with $\alpha = 0.1458$, which is slightly slower than $\alpha = 0.1818$ for the binary put option Delta in Figure 6.18, as expected. While the $L_2$-Errors with respect to the time step are depicted in Figure 6.22. The estimate slope of error convergence, $\beta = 1.9945$, is very close to the theoretical value of 2.00, which means a second-order convergence as described in Section 4.4. There are quiet similar results of $L_\infty$-Errors for both Figure 6.21 and Figure 6.22.
Figure 6.20: Comparison between analytical closed-form Gammas and two-element spectral element method Gammas for the European binary put option at inception.

Figure 6.21: The $L_2$-Errors for the European binary put Gammas with respect to the total number of grid points using the two-element method with $\delta t = 10^{-5}$ year.
Figure 6.22: The $L_2$-Errors for the European binary put Gammas with respect to the time step using two-element method with the total number of grid points = 61.

The comparison of the calculated binary put option Theta with the analytical Theta is plotted in Figure 6.23. The errors for the current put Theta, $\Theta(t = 0)$, are very impressive with $L_2$-Error $= 2.4205 \times 10^{-10}$ and $L_{\infty}$-Error $= 3.6639 \times 10^{-10}$. Because the binary put option Thetas at all the grid points are directly computed from the matrix weak formulation of the Black-Scholes partial differential equation (PDE) for the multi-element spectral element method (ME-SEM), there are no two distinct values at the boundary between two elements.

With the same allocation mechanism of order numbers for the two elements as before, the $L_2$-Errors for the binary put option Theta ($\Theta$) with increasing total number of grid points are graphed in Figure 6.24 with $\alpha = 0.1880$. While the $L_2$-Errors with respect to the time step are depicted in Figure 6.25. The estimate slope of error convergence, $\beta = 1.9960$, is very close to the theoretical value of 2.00, which means a second-order convergence as described in Section 4.4. There are quiet similar results of $L_{\infty}$-Errors for both Figure 6.24 and Figure 6.25.
Figure 6.23: Comparison between analytical closed-form Thetas and two-element spectral element method Thetas for the European binary put option at the inception.

Figure 6.24: The $L_2$-Errors for the European binary put Thetas with respect to the total number of grid points using the two-element method with $\delta t = 10^{-5}$ year.
Figure 6.25: The $L_2$-Errors for the European binary put Thetas with respect to time step using the two-element method with the total number of grid points = 61.

6.4 Comparison of Methods

In the last two sections, an approximation spectral element method and a more accurate discontinuous payoff spectral element method (DP-SEM) are proposed and numerically verified with impressive results. The DP-SEM method is the best way to implement the spectral element method for options with discontinuous payoff. The discontinuous payoff spectral element method is first-time discovered and introduced, which is named as DP-SEM method.

The most simple way to compute the binary put option with the SEM is to use the original payoff and the same mass and stiffness matrix as in the regular put option. However, the original payoff has a gap at the strike. An average method is used to “smooth” the payoff [66, 63]. That is, the payoff at the strike will be $1/2$, the average of the “left” payoff, 1, and the “right” payoff, 0, at the strike. The comparison of the convergence rates of the four methods are shown in Figure 6.26.
Figure 6.26: Comparison of $L_2$-Errors of four different spectral element methods for the binary put option price using the two-element method with $\delta t = 10^{-5}$ year.

Figure 6.27: Comparison of $L_2$-Errors of DP-SEM and approximation spectral element method for the binary put using the two-element method with respect to $\delta t$.  

212
7

SEM on Constant Elasticity of Variance (CEV)

7.1 Model Introduction

The constant elasticity of variance (CEV) model is a local volatility model, which provides an alternative to the Black-Scholes option model to accommodate the observed volatility skews in the markets. The model was first developed by John C. Cox in an unpublished note in 1975 [67], which was reprinted in 1996 [68].

The asset price under the constant elasticity of variance (CEV) local volatility model has the following diffusion process,

$$dS_t = \mu S_t dt + \sigma S_t^\beta dW_t, \quad \beta > 0$$

(7.1)

where $\mu$ is the constant drift, $\sigma$ is the constant volatility parameter and $\beta$ is the constant elasticity parameter. The “instantaneous volatility”, $\nu(S, t)$ is,

$$\nu(S, t) = \sigma S^{\beta-1}$$

(7.2)

and the elasticity of return variance equals $\beta-1$ [69, 70]. Empirical evidence generally better supports the CEV model than the Black-Scholes model [71, 72]. When $\beta = 1$, the CEV model degenerates to the lognormal model, i.e., the GBM model, as in the
Black-Scholes formula. When $\beta < 1$, the elasticity is negative, i.e., the volatility and the price are inversely related. This is the leverage effect commonly observed in the equity markets [73]. When $\beta > 1$, the elasticity is positive, i.e., the volatility and the price are positively correlated. This is so-called inverse leverage effect exists in the commodity markets [74]. The volatility curve experiences either flat, downward or upward with respect to the asset price with different levels of the elasticity.

7.2 Derivation of PDE

In the CEV world, the asset price follows the stochastic differential equation (SDE),

$$dS(t) = \mu S(t)dt + \sigma S(t)^\beta dW(t), \quad \beta > 0$$  (7.3)

where $\mu$ is the drift of the asset and $\sigma$ is the volatility parameter, both of which are constant. The interest rate $r$ shown below will be assumed constant too.

Similarly as in Section 2.1, longing one call and selling an appropriate number of shares, $\Delta(t)$, will make the following portfolio instantaneously risk-free,

$$V(t) = C(S, t) - \Delta(t)S(t)$$  (7.4)

Using Itô’s Lemma and because of self-financing condition, then

$$dV(t) = dC(S, t) - \Delta(t)dS(t)$$

$$= C_t dt + C_SdS + \frac{1}{2}C_{SS}dSdS - \Delta(t)dS(t)$$

$$= C_t dt + C_S\mu S dt + C_S\sigma S^\beta dW + \frac{1}{2}C_{SS}\sigma^2 S^{2\beta}dt - \Delta(t)dS(t)$$

$$= \left[\frac{1}{2}\sigma^2 S^{2\beta}C_{SS} + \mu S(t)C_S + C_t\right]dt$$

$$+ \sigma S^\beta C_S dW(t) - \Delta(t)\left[\mu S(t)dt + \sigma S^\beta dW(t)\right]$$

$$= \left[\frac{1}{2}\sigma^2 S^{2\beta}C_{SS} + \mu S(t)C_S + C_t - \Delta(t)\mu S(t)\right]dt$$

$$+ \left[\sigma S^\beta C_S - \Delta(t)\sigma S^\beta\right]dW(t)$$  (7.5)
Eliminating the diffusion term because of the risk-free portfolio, then,

\[ \sigma S(t)^\beta C_S - \Delta(t)\sigma S(t)^\beta = 0 \quad \Rightarrow \quad C_S = \Delta(t) \]  

\text{(7.6)}

therefore,

\[ dV(t) = \left[ \frac{1}{2} \sigma^2 S^{2\beta}(t)C_{SS} + C_t \right] dt \]  

\text{(7.7)}

Because of absence of arbitrage,

\[ dV(t) = rV(t)dt = r[C - S(t)C_S]dt \]  

\text{(7.8)}

therefore,

\[ \frac{1}{2} \sigma^2 S^{2\beta}(t)C_{SS} + C_t = r[C - S(t)C_S] \]  

\text{(7.9)}

Then, the partial differential equation (PDE) for the CEV model becomes,

\[ \frac{1}{2} \sigma^2 S^{2\beta}(t)C_{SS} + rS(t)C_S + C_t - rC = 0 \]  

\text{(7.10)}

with a set of final and boundary conditions,

\[ C(S, T) = \max\{S(T) - K, 0\} \]  

\text{(7.11)}

\[ C(0, t) = 0, \quad \forall \ 0 \leq t \leq T \]  

\text{(7.12)}

\[ C(S, t) = S, \quad \text{as} \quad S \to \infty \]  

\text{(7.13)}

Similarly, a European put option price \( P \) follows the same partial differential equation (PDE) with the constant elasticity of variance (CEV) model,

\[ \frac{1}{2} \sigma^2 S^{2\beta}(t)P_{SS} + rS(t)P_S + P_t - rP = 0 \]  

\text{(7.14)}

with a set of following final and boundary conditions,

\[ P(S, T) = \max\{K - S(T), 0\} \]  

\text{(7.15)}

\[ P(0, t) = Ke^{-r(T-t)}, \quad \forall \ 0 \leq t \leq T \]  

\text{(7.16)}

\[ P(S, t) = 0, \quad \text{as} \quad S \to \infty \]  

\text{(7.17)}
7.3 SEM on CEV PDE

In this section, the spectral element method (SEM) is applied to solve the partial differential equation (PDE) under the constant elasticity of variance (CEV) local volatility model for European put-style options. Then, the weak (variational) formulation of the partial differential equation is,

$$\int_0^\infty \left[ \frac{1}{2} \sigma^2 S^{2\beta}(t)P_{SS} + rS(t)P_S + P_t - rP \right] v(S) dS = 0 \quad (7.18)$$

where $v(S)$ is a test function in the spectral element method (SEM). Using integration by parts, the integration of the diffusion term (the second derivative term) in the weak formulation (7.18) becomes,

$$\int_0^\infty \frac{1}{2} \sigma^2 S^{2\beta}(t)P_{SS}v(S) dS$$

$$= \frac{1}{2} \sigma^2 \int_0^\infty S^{2\beta}v(S) dP_S = \frac{1}{2} \sigma^2 S^{2\beta}v(S) \bigg|_0^\infty - \frac{1}{2} \sigma^2 \int_0^\infty P_S d[S^{2\beta}v(S)]$$

$$= \frac{1}{2} \sigma^2 S^{2\beta}v(S) \bigg|_0^\infty - \beta \sigma^2 \int_0^\infty P_S S^{2\beta-1}v(S) dS - \frac{1}{2} \sigma^2 \int_0^\infty P_S S^{2\beta}v_S dS \quad (7.19)$$

Then the weak formulation (7.18) of the partial differential equation becomes,

$$\frac{1}{2} \sigma^2 S^{2\beta}v(S) \bigg|_0^\infty + \int_0^\infty (rS - \beta \sigma^2 S^{2\beta-1}) P_S v(S) dS$$

$$- \frac{1}{2} \sigma^2 \int_0^\infty P_S S^{2\beta}v_S dS + \int_0^\infty P_t v(S) dS - r \int_0^\infty P(S) v(S) dS = 0 \quad (7.20)$$

For the purpose of illustration, $0.5 \leq \beta < 1$ is assumed, i.e., the process is between square-root process \[75\] and lognormal process. Under this assumption, the asset price experiences a leverage effect, that is, when the underlying price goes up, the volatility goes down. Consequently, the put-style option will converge to zero
rapidly when the price goes up. Therefore the stable Gauss-Radau-Laguerre (GRL) basis functions, which exponentially decay, can be used for the last element as in the previous chapters. The Gauss-Lobatto-Legendre (GLL) basis functions will be used for the other element(s). Moreover, with assumption of $\beta \geq 0.5$, the term $\beta \sigma^2 S^{2\beta-1}$ in the weak formulation will either stays at one ($\beta = 0.5$) or goes to zero ($\beta > 0.5$) when the asset price $S$ goes to zero. Hence, the boundary term will vanish as in the Black-Scholes PDE and the integrations in the weak formulation are “well-defined”, which can be properly solved by the Gauss quadrature rules. For other possible scenarios of the elasticity parameter $\beta$, careful boundary condition enforcement and proper selection of basis functions are needed for the implementation of the spectral element method (SEM). Then, in the weak formulation, the mass matrix and stiffness matrix in each element can be formulated as follows,

$$M_{ji} = J \int_{\Omega} \phi_i(x)\phi_j(x)dx$$

(7.21)

$$S_{ji} = -\int_{\Omega} \left[ rS(x) - \beta \sigma^2 S(x)^{2\beta-1} \right] \frac{\partial \phi_i(x)}{\partial x} \phi_j(x)dx$$

$$+ \frac{1}{2J} \sigma^2 \int_{\Omega} S(x)^{2\beta} \frac{\partial \phi_i(x)}{\partial x} \frac{\partial \phi_j(x)}{\partial x} dx$$

$$+ rJ \int_{\Omega} \phi_i(x)\phi_j(x)dx$$

(7.22)

The global mass matrix and stiffness matrix can be constructed in the same way as in solving the Black-Scholes partial differential equation (PDE) using the spectral element method (SEM). The integration terms in the mass and stiff matrices can be computed using the associated quadratures as follows,

$$\int_{\Omega} \phi_i(x)\phi_j(x)dx = \sum_{k=0}^{n} w_k \phi_i(x_k)\phi_j(x_k) = w_i \delta_{ij}$$

(7.23)
\[
\int_{\Omega} \left[ rS(x) - \beta \sigma^2 S(x)^{2\beta - 1} \right] \frac{\partial \phi_i(x)}{\partial x} \phi_j(x) \, dx \\
= \sum_{k=0}^{n} w_k \left[ rS(x_k) - \beta \sigma^2 S(x_k)^{2\beta - 1} \right] \frac{\partial \phi_i(x_k)}{\partial x} \phi_j(x_k) \\
= w_j D_{ji} \left[ rS(x_j) - \beta \sigma^2 S(x_j)^{2\beta - 1} \right] 
\]  
(7.24)

\[
\int_{\Omega} S(x)^{2\beta} \frac{\partial \phi_i(x)}{\partial x} \frac{\partial \phi_j(x)}{\partial x} \, dx = \sum_{k=0}^{n} w_k S(x_k)^{2\beta} \frac{\partial \phi_i(x_k)}{\partial x} \frac{\partial \phi_j(x_k)}{\partial x} \\
= \sum_{k=0}^{n} w_k D_{ki} D_{kj} S(x_k)^{2\beta} 
\]  
(7.25)

Exact same procedures as in the spectral element method (SEM) on the Black-Scholes partial differential equation (PDE) except the above revised mass and stiffness matrices can be implemented to solve European put-style options under the CEV model with the elasticity parameter $\beta$ between 0.5 and 1.

For a European binary cash-or-nothing put option under the CEV model with $0.5 \leq \beta < 1$, there is a valuation formula as follows [36, 69],

\[
P(S, t) = e^{-r(T-t)} \left[ 1 - \chi^2(\xi; \theta, \lambda) \right] 
\]  
(7.26)

where

\[
\theta = (1 - \beta)^{-1} 
\]  
(7.27)

\[
\xi = \frac{\theta^2}{\sigma^2} \frac{2r(1 - \beta)}{1 - e^{-2r(1 - \beta)(T-t)}} S^{2(1-\beta)} 
\]  
(7.28)

\[
\lambda = \frac{\theta^2}{\sigma^2} \frac{2r(1 - \beta)}{1 - e^{-2r(1 - \beta)(T-t)}} \left[ Ke^{-r(T-t)} \right]^{2(1-\beta)} 
\]  
(7.29)

and $\chi^2(\xi; \theta, \lambda)$ is the cumulative distribution function (CDF) of a noncentral chi-squared distribution with $\theta$ degrees of freedom and noncentrality parameter $\lambda$.  

218
7.4 Numerical Results

A European binary cash-or-nothing put option under the CEV model is computed using the spectral element method (SEM) with the following option parameters,

\[
\begin{align*}
\text{Strike Price:} & \quad K = 40.0 \\
\text{Interest Rate:} & \quad r = 5.0\% \\
\text{Volatility Parameter:} & \quad \sigma = 75\% \\
\text{Elasticity Parameter:} & \quad \beta = 0.75 \\
\text{Option Maturity:} & \quad T = 0.5 \text{ yr}
\end{align*}
\]

For illustration purpose, the final payoff will be scaled from $1 to the strike price of $K and the stock volatility parameter $\sigma$ will be set as high as 75%. First, the binary put option surface and prices before changing the elasticity parameter and becoming a real CEV process, i.e., $\beta = 1.00$, are shown in Figure 7.1 and Figure 7.2. The rest figures are shown under the CEV model with $\beta = 0.75$.

The option surface with respect to the stock price and time-to-maturity is plotted in Figure 7.3 with $\beta = 0.75$. While Figure 7.4 shows the SEM option prices under the CEV model, which are compared with the computation results from the Monte Carlo method and the standard European binary put under the lognormal process. The $L_2$-Error is $5.9131 \times 10^{-4}$ comparing with the MCM. In the MCM, the $L_2$-Error is $5.1231 \times 10^{-4}$ itself, or equivalently, the mean standard error is about $0.01$.

Figure 7.5 illustrates the option price again from the SEM, which is compared with the analytical solution with $L_2$-Error $= 6.4153 \times 10^{-12}$ and $L_\infty$-Error $= 9.7257 \times 10^{-12}$. While Figure 7.6 shows the option price error convergence with respect to the number of grid points and Figure 7.7 depicts the errors with respect to the time step, where one is exponential convergence and the other is second-order convergence. The Greeks, Delta, Gamma and Theta, are graphed in Figure 7.8, Figure 7.9 and Figure 7.10, where the analytical values are omitted.
Figure 7.1: The European binary put option prices over time-to-maturity and stock price using the two-element spectral element method under the lognormal model.

Figure 7.2: Comparison of option prices between the analytical values and the spectral element method for the European binary put under the lognormal model.
Figure 7.3: The European binary put option prices over time-to-maturity and stock price using the two-element spectral element method under the CEV with $\beta = 0.75$.

Figure 7.4: Comparison of option prices between SEM and MCM for the European binary put under the CEV with $\beta = 0.75$ along with the standard binary put price.
Figure 7.5: Comparison of option prices between the analytical values and the spectral element method for the European binary put under the CEV with $\beta = 0.75$.

Figure 7.6: The $L_2$-Errors for the European binary put option prices with respect to the total number of grid points under the CEV with $\beta = 0.75$ and $\delta t = 10^{-5}$ year.
Figure 7.7: The $L_2$-Errors for the European binary put option prices with respect to time step under the CEV with $\beta = 0.75$ and the total number of grid points = 61.

Figure 7.8: The computed Deltas from the spectral element method for the European binary put option under the CEV model with $\beta = 0.75$ and $\delta t = 10^{-5}$ year.
Figure 7.9: The computed Gammas from the spectral element method for the European binary put option under CEV model with $\beta = 0.75$ and $\delta t = 10^{-5}$ year.

Figure 7.10: The computed Thetas from the spectral element method for the European binary put option under CEV model with $\beta = 0.75$ and $\delta t = 10^{-5}$ year.
The Stochastic Alpha Beta Rho (SABR) Model

8.1 Model Introduction

The Black-1976 model, a variant of the Black-Scholes formula, has been the benchmark model for European options on futures and forwards [76]. The Black formula for European call options on forwards is as follows [36],

\[
C(S, t) = e^{-r(T-t)} \left[ F(t) \mathcal{N}(d_1) - KN(d_2) \right]
\]

(8.1)

where \( F(t) \) is the forward price at time \( t \), and,

\[
d_1 = \frac{\ln \frac{F(t)}{K} + \frac{1}{2} \sigma^2 (T-t)}{\sigma \sqrt{T-t}}
\]

(8.2)

\[
d_2 = d_1 - \sigma \sqrt{T-t}
\]

(8.3)

and for put options on forwards is,

\[
P(S, t) = e^{-r(T-t)} \left[ KN(-d_2) - F(t) \mathcal{N}(-d_1) \right]
\]

(8.4)

with same \( d_1 \) and \( d_2 \) in the closed formula for call options.
The Black model has a main drawback of the constant volatility assumption, and at most, the model assumes a deterministic time varying volatility [77]. The model fails to capture the dynamic behavior of the option skewness, which leads to possible incorrect hedges. In 2002, Hagan et al. proposed a stochastic volatility model which attempts to capture the volatility smile, named as stochastic alpha beta rho (SABR) model [78]. The dynamics of the SABR model is as follows,

\[ dF(t) = \alpha(t) F(t)^{\beta} dW(t) \]  \hspace{1cm} (8.5)

\[ d\alpha(t) = \nu \alpha(t) dZ(t) \]  \hspace{1cm} (8.6)

\[ dW(t)dZ(t) = \rho dt \]  \hspace{1cm} (8.7)

where \( \alpha(t) \) is the initial volatility of the forward price, \( \beta \) is the exponent parameter as in the CEV model [79], \( \nu \) is a constant “volatility of volatility” parameter, and \( dW \) and \( dZ \) are two correlated Wiener processes with correlation of \( \rho \) [77].

Hagan et al., in their original paper [78], uses singular perturbation techniques and obtains an asymptotic solution for an implied volatility, \( \sigma_B \), for the Black-1976 formula [80], which is named as Black equivalent volatility as in [81]. The \( \sigma_B \) is,

\[ \sigma_B = \frac{\alpha}{(FK)^{(1-\beta)/2}} \left( 1 + \frac{(1-\beta)^2}{24} \left( \ln \frac{F}{K} \right)^2 + \frac{(1-\beta)^4}{1920} \left( \ln \frac{F}{K} \right)^4 \right) \times \frac{z}{\chi(z)} \]

\[ \times \left[ 1 + \left( \frac{(1-\beta)^2}{24} \frac{\alpha^2}{(FK)^{1-\beta}} + \frac{1}{4} \frac{\rho \beta \nu \alpha}{(FK)^{1-\beta/2}} + \frac{2 - 3 \rho^2}{24} \nu^2 \right) (T-t) \right] \]  \hspace{1cm} (8.8)

where

\[ z = \frac{\nu}{\alpha} (FK)^{(1-\beta)/2} \ln(F/K) \]  \hspace{1cm} (8.9)

\[ \chi(z) = \ln \left( \frac{\sqrt{1 - 2 \rho z + z^2} + z - \rho}{1 - \rho} \right) \]  \hspace{1cm} (8.10)

In other words, after having the approximated \( \sigma_B \), the Black-1976 model can be used to get the “correct” option price by inputting the asymptotic Black equivalent
volatility, $\sigma_B$, into the model. For some special cases, when $\beta = 1.00$, i.e., the lognormal process, the Black equivalent volatility is reduced to [80],

$$
\sigma_B = \alpha \times \frac{z}{\chi(z)} \times \left[ 1 + \left( \frac{\rho \beta \nu \alpha}{4} + \frac{2 - 3 \rho^2}{24 \nu^2} \right) (T - t) \right]
$$

(8.11)

where

$$
z = \frac{\nu}{\alpha} \ln(F/K), \quad \chi(z) = \ln \left( \frac{\sqrt{1 - 2\rho z + z^2} + z - \rho}{1 - \rho} \right)
$$

(8.12)

when $F = K$, i.e., at-the-money, the $\sigma_B$ is reduced to [80],

$$
\sigma_B = \frac{\alpha}{F^{1-\beta}} \times \left[ 1 + \left( \frac{(1 - \beta)^2 \alpha^2}{24} F^{-2-2\beta} + \frac{1}{4} \frac{\rho \beta \nu \alpha}{F^{1-\beta}} + \frac{2 - 3 \rho^2}{24 \nu^2} \right) (T - t) \right]
$$

(8.13)

For $\beta < 1$, there is some approximation improvement to the original formula in [82].

### 8.2 Illustrative Example

The SABR model, a stochastic version of the CEV, is widely used in the fixed-income markets. For the consistency of this dissertation, the SABR model for equity forwards will be explored only, where the interest rate is assumed to be constant.

- **Strike Price:** $K = 40.0$
- **Interest Rate:** $r = 5.0\%$
- **Skew Parameter:** $\beta = 0.80$
- **Option Maturity:** $T = 0.5 \text{ yr}$
- **Correlation Parameter:** $\rho = -0.6$
- **Volatility of Volatility:** $\nu = 0.40$

With the above parameters and using the asymptotic formula (8.8), the European put price surface with respect to the underlying forward $F$ and initial volatility $\alpha$ is plotted in Figure 8.1. Approximated $\sigma_B$ surface is graphed in Figure 8.3 with a clear view of volatility smile. Approximated option prices and implied volatilities for selected initial volatilities are illustrated in Figure 8.2 and Figure 8.4 respectively.
**Figure 8.1**: Approximated European put option price surface over the forward price and the initial volatility under the SABR model using the asymptotic formula.

**Figure 8.2**: Approximated European put prices for selected initial volatilities.
Figure 8.3: Approximated Black equivalent volatility surface over the forward price and the forward initial volatility under SABR model using the asymptotic formula.

Figure 8.4: Approximated Black equivalent volatilities for selected initial vols.
8.3 Derivation of PDE

As introduced in the first section, the dynamics of the stochastic alpha beta rho (SABR) volatility model for a forward price $F(t)$ is as follows,

\[
dF(t) = \alpha(t) F(t)^\beta dW(t) \tag{8.14}
\]

\[
d\alpha(t) = \nu \alpha(t) dZ(t) \tag{8.15}
\]

\[
dW(t)dZ(t) = \rho dt \tag{8.16}
\]

An option $P$ on the forward price $F$ have a growth rate of $r$ under the risk-neutral measure. In other words, the discounted option value $e^{-rt}P$ is martingale and has a zero drift under the risk-neutral measure, that is,

\[
E[e^{-rt}P] = 0 \tag{8.17}
\]

and,

\[
de^{-rt}P = e^{-rt}dP - e^{-rt}rPdt \tag{8.18}
\]

By using Itô’s Lemma,

\[
dP = \frac{\partial P}{\partial t} dt + \frac{\partial P}{\partial F} dF + \frac{\partial P}{\partial \alpha} d\alpha
\]

\[
+ \frac{1}{2} \frac{\partial^2 P}{\partial F^2} dF^2 + \frac{\partial^2 P}{\partial F \partial \alpha} dF d\alpha + \frac{1}{2} \frac{\partial^2 P}{\partial \alpha^2} d\alpha^2 \tag{8.19}
\]

\[
= \frac{\partial P}{\partial t} dt + \frac{\partial P}{\partial F} \alpha F^\beta dW + \frac{\partial P}{\partial \alpha} \nu \alpha dZ
\]

\[
+ \frac{1}{2} \frac{\partial^2 P}{\partial F^2} \alpha^2 F^{2\beta} dt + \frac{\partial^2 P}{\partial F \partial \alpha} \rho \nu \alpha^2 F^\beta dt + \frac{1}{2} \frac{\partial^2 P}{\partial \alpha^2} \nu^2 \alpha^2 dt \tag{8.20}
\]

\[
= \frac{\partial P}{\partial F} \alpha F^\beta dW + \frac{\partial P}{\partial \alpha} \nu \alpha dZ
\]

\[
+ \left[ \frac{\partial P}{\partial t} + \frac{1}{2} \frac{\partial^2 P}{\partial F^2} \alpha^2 F^{2\beta} + \frac{\partial^2 P}{\partial F \partial \alpha} \rho \nu \alpha^2 F^\beta + \frac{1}{2} \frac{\partial^2 P}{\partial \alpha^2} \nu^2 \alpha^2 \right] dt \tag{8.21}
\]
with,
\[ dF \, dF = \alpha^2 F^{2\beta} dt, \quad dF \, d\alpha = \rho \nu \alpha^2 F^{\beta} dt, \quad d\alpha \, d\alpha = \nu^2 \alpha^2 dt \quad (8.22) \]

Since,
\[ de^{-rt} P = e^{-rt}[dP - rP dt] \quad (8.23) \]

and \( e^{-rt} P \) is martingale under the risk-neutral measure, the drift term in \( e^{-rt} P \) disappears, i.e., the drift term becomes zero. The zero drift leads to the partial differential equation (PDE) of the SABR volatility model as follows,
\[ \frac{\partial P}{\partial t} + \frac{1}{2} \alpha^2 F^{2\beta} \frac{\partial^2 P}{\partial F^2} + \rho \nu \alpha^2 F^{\beta} \frac{\partial^2 P}{\partial F \partial \alpha} + \frac{1}{2} \nu^2 \alpha^2 \frac{\partial^2 P}{\partial \alpha^2} - rP = 0 \quad (8.24) \]

The SABR partial differential equation (PDE) is essentially a multi-dimensional PDE. Similar to the SEM on a one-dimensional problem, the two-dimensional PDE can be efficiently solved by the multi-dimensional SEM (MD-SEM)[83, 84].

### 8.4 MD-SEM on SABR

The SABR partial differential equation (PDE) for a European option \( P \) is,
\[ \frac{\partial P}{\partial t} + \frac{1}{2} \alpha^2 F^{2\beta} \frac{\partial^2 P}{\partial F^2} + \rho \nu \alpha^2 F^{\beta} \frac{\partial^2 P}{\partial F \partial \alpha} + \frac{1}{2} \nu^2 \alpha^2 \frac{\partial^2 P}{\partial \alpha^2} - rP = 0 \quad (8.25) \]

For the illustration purpose, \( \beta \) is assumed to be 1.00, i.e., the forward price follows the lognormal process with a stochastic volatility, and the correlation between two Brownian motions is assumed to be negative [85], i.e., \( \rho < 0 \). The option is assumed to decay rapidly to zero when the forward price \( F(t) \) goes to infinity while the \( \alpha(t) \) is fixed. For example, standard European put options under the above contraints generally have such decay properties. Then the SABR PDE is reduced to,
\[
\frac{\partial P}{\partial t} + \frac{1}{2} \alpha^2 F^2 \frac{\partial^2 P}{\partial F^2} + \rho \nu \alpha^2 F \frac{\partial^2 P}{\partial F \partial \alpha} \\
+ \frac{1}{2} \nu^2 \alpha^2 \frac{\partial^2 P}{\partial \alpha^2} - rP = 0
\]

(8.26)

The weak (variational) formulation for the partial differential equation is,

\[
\int_0^\infty \int_0^{\alpha_m} \left[ \frac{\partial P}{\partial t} + \frac{1}{2} \alpha^2 F^2 \frac{\partial^2 P}{\partial F^2} + \rho \nu \alpha^2 F \frac{\partial^2 P}{\partial F \partial \alpha} \\
+ \frac{1}{2} \nu^2 \alpha^2 \frac{\partial^2 P}{\partial \alpha^2} - rP \right] \phi(F) \varphi(\alpha) dFd\alpha = 0 
\]

(8.27)

where \( \phi(F) \varphi(\alpha) \) is the test function, a tensor product of two one-dimensional test functions [61], and \( \alpha_m \) is the maximum initial volatility for the problem to solve. As in one-dimensional spectral element method (SEM), the Galerkin-NI approach will be implemented, therefore \( \phi(F) \varphi(\alpha) \) will be the trial function as well. That is, the option price will be expanded by the same basis functions as follows,

\[
P(F, \alpha) = P(x, \xi) = \sum_{i=0}^{n} \sum_{j=0}^{m} \phi_i(x) \varphi_j(\xi) P(x_i, \xi_j) 
\]

(8.28)

A European put option will be implemented with the MD-SEM here, which has only one non-smooth point (but continuous) in the forward price at maturity, while the option price is smooth in the whole domain with respect to the initial volatility. Therefore the two-dimensional plane by the \( F(t) \) and \( \alpha(t) \) can be separated into two elements, \([0, K] \times [0, \alpha_m] \) and \([K, \infty) \times [0, \alpha_m] \). Consequently, the basis functions will be chosen as GLL×GLL and GRL×GLL for the two elements respectively. The following derivations will be assumed under the global basis functions.

Then, the integration for the second derivative (diffusion) term for \( F(t) \) in the weak formulation can be integrated by parts as follows,
\[
\int_0^\infty \frac{1}{2} \alpha^2 F^2 \frac{\partial^2 P}{\partial F^2} \phi(F) dF \\
= \frac{1}{2} \alpha^2 F^2 \phi(F) P_F \bigg|_0^\infty - \alpha^2 \int_0^\infty P_F \phi(F) dF - \frac{1}{2} \alpha^2 \int_0^\infty P_F^2 \phi_F dF \\
= -\alpha^2 \int_0^\infty P_F \phi(F) dF - \frac{1}{2} \alpha^2 \int_0^\infty P_F^2 \phi_F dF \\
(8.29)
\]

since the boundary term vanishes as in the one-dimensional case. Similarly, the integration for the second derivative (diffusion) term for \( \alpha(t) \) is as follows,

\[
\int_0^{\alpha_m} \frac{1}{2} \nu^2 \alpha^2 \frac{\partial^2 P}{\partial \alpha^2} \varphi(\alpha) d\alpha \\
= \frac{1}{2} \nu^2 \alpha^2 \varphi(\alpha) P_\alpha \bigg|_0^{\alpha_m} - \nu^2 \int_0^{\alpha_m} P_\alpha \alpha \varphi(\alpha) d\alpha - \frac{1}{2} \nu^2 \int_0^{\alpha_m} P_\alpha \alpha^2 \varphi_\alpha d\alpha \\
= -\nu^2 \int_0^{\alpha_m} P_\alpha \alpha \varphi(\alpha) d\alpha - \frac{1}{2} \nu^2 \int_0^{\alpha_m} P_\alpha \alpha^2 \varphi_\alpha d\alpha \\
(8.30)
\]

and the boundary term above is approximated to cease to exist as well. The reason is that when \( \alpha = 0 \), the term is zero precisely, while \( \alpha \to \infty, P_\alpha \to 0 \) [86]. Therefore, similar to the case in [30], the Neumann boundary is approximated to be zero when \( \alpha = \alpha_m \) and \( \alpha_m \) is large enough. Possible more precise boundary conditions at \( \alpha = \alpha_m \) can be implemented, but may incur extra computational cost.

The option price will be discretized both in the forward price \( F \) and in the initial volatility \( \alpha \). The two-dimensional option price mesh will be reshaped to a column vector for the matrix formulation of the weak form in the MD-SEM. The mass matrix and stiffness matrix need to be carefully constructed accordingly. The nodes in the mesh can be traversed either vertically or horizontally first. However, traversing along \( \alpha \) first, i.e., row by row, is preferred for implementing the two-element approach as shown in Figure 8.5. It is because the formation of global mass and stiffness matrices from local mass and stiffness matrices will be much easier.
8.5 Numerical Results

A European put option under the SABR model has the following parameters,

\[
\begin{align*}
\text{Strike Price: } & \quad K = 40.0 \\
\text{Interest Rate: } & \quad r = 5.0\% \\
\text{Skew Parameter: } & \quad \beta = 1.00 \\
\text{Option Maturity: } & \quad T = 0.5 \text{ yr} \\
\text{Correlation Parameter: } & \quad \rho = -0.4 \\
\text{Volatility of Volatility: } & \quad \nu = 0.20
\end{align*}
\]

Using the two-element multi-dimensional spectral element method (MD-SEM), the European put option price surface with respect to the underlying forward price \( F \) and initial forward volatility \( \alpha \) is plotted in Figure 8.6 with \( dt \approx 10^{-3} \) year. The \( L_2 \)-Error of the SEM option prices versus the approximated values for the whole surface is \( 2.6428 \times 10^{-5} \), while the computed SEM prices for selected initial volatilities are illustrated in Figure 8.7 and compared with the values from the asymptotic formula. Figure 8.8 and Figure 8.9 demonstrate the convergence rate with respect to the number of grid points on \( F \) and the time step \( \delta t \).
Figure 8.6: Option price surface from two-element MD-SEM with \((36 + 36) \times 16\) points for two elements in the forward price and the initial volatility respectively.

Figure 8.7: Option prices from two-element MD-SEM with \((36 + 36) \times 16\) points for selected initial forward volatilities with comparison with the asymptotic formula.
Figure 8.8: The MD-SEM $L_2$-Errors for the European put option price with respect to the total number of grid points on the forward price $F$ under the SABR model.

Figure 8.9: The MD-SEM $L_2$-Errors for the European put option price with respect to the time step $\delta t$ under the SABR model with $(36 + 36) \times 16$ points on $F$ and $\alpha$. 

236
Conclusion, Discussion and Further Work

In this dissertation, the crucial techniques for the spectral element method (SEM), Gauss quadrature rules, are summarized and derived in great details. A novel combination of eigenvalue method and the Newton-Raphson method to find accurate grid points is proposed and numerically verified. Spectral differential matrices for stable Gauss-Hermite, stable Gauss-Radau-Laguerre (GRL) and Gauss-Lobatto-Legendre (GLL) basis functions are developed and examined, which are used for the SEM on European option pricing and Greek computation. SEM related numerical techniques and error convergence analysis are thoroughly explored.

The procedures to implement SEMs on Black-Scholes partial differential equations (PDEs) in one-element and multi-element approaches are extensively investigated. The computed results from European put options with and without dividend are compared with the finite difference methods (FDMs) under the same Crank-Nicolson approach for time integration. The results show that the SEMs are more accurate and efficient than the FDMs. A European condor option is solved with a five-element SEM with high accuracy despite the complicated final payoff, which demonstrates the flexibility of the multi-element SEM (ME-SEM).
The innovative discontinuous payoff SEM (DP-SEM) is theoretically developed and numerically validated on a European binary put option. The computational accuracy is comparable to a regular put option. The PDE under the constant elasticity of variance (CEV) local volatility model is derived. The SEM CEV results on a binary put option are compared with the Monte Carlo method (MCM) and the valuation formula. The stochastic alpha beta rho (SABR) volatility model is introduced and its PDE is derived. The multi-dimensional SEM (MD-SEM) is carefully developed and realized on the SABR model. The computed SABR results on a put option are compared with the values from the asymptotic formula.

The numerical solutions for the option Greeks (Delta, Gamma and Theta) under the spectral element method (SEM) are developed and applied. The error converge properties for both option prices and option Greeks with respect to the total number of grid points and the time step are studied and illustrated, which show exponential convergence for the spectral element method (SEM) and second-order convergence for the Crank-Nicolson method for the time integration.

In summary, SEMs have been researched to solve a variety of key option pricing problems, for example, with and without dividend, continuous payoff and discontinuous payoff, constant and stochastic volatility, one-dimensional and multi-dimensional. The one-element and multi-element approaches for the method itself are compared as well. The techniques developed in this dissertation provide some basic elements for the general European option pricing, where a complicated option can possibly be solved by combining the key elements presented.

For further work, spectral element methods can be explored for arithmetic Brownian motions [87], moving boundaries problems [88], higher dimensional options [89], other path-dependent options, such as Asian options [90] and barrier options [91], and non-linear PDEs [92]. Spectral element methods provide practical and efficient techniques for solving financial partial differential equations.
Bibliography


method for the solution of partial differential equations on infinite domains:
application to the diffusion of tumour angiogenesis factors. Mathematical and


[56] E. Yakimiw. Accurate computation of weights in classical Gauss-Christoffel
1996.

[57] Tao Tang. The Hermite spectral method for Gaussian-type functions. SIAM

flow in a channel expansion. Journal of Computational Physics, 54(3):468–488,
June 1984.


[63] Hongjoong Kim and Kyoung-Sook Moon Moon. Variable time-stepping hybrid
finite difference methods for pricing binary options. Bulletin of the Korean


[65] Rolf Rannacher. Finite element solution of diffusion problems with irregular


Biography

Tianyao Yue was born in 1978 in China. He received a B.S. degree from Peking University in 2001, an M.F.E. degree from the University of California, Berkeley in 2007, an M.S. degree in 2003 and a Ph.D. degree in 2012 both from Duke University. While studying at Duke, he received full tuition scholarship, research assistantship, teaching assistantship and conference travel fellowship.