Delft University of Technology
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft Institute of Applied Mathematics

CVA calculation, an extended marked branching diffusion approach

A thesis submitted to the
Delft Institute of Applied Mathematics
in partial fulfillment of the requirements

for the degree

MASTER OF SCIENCE
in
APPLIED MATHEMATICS

by

Laurens Jan Borsje

Delft, the Netherlands
June 2014

MSc THESIS APPLIED MATHEMATICS

"CVA calculation, an extended marked branching diffusion approach"

Laurens Jan Borsje

Delft University of Technology

Daily supervisor

dr. J. H. M. Anderluh

Responsible professor

Prof. dr. F. H. J. Redig

Other thesis committee members

Prof. dr. ir. C. W. Oosterlee

dr. D. Fedorets

June, 2014

Delft

Contents

1 Abstract **6**

2 Introduction **7**

3 CVA **8**
3.1 Example PDE's . . . 8
3.1.1 Type I . . . 10
3.1.2 Type II . . . 11

4 Branching Diffusion **12**
4.1 Killed Brownian Motion . . . 12
4.2 McKean . . . 13
4.3 Branching Particle Process . . . 15
4.3.1 Stochastic Rates . . . 15
4.3.2 Stochastic Default Rates . . . 16
4.3.3 Immortal Descendants . . . 16
4.4 Branch and Bound . . . 16
4.5 Marked Branch and Bound . . . 20
4.6 Convergence Considerations and Conditions . . . 23

5 Polynomial Approximation **24**
5.1 Interval . . . 24
5.2 Example . . . 24
5.2.1 Two Sided Payoffs . . . 25
5.3 Bounds . . . 26
5.3.1 Bounds on Two Sided Payoffs . . . 28

6 Interest Rates **29**

7 Analytical Results **34**
7.1 Type I . . . 34
7.2 Type II . . . 35
7.3 Exact Solutions for the Constant and Hull White Interest Rate Models . . . 35

8 Implementation **37**
8.1 Matlab . . . 37
8.2 C++ . . . 37
8.3 Functionality . . . 37

9 Results **38**
9.1 Parameters . . . 38
9.2 Hull White Rates . . . 38
9.2.1 Parameters . . . 38
9.2.2 Type I . . . 38
9.2.3 Type II . . . 43

10 Conclusions **49**
10.1 Future research . . . 49

A Personal Note **50**

B Results Continued **51**
B.1 Constant Rates . . . 51
B.1.1 Parameters . . . 51
B.1.2 Type I . . . 51
B.1.3 Type II . . . 56
B.2 CIR Rates . . . 60
B.2.1 Parameters . . . 60
B.2.2 Type I . . . 61
B.2.3 Type II . . . 66

C Animation frames **71**

D P_n Coefficients 74

1 Abstract

The marked branching diffusion algorithm as proposed by [Henry-Labordére, 2012], based on the particle diffusion introduced by [McKean, 1975], is extended upon to include stochastic interest rate models. This extended branching diffusion algorithm is used to solve pricing PDE's for equity derivatives including CVA, using the two types of default conventions as in [Brigo and Morini, 2011]. Analytical results are then used to evaluate the performance of the algorithms in the case of one sided payoffs at maturity in a constant and Hull White interest rate world. An implementation of the lower and upper bounds, as suggested by [Henry-Labordére et al., 2013], for the error introduced by the algorithms polynomial approximation is also given. The main results are the mismatches in price introduced in far out of the money derivatives having value close to zero and the extension to the stochastic interest rate framework.

2 Introduction

Since the fall of Lehman Brothers in 2008 and the subsequent crisis the financial industry has changed. Previous truths are now being questioned. Banks are no longer considered too big to fail, governments are reducing their (implicit) support, while credit worthiness and ratings have dropped across the market. A counterparty defaulting before maturity of a deal is now a real possibility and should be taken into consideration when pricing deals. This Credit Value Adjustment (CVA) will depend on the creditworthiness of the counterparty as well as the collateral agreements made.

ING, a global financial institution of Dutch origin, engages in numerous deals per day. Most of these deals are collateralized and rebalanced daily, reducing but not eliminating CVA completely. However, Over The Counter (OTC) derivatives might not have this collateralization and contain a significant exposure to a counterparty defaulting. From a risk management perspective it is vital that CVA is incorporated in these deals to compensate for this exposure and possibly use this charge to hedge the exposure in Credit Default Swap (CDS) market. The main focus of this thesis will be on this type of uncollateralized deals.

[Henry-Labordére, 2012] proposes the use of a marked branching diffusion approach to incorporate CVA when pricing derivatives. This thesis will deal with the implementation of this approach and extend upon it by finding an answer to the following main research question: Can the marked branching diffusion approach be used to price deals including CVA? This will be done using the following subquestions: Can the marked branching diffusion approach be extended to include stochastic interest rates? What are the advantages and disadvantages of using the marked branching diffusion approach?

These questions will be treated within an equity framework. Although there are not many OTC derivatives in this market having large CVA components it is the most basic setting to consider. It is possible to extend the methods and algorithms to other classes such as FX and Fixed income, however this introduces additional class specific complications not related to the branching diffusion method.

This thesis is organized as follows: Example problems and different default type agreements will be discussed in Section 3, while Labordères marked branching approach and earlier results from [McKean, 1975] will be explained in Section 4. The polynomial approximations of x^+ and bounds of these will be derived in Section 5. Stochastic interest rates will be introduced in Section 6. Analytical results used to evaluate the branch and bound results will be discussed in Section 7. The implementation in C++ and Matlab will be briefly discussed in Section 8 while the results are treated in Section 9. The findings, conclusions and further research will be discussed in Section 10

3 CVA

To calculate CVA it is important to know what actually happens when the counterparty defaults. Note that CVA is a one-sided adjustment and that the default of the party issuing the deal, Debt Value Adjustment (DVA), is not taken into consideration. It is possible to include DVA using the framework discussed in this thesis, however this will not be implemented. Suppose a simple case where a deal will be canceled when the counterparty defaults freeing both parties from any obligations to pay. An example of this is given in Section 4.1. This is however not what happens in practice since it can be assumed that both parties will try and get their moneys worth. Two types of default conventions will be used throughout the thesis, conveniently named type I and type II:

I. When the counterparty defaults a portion R of the market to market value of the deal u (including CVA) is recovered if it has a positive value. The counterparty is under some form of curator and should pay of as much of its obligations as possible. If the market to market value of the deal would be negative it is assumed that the full amount will be payed out to the defaulting counterparty. The payoff at default equals:

$$F_I = -u^- + Ru^+. \tag{1}$$

II. Type II is similar to type I however instead of the market to market value of derivative, a riskfree (no risk of defaulting counterparty) valuation v is used to value the derivative at default, leading to the following payoff at default:

$$F_{II} = -v^- + Rv^+. \tag{2}$$

[Brigo and Morini, 2011] gives an excellent description of the advantages and implications of the two types. The Partial differential Equations (PDE's) corresponding with the different default agreements will be derived in the next section.

3.1 Example PDE's

Two example problems will be given in this section corresponding with type I and type II close out conventions discussed above. These will be used and extended upon throughout the thesis. It is assumed that all partial derivatives to time t and underlying X_t of u, v and F used exist and are integrable. Consider some underlying stock X and a risk free bond B with the following dynamics under the risk free measure $\mathbb{Q}$ in a constant rate r world:

$$\begin{aligned} \mathrm{d}X_t &= rX_t\mathrm{d}t + \sigma X_t \mathrm{d}W_t, \\ \mathrm{d}B_t &= r\mathrm{d}t. \end{aligned} \tag{3}$$

Furthermore suppose that the counterparty gives out a risky bond P, paying out zero at the moment they default with the following dynamics under $\mathbb{Q}$:

$$\mathrm{d}P_t = (r+\beta)P_t\mathrm{d}t - P_t\mathrm{d}J_t. \tag{4}$$

Where J is a Poisson jump process with intensity β and jumps of size one. The dynamics of the derivative $u(t, X, J)$ are needed to derive a pricing PDE for $u(t, X, J)$ with the counterparty. Using [Shreve, 2004] Itô's lemma for two dimensional processes with jumps :

$$\begin{aligned} u(t, X_t, J_t) = u(0, X_0, J_0) &+ \int_0^t \frac{\partial u}{\partial t}(s, X_s, J_s)\mathrm{d}s + \int_0^t \frac{\partial u}{\partial X}(s, X_s, J_s)\mathrm{d}X_s + \frac{1}{2}\int_0^t \frac{\partial^2 u}{\partial X^2}(s, X_s, J_s)\mathrm{d}X_s\mathrm{d}X_s \\ &+ \sum_{0\le s\le t}(F(s, X_s) - u(s, X_{s-}, J_{s-})). \end{aligned} \tag{5}$$

Where X_{-s} equals $\lim_{t\uparrow s} X_t$ and $(F(s, X_s) - u(s, X_{s-}, J_{s-})$ is the size of the jump in u if a jump happens at time s. Suppose that $F(s, X_s)$ (the value after a jump) is continuous at $(s, X_s,)$:

$$\begin{aligned} \sum_{0\le s\le t}(F(s, X_s) - u(s, X_{s-}, J_{s-}) &= \sum_{0\le s\le t}(F(s, X_{-s}) - u(s, X_{s-}, J_{s-})\Delta J_s, \\ &= \int_0^t F(s, X_s) - u(s, X_s, J_s)\mathrm{d}J_s. \end{aligned} \tag{6}$$

Where ΔJ_s is the jump size at s. Using the above:

$$u(t,X_t,J_t) = u(0,X_0,J_0) + \int_0^t \frac{\partial u}{\partial t}(s,X_s,J_s)\mathrm{d}s + \int_0^t \frac{\partial u}{\partial X}(s,X_s,J_s)\mathrm{d}X_s + \frac{1}{2}\int_0^t \frac{\partial^2 u}{\partial X^2}(s,X_s,J_s)\mathrm{d}X_s\mathrm{d}X_s + \int_0^t F(s,X_s) - u(s,X_s,J_s)\mathrm{d}J_s.$$

This means a differential notation for $\mathrm{d}u$ is found:

$$\mathrm{d}u = \frac{\partial u}{\partial t}(t,X_t,J_t)\mathrm{d}t + \frac{\partial u}{\partial X}(t,X_t,J_t)\mathrm{d}X + \frac{\partial^2 u}{\partial X^2}(t,X_t,J_t)\mathrm{d}X_t\mathrm{d}X_t + (F(t,X_t) - u(t,X_t,J_t))\mathrm{d}J. \tag{7}$$

Now consider the following risk free portfolio Π:

$$\Pi = u + c_1 X + c_2 P. \tag{8}$$

For this portfolio to be risk free it should have the same dynamics as a risk free bond:

$$\begin{aligned} r\Pi\mathrm{d}t &= \mathrm{d}\Pi, \qquad (9)\\ &= \mathrm{d}u + c_1\mathrm{d}X + c_2\mathrm{d}P,\\ &= \frac{\partial u}{\partial t}\mathrm{d}t + \frac{\partial u}{\partial X}\mathrm{d}X + (F-u)\mathrm{d}J + \frac{1}{2}\frac{\partial^2 u}{\partial X^2}\mathrm{d}X\mathrm{d}X + c_1\mathrm{d}X + c_2\mathrm{d}P,\\ &= \frac{\partial u}{\partial t}\mathrm{d}t + \frac{\partial u}{\partial X}(rX\mathrm{d}t + \sigma X\mathrm{d}W) + (F-u)\mathrm{d}J + \frac{1}{2}\frac{\partial^2 u}{\partial X^2}\sigma^2 X^2\mathrm{d}t + c_1(rX\mathrm{d}t + \sigma X\mathrm{d}W) + c_2((r+\beta)P\mathrm{d}t - P\mathrm{d}J),\\ &= \left(\frac{\partial u}{\partial t} + rX\frac{\partial u}{\partial X} + \sigma^2 X^2\frac{1}{2}\frac{\partial^2 u}{\partial X^2} + c_1 rX + c_2(r+\beta)\right)\mathrm{d}t + \left(\sigma X\frac{\partial u}{\partial X} + c_1\sigma X\right)\mathrm{d}W + (F - u - c_2 P)\,\mathrm{d}J. \end{aligned}$$

Use the following weights in the portfolio, assuming continuous rebalancing is possible:

$$c_1 = -\frac{\partial u}{\partial X}, \tag{10}$$

$$c_2 = (F-u)\frac{1}{P}. \tag{11}$$

This corresponds to regular delta hedging as in Black Scholes to compensate movement of X due to randomness introduced by $\mathrm{d}W$ and buying and selling risky bonds from the counterparty to compensate for a jump if the counterparty defaults. Using the weights c_1 and c_2 equation (9) can be written as:

$$r(u - \frac{\partial u}{\partial X}X + (F-u))\mathrm{d}t = \left(\frac{\partial u}{\partial t} + rX\frac{\partial u}{\partial X} + \sigma^2 X^2\frac{1}{2}\frac{\partial^2 u}{\partial X^2} - \frac{\partial u}{\partial X}rX + (F-u)\frac{1}{P}(r+\beta)\right)\mathrm{d}t. \tag{12}$$

By rearranging the terms and adding a payoff at maturity T gives a pricing PDE for $u, t \leq T$ still dependent on F, the payout in case of a default of the counterparty:

$$\begin{aligned} \frac{\partial u}{\partial t} + rX\frac{\partial u}{\partial X} + \sigma^2 X^2\frac{1}{2}\frac{\partial^2 u}{\partial X^2} + \beta(F-u) - ru = 0, \qquad (13)\\ u(T,X) = g(x). \end{aligned}$$

Notice how jump process J is no longer part of the equation, its effects have been included by the term $\beta(F-u)$. Every derivative u can be hedged. Thus the market model is complete. The first and second fundamental theorem of asset pricing guarantee absence of arbitrage and uniqueness of the value of a derivative as well as the martingale property for derivatives in this setting. Constant rates allow for using the substitutions $u(t,X_t) = e^{-rT-t}u(t,X_t), g(x) = g(x)e^{-rT-t}u(t,X_t)$ and $v(t,X_t) = e^{-rT-t}v(t,X_t)$ and the fact that $F(t,u_t)$ is linear in u and v:

$$
\begin{aligned}
F_I(t, \lambda u_t) &= \lambda F_I(t, u_t), \\
F_{II}(t, \lambda v_t) &= \lambda F_{II}(t, v_t).
\end{aligned} \tag{14}
$$

Thus equation (13) can be reformulated to:

$$
\begin{aligned}
\frac{\partial u}{\partial t} + rX\frac{\partial u}{\partial X} + \sigma^2 X^2 \frac{1}{2}\frac{\partial^2 u}{\partial X^2} + \beta(F - u) = 0, \\
u(T, X) = g(x).
\end{aligned} \tag{15}
$$

3.1.1 Type I

To get the type I example PDE start from (15) and use the default convention corresponding with type I:

$$
\begin{aligned}
\frac{\partial u}{\partial t} + rX\frac{\partial u}{\partial X} + \sigma^2 X^2 \frac{1}{2}\frac{\partial^2 u}{\partial X^2} + \beta(-u^- + Ru^+ - u) = 0, \\
u(T, X) = g(x).
\end{aligned} \tag{16}
$$

The solution to (16) can also be represented stochastically using the following dynamics for u, no longer dependent on J and substituting F_I back to reduce clutter:

$$
\begin{aligned}
\mathrm{d}u(t, X_t) &= \left(\frac{\partial u}{\partial t}(t, X_t) + rX_t\frac{\partial u}{\partial x}(t, X_t) + \frac{1}{2}\sigma^2 X_t^2 \frac{\partial^2 u}{\partial x^2}(t, X_t)\right)\mathrm{d}t + \frac{\partial u}{\partial x}(t, X_t)\sigma X_t \mathrm{d}W_t, \\
&= \beta(u(t, X_t) - F_I(t, u))\mathrm{d}t + \frac{\partial u}{\partial x}(t, X_t)\sigma \mathrm{d}W_t.
\end{aligned} \tag{17}
$$

Via a Feynman Kac [Shreve, 2004] type argument by reordering terms, using an integrating factor $e^{-\beta t}$ and integrating left and right hand side:

$$
\begin{aligned}
\mathrm{d}u(t, X_t) - \beta u(t, X_t)\mathrm{d}t &= -\beta F_I(t, u_t)\mathrm{d}t + \frac{\partial u}{\partial x}(t, X_t)\sigma \mathrm{d}W_t, \\
e^{-\beta t}\mathrm{d}u(t, X_t) - e^{-\beta t}\beta u(t, X_t)\mathrm{d}t &= -e^{-\beta t}\beta F_I(t, u_t)\mathrm{d}t + e^{-\beta t}\frac{\partial u}{\partial x}(t, X_t)\sigma \mathrm{d}W_t, \\
\mathrm{d}e^{-\beta t}u(t, X_t) &= -e^{-\beta t}\beta F_I(t, u_t)\mathrm{d}t + e^{-\beta t}\frac{\partial u}{\partial x}(t, X_t)\sigma \mathrm{d}W_t, \\
e^{-\beta T}u(T, X_T) - e^{-\beta t}u(t, X_t) &= -\int_t^T e^{-\beta s}\beta F_I(s, u_s)\mathrm{d}s + \int_t^T e^{-\beta s}\frac{\partial u}{\partial x}(s, X_s)\sigma \mathrm{d}W_s, \\
u(t, X_t) &= e^{-\beta(T-t)}u(T, X_T) + \int_t^T e^{-\beta(s-t)}\beta F_I(s, u_s)\mathrm{d}s - \int_t^T e^{-\beta(s-t)}\frac{\partial u}{\partial x}(s, X_s)\sigma \mathrm{d}W_s.
\end{aligned} \tag{18}
$$

Taking expectations given filtration on time t:

$$
\begin{aligned}
\mathbb{E}_t[u(t, X_t)] &= \mathbb{E}_t\left[e^{-\beta(T-t)}u(T, X_T)\right] + \mathbb{E}_t\left[\int_t^T e^{-\beta(s-t)}\beta F_I(s, u_s)\mathrm{d}s\right] - \mathbb{E}_t\left[\int_t^T e^{-\beta(s-t)}\frac{\partial u}{\partial x}(s, X_s)\sigma \mathrm{d}W_s\right], \\
u(t, X_t) &= \mathbb{E}_t\left[e^{-\beta(T-t)}g(X_T)\right] + \mathbb{E}_t\left[\int_t^T e^{-\beta(s-t)}\beta F_I(s, u_s)\mathrm{d}s\right].
\end{aligned} \tag{19}
$$

When an independent random variable τ', exponentially distribution with parameter β is introduced, such that $\tau = t + \tau'$ the above can be rewritten to to the following:

$$
u(t, X_t) = \mathbb{E}_t\left[I_{\tau > T}g(X_T)\right] + \mathbb{E}_t\left[I_{\tau \leq T}F_I(\tau, u_\tau)\right]. \tag{20}
$$

Where I is the indicator function. Intuitively this can be seen as a an expected payoff of a derivative restricted to the outcome space where the counterparty does not default before time T plus an expected payoff restricted to the outcome space where the counterparty defaults at time $\tau \leq T$. When $u(\tau, X_\tau)$ in the second expectation of the above stochastic form is replaced an iterated expectation is found:

$$
\begin{aligned}
u(t, X_t) &= \mathbb{E}_t \left[\mathbb{I}_{\tau > T} g(X_T) \right] + \mathbb{E}_t \left[\mathbb{I}_{\tau \leq T} F_I(\tau, u_\tau) \right], \\
&= \mathbb{E}_t \left[\mathbb{I}_{\tau_1 > T} g(X_T) \right] + \mathbb{E}_t \left[\mathbb{I}_{\tau_1 \leq T} F_I(\tau_1, \mathbb{E}_t \left[\mathbb{I}_{\tau_2 > T} g(X_T) \right] + \mathbb{E}_t \left[\mathbb{I}_{\tau_2 \leq T} F_I(\tau_2, u_{\tau_2}) \right]) \right].
\end{aligned} \tag{21}
$$

Where $\tau_1 = t + \tau_1'$ and $\tau_2 = t + \tau_1' + \tau_2'$ with τ_i''s exponentially distributed with parameter β. This can be repeated indefinitely, giving rise to significant problems when a numerical integration or Monte Carlo method is used to directly evaluate the expressions. The number of simulations or gridpoints (n) needed increases by a factor n with each iterated integral/expectation, making the process impossible to implement. This could be circumvented by using an approximation of u (not requiring iterated expectations to compute) at some iteration.

3.1.2 Type II

The PDE for problems of type II is derived in a similar fashion to the type I PDE previously. Starting from (15) and substituting F_{II}:

$$
\begin{aligned}
\frac{\partial u}{\partial t} + rX\frac{\partial u}{\partial X} + \frac{1}{2}\sigma^2 X^2 \frac{\partial^2 u}{\partial X^2} + \beta(-v^- + Rv^+ - u) = 0, \\
u(T, X) = g(x).
\end{aligned} \tag{22}
$$

Where v is the solution to:

$$
\begin{aligned}
\frac{\partial v}{\partial t} + rX\frac{\partial v}{\partial X} + \frac{1}{2}\sigma^2 X^2 \frac{\partial^2 v}{\partial X^2} = 0, \\
v(T, x) = g(x).
\end{aligned} \tag{23}
$$

Using a Feynman Kac type argument the above is solved by $v(t, X_t) = \mathbb{E}_t \left[g(X_T) \right]$ and proceeding as before results in a stochastic representation:

$$
u(t, X_t) = \mathbb{E}_t \left[\mathbb{I}_{\tau > T} g(X_T) \right] + \mathbb{E}_t \left[\mathbb{I}_{\tau \leq T} F_{II}(\tau, u_\tau) \right]. \tag{24}
$$

This can again intuitively be seen as a an expected payoff of a derivative restricted to the outcome space where the counterparty does not default before time T plus an expected payoff restricted to the outcome space where the counterparty defaults at time $\tau \leq T$. However this time substituting $F_{II}(\tau, u_\tau)$ pays off and results in the following:

$$
\begin{aligned}
u(t, X_t) &= \mathbb{E}_t \left[\mathbb{I}_{\tau > T} g(X_T) \right] + \mathbb{E}_t \left[\mathbb{I}_{\tau \leq T} (-v(\tau, X_\tau)^- + Rv(\tau, X_\tau)^+) \right], \\
&= \mathbb{E}_t \left[\mathbb{I}_{\tau > T} g(X_T) \right] + \mathbb{E}_t \left[\mathbb{I}_{\tau \leq T} (-\mathbb{E}_t \left[g(X_T) \right]^- + R\mathbb{E}_t \left[g(X_T) \right]^+) \right].
\end{aligned} \tag{25}
$$

This stochastic representation can be computed numerically or using a Monte Carlo Method and corresponds with (21) where an approximation is used at τ_1.

4 Branching Diffusion

In this section the branch and bound algorithm will be explained. Starting with a simple example of killed Brownian motion and building up to marked branching diffusion. A branching diffusion process will be defined and convergence conditions will be discussed. Both type I and type II PDE's will be discussed by approximating F with a polynomial.

4.1 Killed Brownian Motion

Consider the following example from [Steele, 2001]. Assume for now that $u(t,x)$ is the unique solution of the initial value problem:

$$\begin{aligned} \frac{\partial u}{\partial t} &= \frac{1}{2}\frac{\partial^2 u}{\partial x^2}, \\ u(0,x) &= f(x). \end{aligned} \tag{26}$$

The solution can be written as a Gaussian integral, applying a change of variables $y = u - x$:

$$\begin{aligned} u(t,x) &= \frac{1}{\sqrt{2\pi t}}\int_{-\infty}^{\infty} f(u)e^{-(u-x)^2/(2t)}\mathrm{d}u, \\ &= \frac{1}{\sqrt{2\pi t}}\int_{\mathbb{R}} f(x+y)e^{-(y)^2/(2t)}\mathrm{d}y, \\ &= \mathbb{E}\left[f(x+W_t)\right]. \end{aligned} \tag{27}$$

Now consider the killed process X_t defined as a Brownian motion killed at time τ, a non-negative random variable, and extend f to

$$\begin{aligned} \tilde{f} &: (\mathbb{R}, I) \to \mathbb{R}, \\ \tilde{f}(x,i) &= \begin{cases} f(x) & \text{if } i = 0, \\ 0 & \text{if } i = 1. \end{cases} \end{aligned} \tag{28}$$

Where $I = \{0,1\}$ and $i \in I$,

$$X_t = \begin{cases} (W_t, 0) & \text{if } 0 \le t < \tau, \\ (W_\tau, 1) & \text{if } \tau \le t (\text{coffin state}). \end{cases} \tag{29}$$

The process $X_t \in (\mathbb{R}, I)$ is called exponentially killed Brownian motion with instantaneous killing rate β if τ has an exponential distribution with parameter β (independent of W_t). The question of interest becomes: Is there an initial value problem satisfied by:

$$\tilde{u}(t,x) = \mathbb{E}[\tilde{f}((x,0) + X_t)]. \tag{30}$$

Note that $\tilde{u}(0,x) = \tilde{f}(x,0) = f(x)$, giving a boundary condition. An expression for $\frac{\partial \tilde{u}}{\partial t}$ remains to be found. Splitting the expectation gives:

$$\begin{aligned} \tilde{u}(x,t) &= \mathbb{E}[\tilde{f}((x,0) + X_t)], \\ &= \mathbb{E}[f(x+W_t)\mathbb{I}_{\tau > t}] + 0, \\ &= e^{-\beta t}\mathbb{E}[f(x+W_t)]. \end{aligned} \tag{31}$$

Working out the partial derivative to t gives:

$$\begin{aligned}\frac{\partial \tilde{u}}{\partial t} &= e^{-\beta t}\frac{\partial}{\partial t}\mathbb{E}[f(x+W_t)] - \beta e^{-\beta t}\mathbb{E}[f(x+W_t)], \\ &= e^{-\beta t}\frac{\partial u}{\partial t} - \beta\tilde{u}, \\ &= e^{-\beta t}\frac{1}{2}\frac{\partial^2 u}{\partial x^2} - \beta\tilde{u}, \\ &= \frac{1}{2}\frac{\partial^2}{\partial x^2}e^{-\beta t}\mathbb{E}[f(x+W_t)] - \beta\tilde{u}, \\ &= \frac{1}{2}\frac{\partial^2 \tilde{u}}{\partial x^2} - \beta\tilde{u}.\end{aligned} \tag{32}$$

Notice how similar this PDE is to (26). Actually Feynman Kac can also be used to get the result immediately.

$$\begin{aligned}\frac{\partial u}{\partial t} &= \frac{1}{2}\frac{\partial^2 u}{\partial x^2} + q(t,x)u(t,x), \\ u(0,x) &= f(x).\end{aligned} \tag{33}$$

This can be represented stochastically as:

$$u(t,x) = \mathbb{E}\left[f(x+W_t)e^{\int_0^t q(s,x)}\mathrm{d}s\right] \tag{34}$$

4.2 McKean

The following example is from [McKean, 1975] and introduces the a branching diffusion process to solve a similar PDE.

$$\begin{aligned}\frac{\partial u}{\partial t} &= \frac{1}{2}\frac{\partial^2 u}{\partial x^2} + u^2 - u, \text{ where} \\ u(0,x) &= f(x).\end{aligned} \tag{35}$$

The above can be considered as a pricing PDE for a derivative u based on stock X moving according to a regular Brownian motion in a zero rate world, where at default (or an other exponentially distributed event with parameter 1) the value of the derivative is squared. Similar to (16) the above can be rewritten to a stochastic form with the use of a Feynman Kac type argument:

$$u(t,x) = \mathbb{E}_{0,x}\left[f(x+z_1^0(t))\right]e^{-t} + \mathbb{E}_{0,x}\left[I_{\tau\le t}u^2(t-s, x+W_\tau)\right]. \tag{36}$$

Now define a branching process z^{t_0} where a single particle starts at the origin at time t_0 and begins a standard Brownian motion. After exponentially distributed time, with parameter $\beta = 1$, the particle dies and leaves two descendants who follow the same process. Let $z_1^{t_0}(t), \cdots, z_n^{t_0}(t)$ be the location of the particles 1 to $n(z^{t_0})(t)$ generated by the original particle and its descendants at time t. (The above definition of a simple branching process suffices for this example. A complete definition and some results on branching processes can be found in section 4.3.) Also let τ be the time of death of the starting particle. The following derivation will show $\hat{u}(x,t)$ solves (35):

$$\hat{u}(t,x) = \mathbb{E}_{0,x}\left[\prod_{i=1}^{n(z^0)(t)} f(x+z_i^0(t))\right]. \tag{37}$$

For now assume $||f||_{L_\infty} \le 1$, this guarantees boundedness. Splitting the expectation gives:

$$= \mathbb{E}_{0,x}\left[I_{\tau>t}\prod_{i=1}^{n(z^0)(t)} f(x+z_i^0(t))\right] + \mathbb{E}_{0,x}\left[I_{\tau\le t}\prod_{i=1}^{n(z^0)(t)} f(x+z_i^0(t))\right]. \tag{38}$$

Consider the first part on the outcome space restricted to the set where $\{\tau > t\}, n(z^0)(t) = 1$:

$$\mathbb{E}_{0,x}\left[I_{\tau>t}\prod_{i=1}^{n(z^0)(t)} f(x+z_i^0(t))\right] = \mathbb{E}_{0,x}\left[I_{\tau>t} f(x+z_1^0(t))\right]. \tag{39}$$

By Independence,

$$\begin{aligned} &= \mathbb{E}_{0,x}\left[I_{\tau>t}\right]\mathbb{E}_{0,x}\left[f(x+z_1^0(t))\right], \\ &= \mathbb{E}_{0,x}\left[f(x+z_1^0(t))\right]\mathbb{P}[\tau > t], \\ &= \mathbb{E}_{0,x}\left[f(x+z_1^0(t))\right]e^{-t}. \end{aligned} \tag{40}$$

For the second term in (38) use iterated expectation:

$$\mathbb{E}_{0,x}\left[I_{\tau\le t}\prod_{i=1}^{n(z^0)(t)} f(x+z_i^0(t))\right] = \mathbb{E}_{0,x}\left[\mathbb{E}_{0,x}\left[I_{\tau\le t}\prod_{i=1}^{n(z^0)(t)} f(x+z_i^0(t))|\mathcal{F}_\tau\right]\right]. \tag{41}$$

Where $\mathcal{F}_\tau$ is the information known at time τ, including the death of the first particle. The indicator $I_{\tau\le t}$ is known at time τ, thus can be taken out of the inner expectation.

$$= \mathbb{E}_{0,x}\left[I_{\tau\le t}\mathbb{E}_{0,x}\left[\prod_{i=1}^{n(z^0)(t)} f(x+z_i^0(t))|\mathcal{F}_\tau\right]\right]. \tag{42}$$

The inner expectation can be rewritten, the first particle dies at time $\tau \le t$ and generates two new particles behaving according to particle branching processes $\hat{z}^\tau(t)$ and $\tilde{z}^\tau(t)$. Note that these are identical in distribution to $\hat{z}^0(s)$ and $\tilde{z}^0(s)$ if $s = t - \tau$ and $n(z^0)(t)$ equals $n(\tilde{z}^0)(s) + n(\hat{z}^0)(s)$ in distribution if $\tau \le t$:

$$= \mathbb{E}_{0,x}\left[I_{\tau\le t}\mathbb{E}_{0,x}\left[\prod_{i=1}^{n(\hat{z}^0)(s)} f(x+z_1^0(\tau)+\hat{z}_i^0(s))\prod_{i=1}^{n(\tilde{z}^0)(s)} f(x+z_1^0(\tau)+\tilde{z}_i^0(s))|\mathcal{F}_\tau\right]\right]. \tag{43}$$

By Independence of $\hat{z}^0(s)$ and $\tilde{z}^0(s)$ of $\mathcal{F}_\tau$ this equals the unconditional expectation in the point s and z^*:

$$= \mathbb{E}_{0,x}\left[I_{\tau\le t}\mathbb{E}_{0,x}\left[\prod_{i=1}^{n(\hat{z}^0)(s)} f(x+z^*+\hat{z}_i^0(s))\prod_{i=1}^{n(\tilde{z}^0)(s)} f(x+z^*+\tilde{z}_i^0(s))\right]_{z^*=z_1^0(\tau),s=t-\tau}\right]. \tag{44}$$

Note that $\mathbb{E}_{0,x}\left[\prod_{i=1}^{n(\hat{z}^s)(t)} f(x+z_1^0(s)+\hat{z}_i^s(t))|\mathcal{F}_s\right]$ and $\mathbb{E}_{0,x}\left[\prod_{i=1}^{n(\tilde{z}^s)(t)} f(x+z_1^0(s)+\tilde{z}_i^s(t))|\mathcal{F}_s\right]$ are independent and identically distributed, thus the expectation of the product equals the product of expectations.

$$= \mathbb{E}_{0,x}\left[I_{\tau\le t}\mathbb{E}_{0,x}\left[\prod_{i=1}^{n(\hat{z}^0)(s)} f(x+z^*+\hat{z}_i^0(s))\right]^2_{z^*=z_1^0(\tau),s=t-\tau}\right]. \tag{45}$$

$\hat{z}^0(s)$ equals $z^0(s)$ in distribution.

$$= \mathbb{E}_{0,x}\left[I_{\tau\le t}\mathbb{E}_{0,x}\left[\prod_{i=1}^{n(z^0)(s)} f(x+z^*+z_i^0(s))\right]^2_{z^*=z_1^0(\tau),s=t-\tau}\right]. \tag{46}$$

Using the definition of $\hat{u}$ and rewriting:

$$\begin{aligned} &= \mathbb{E}_{0,x}\left[I_{\tau\le t}\hat{u}(s, x+z^*)^2_{z^*=z_1^0(\tau), s=t-\tau}\right]. \\ &= \mathbb{E}_{0,x}\left[I_{\tau\le t}\hat{u}(t-\tau, x+z_1^0(\tau))^2\right]. \end{aligned} \tag{47}$$

Now integrate out τ

$$= \mathbb{E}_{0,x}\left[\int_0^t \hat{u}(t-s, x+z_1^0(s))^2 \mathrm{d}s\right]. \tag{48}$$

Adding the two terms together gives:

$$\hat{u}(t,x) = \mathbb{E}_{0,x}\left[f(x+z_1^0(t))\right]e^{-t} + \mathbb{E}_{0,x}\left[\int_0^t e^{-s}\hat{u}^2(t-s, x+z_1^0(s))\mathrm{d}s\right]. \tag{49}$$

This is exactly the stochastic representation of u (36), thus $\hat{u}$ is a solution to (35).

4.3 Branching Particle Process

A branching diffusion process $z^{t_0,x_0}(t)$ is a stochastic process starting with a single particle at time t_0 and location x_0 moving according to the Itô generator $\mathcal{L}$. The particle dies after exponentially distributed time τ, with parameter β.

$$P\{\tau < t\} = \int_0^t e^{-\beta(s)s}\mathrm{d}s \tag{50}$$

When the particle dies it produces $n \ge 0$ descendants at its location, where:

$$\mathbb{P}\{n=k\} = p_k, \quad 0 \le p_k \le 1 \forall k \in \mathbb{Z}^+, \quad \sum_{k=0}^{\infty} p_k = 1. \tag{51}$$

The descendant particles behave in the same way as the original particle, id est, moving according to $\mathcal{L}$, dying after random time τ and producing descendants according to (51). Note that the number of particles can increase exponentially if $\sum_{k=1}^{\infty} kp_k > 1$.

4.3.1 Stochastic Rates

In case of a (stochastic) rate process being used, the branching particle process considered, z^{t,x_t,r_t}, is slightly changed. If the Itô generator of the equation corresponds to a geometric Brownian motion with parameters μ and σ each particle z_i will move according to:

$$\mathrm{d}z_i(t) = r_i(t)z_i(t)\mathrm{d}t + \sigma z_i(t)\mathrm{d}W_i. \tag{52}$$

Where each particle will have its own stochastic rate process $r_i(t)$, starting with the value of the rate process of the parent particle at its death.

4.3.2 Stochastic Default Rates

If the default rate is not constant but a stochastic parameter, the process is extended to include this. Each particle i will decay after random time τ_i, where τ_i has the following distribution:

$$P\{\tau_i \leq t\} = \int_0^t \beta_i(s)e^{-\beta_i(s)s}\mathrm{d}s. \tag{53}$$

Where $\beta_i(t)$ will behave according to its own random process where its starting value is the value of the β of the parent process at the time of its death, similar to the stochastic rate process.

4.3.3 Immortal Descendants

The branch and bound algorithm for type II problems will often use a slightly different branching particle process y^{t_0,x_0}. This process is defined in the same way as z^{t_0,x_0}, however particles of the second generation (the particles generated by the first decay) will not decay themselves, id est they become immortal.

4.4 Branch and Bound

The branch and bound algorithm is the first attempt to approximate the solution to (16), this requires extending the technique used in Section 4.2 and using a P_∞ approximation, where the coefficients sum to one, of F_I.

$$\sum_{k=0}^{\infty} p_k u^k(t, X_t) \approx F_I(u), \tag{54}$$
$$\sum_{k=0}^{\infty} p_k = 1, \quad 0 \leq p_k \leq 1 \forall k.$$

Using this polynomial approximation the type I PDE of (15) is approximated by:

$$\frac{\partial u}{\partial t}(t, X_t) + rX_t\frac{\partial u}{\partial x} + \sigma^2 X_t^2 \frac{1}{2}\frac{\partial^2 u}{\partial x^2} + \beta\left(\sum_{k=0}^{\infty} p_k u^k(t, X_t) - u(t, X_t)\right) = 0, \tag{55}$$
$$u(T, X_T) = g(X_T),$$
$$\sum_{k=0}^{\infty} p_k = 1, \quad 0 \leq p_k \leq 1 \forall k,$$
$$\beta \geq 0.$$

Equation (55) can be rewritten to a stochastic form using a Feynman Kac type argument:

$$u(t, X_t) = \mathbb{E}_{t,x}\left[I_{\tau>T} f(X_T)\right] + \mathbb{E}_{t,x}\left[I_{\tau\leq T}\sum_{k=0}^{\infty} p_k u^k(\tau, X_\tau)\right] \tag{56}$$

To approximate the solution of a type I PDE, using F_I as payoff at default define a branching particle process z^{t,X_t} as in section 4.3 using the p_k's from (55), default rate β and the following dynamics for the particles:

$$\mathrm{d}z_i(t) = rz_i(t)\mathrm{d}t + \sigma z_i(t)\mathrm{d}W_i. \tag{57}$$

Now consider the following expectation $\hat{u}(t, x)$ similar to (37):

$$\hat{u}(t, x) = \mathbb{E}_{t,x}\left[\prod_{i=1}^{n(z^{t,x})(T)} g(z_i^{t,x}(T))\right]. \tag{58}$$

Assume $||g||_{L_\infty} \leq 1$ to guarantee boundedness and split the expectation where τ is the time of death of the original particle.

$$= \mathbb{E}_{t,x}\left[I_{\tau>T}\prod_{i=1}^{n(z^{t,x})(T)} g(z_i^{t,x}(T))\right] + \mathbb{E}_{t,x}\left[I_{\tau\leq T}\prod_{i=1}^{n(z^{t,x})(T)} g(z_i^{t,x}(T))\right]. \tag{59}$$

Using a similar argument as before the left part of (59):

$$\mathbb{E}_{t,x}\left[I_{\tau>T}\prod_{i=1}^{n(z^{t,x})(T)} g(z_i^{t,x}(T))\right] = \mathbb{E}_{t,x}\left[g(X_T)\right]e^{-\beta(T-t)}. \tag{60}$$

For the second term in (58) use iterated expectations again:

$$\mathbb{E}_{t,x}\left[I_{\tau\leq T}\prod_{i=1}^{n(z^{t,x})(T)} g(z_i^{t,x}(T))\right] = \mathbb{E}_{t,x}\left[\mathbb{E}_{t,x}\left[I_{\tau\leq T}\prod_{i=1}^{n(z^{t,x})(T)} g(z_i^{t,x}(T))|\mathcal{F}_\tau\right]\right]. \tag{61}$$

The indicator $I_{\tau\leq T}$ is $\mathcal{F}_\tau$ measurable and can be taken out of the inner expectation.

$$= \mathbb{E}_{t,x}\left[I_{\tau\leq T}\mathbb{E}_{t,x}\left[\prod_{i=1}^{n(z^{t,x})(T)} g(z_i^{t,x}(T))|\mathcal{F}_\tau\right]\right]. \tag{62}$$

The first particle dies at time $\tau \leq T$ and leaves k descendants with probability p_k.

$$= \mathbb{E}_{t,x}\left[I_{\tau\leq T}\sum_{k=0}^{\infty} p_k \mathbb{E}_{t,x}\left[\prod_{j=1}^{k}\prod_{i=1}^{n(z^{t,x,j})(T)} g(z_1^{t,x}(\tau)\cdot z_i^{\tau,1,j}(T))|\mathcal{F}_\tau\right]\right]. \tag{63}$$

Where $z^{\tau,1,j}$ is the branching particle process corresponding with the j-th descendent of the first particle starting at time τ and location 1. Note that all the k descendant particle processes are identically independently distributed (i.i.d.) with the same distribution as $z^{s,1}(T)$, where $s = \tau$, and are independent of $\mathcal{F}_\tau$. The conditional expectation above can now be replaced by an expectation in a point.

$$= \mathbb{E}_{t,x}\left[I_{\tau\leq T}\sum_{k=0}^{\infty} p_k \mathbb{E}_{t,x}\left[\prod_{j=1}^{k}\prod_{i=1}^{n(z^{s,x,j})(T)} g(z^*\cdot z_i^{s,1,j}(T))\right]_{z^*=z_1^{t,x}(\tau),s=\tau}\right]. \tag{64}$$

Al the terms in $\prod_{i=1}^{n(z^{t,x,j})(T)} g(z^*\cdot z_i^{s,1,j}(T))$ are independent thus the expectation of the product equals the product of expectations.

$$= \mathbb{E}_{t,x}\left[I_{\tau\leq T}\sum_{k=0}^{\infty} p_k \prod_{j=1}^{k}\mathbb{E}_{t,x}\left[\prod_{i=1}^{n(z^{s,x,j})(T)} g(z^*\cdot z_i^{s,1,j}(T))\right]_{z^*=z_1^{t,x}(\tau),s=\tau}\right]. \tag{65}$$

Using the fact that all $z_i^{s,1,j}(T)$ are identically distributed to $z_i^{s,1}(T)$.

$$= \mathbb{E}_{t,x}\left[I_{\tau\leq T}\sum_{k=0}^{\infty} p_k \mathbb{E}_{t,x}\left[\prod_{i=1}^{n(z^{s,x,})(T)} g(z^*\cdot z_i^{s,1,j}(T))\right]^k_{z^*=z_1^{t,x}(\tau),s=\tau}\right] \tag{66}$$

Using the definition of $\hat{u}(t,x)$ gives:

$$= \mathbb{E}_{t,x}\left[I_{\tau\le T}\sum_{k=0}^{\infty} p_k \hat{u}(z^*,s)^k_{z^*=z_1^{t,x}(\tau),s=\tau}\right], \tag{67}$$
$$= \mathbb{E}_{t,x}\left[I_{\tau\le T}\sum_{k=0}^{\infty} p_k \hat{u}(z_1^{t,x},\tau)^k\right].$$

Adding the two terms together gives:

$$\hat{u}(t,x) = \mathbb{E}_{t,x}\left[g(X_T)\right]e^{-\beta(T-t)} + \mathbb{E}_{t,x}\left[I_{\tau\le T}\sum_{k=0}^{\infty} p_k \hat{u}(z_1^{t,x},\tau)^k\right]. \tag{68}$$

This is the stochastic form of (55), thus $\hat{u}(t,x)$ is a solution. Similarly the Type II PDE (22) can be approximated by using a polynomial approximation of F_{II}:

$$\frac{\partial u}{\partial t}(t,X_t) + rX_t\frac{\partial u}{\partial x} + \sigma^2 X_t^2 \frac{1}{2}\frac{\partial^2 u}{\partial x^2} + \beta\left(\sum_{k=0}^{\infty} p_k v^k(t,X_t) - u(t,X_t)\right) = 0, \tag{69}$$
$$u(T,X_T) = g(X_T),$$
$$\sum_{k=0}^{\infty} p_k = 1, \quad 0 \le p_k \le 1 \forall k,$$
$$\beta \ge 0.$$

Where v is the solution to:

$$\frac{\partial v}{\partial t}(t,X_t) + rX_t\frac{\partial v}{\partial x} + \sigma^2 X_t^2 \frac{1}{2}\frac{\partial^2 v}{\partial x^2} = 0, \tag{70}$$
$$v(T,X_T) = g(X_T),$$
$$\sum_{k=0}^{\infty} p_k = 1, \quad 0 \le p_k \le 1 \forall k,$$
$$\beta \ge 0.$$

Equation (69) can be rewritten to a stochastic form by a Feynman Kac type argument:

$$u(t,X_t) = \mathbb{E}_{t,x}\left[I_{\tau>T} f(X_T)\right] + \mathbb{E}_{t,x}\left[I_{\tau\le T}\sum_{k=0}^{\infty} p_k v^k(\tau,X_\tau)\right]. \tag{71}$$

To approximate the solution of a type II PDE using F_{II} as payoff at default, define a branching particle process $y^{t,x}$ as in section 4.3 using the p_k's from (55), default rate β and the following dynamics for the particles:

$$\mathrm{d}y_i(t) = ry_i(t)\mathrm{d}t + \sigma y_i(t)\mathrm{d}W_i. \tag{72}$$

Note that in this case the first decay produces immortal descending particles. Now consider the following expectation $\hat{u}(t,x)$ similar to (58):

$$\hat{u}(t,x) = \mathbb{E}_{t,x}\left[\prod_{i=1}^{n(y^{t,x})(T)} g(y_i^{t,x}(T))\right]. \tag{73}$$

Assume $||g||_{L_\infty} \leq 1$ to guarantee boundedness and split the expectation where τ is the time of death of the original particle.

$$= \mathbb{E}_{t,x}\left[I_{\tau>T}\prod_{i=1}^{n(y^{t,x})(T)} g(y_i^{t,x}(T))\right] + \mathbb{E}_{t,x}\left[I_{\tau\leq T}\prod_{i=1}^{n(y^{t,x})(T)} g(y_i^{t,x}(T))\right]. \tag{74}$$

The left part is similar to the left part of (59):

$$\mathbb{E}_{t,x}\left[I_{\tau>T}\prod_{i=1}^{n(y^{t,x})(T)} g(z_i^{t,x}(T))\right] = \mathbb{E}_{t,x}\left[g(X_T)\right] e^{-\beta(T-t)}. \tag{75}$$

For the second term in (74) use iterated expectations again:

$$\mathbb{E}_{t,x}\left[I_{\tau\leq T}\prod_{i=1}^{n(y^{t,x})(T)} g(y_i^{t,x}(T))\right] = \mathbb{E}_{t,x}\left[\mathbb{E}_{t,x}\left[I_{\tau\leq T}\prod_{i=1}^{n(y^{t,x})(T)} g(y_i^{t,x}(T))|\mathcal{F}_\tau\right]\right]. \tag{76}$$

The indicator $I_{\tau\leq T}$ is $\mathcal{F}_\tau$ measurable and can be taken out of the inner expectation.

$$= \mathbb{E}_{t,x}\left[I_{\tau\leq T}\mathbb{E}_{t,x}\left[\prod_{i=1}^{n(y^{t,x})(T)} g(y_i^{t,x}(T))|\mathcal{F}_\tau\right]\right]. \tag{77}$$

The first particle dies at time $\tau \leq T$ and leaves k immortal descendants with probability p_k

$$= \mathbb{E}_{t,x}\left[I_{\tau\leq T}\sum_{k=0}^{\infty} p_k \mathbb{E}_{t,x}\left[\prod_{j=1}^{k}\prod_{i=1}^{n(y^{t,x,j})(T)} g(y_1^{t,x}(\tau)\cdot y_i^{\tau,1,j}(T))|\mathcal{F}_\tau\right]\right]. \tag{78}$$

Note that $n(y^{t,x,j})(T) = 1$ since these particles will not produce any more particles.

$$= \mathbb{E}_{t,x}\left[I_{\tau\leq T}\sum_{k=0}^{\infty} p_k \mathbb{E}_{t,x}\left[\prod_{j=1}^{k} g(y_1^{t,x}(\tau)\cdot y_1^{\tau,1,j}(T))|\mathcal{F}_\tau\right]\right]. \tag{79}$$

Each $y_1^{\tau,1,j}(T)$ is i.i.d. independent of $\mathcal{F}_\tau$. The expectation of the product can be written as the product of expectations and can be taken together since they are i.i.d.

$$= \mathbb{E}_{t,x}\left[I_{\tau\leq T}\sum_{k=0}^{\infty} p_k \mathbb{E}_{t,x}\left[g(y_1^{t,x}(\tau)\cdot y_1^{\tau,1,1}(T))|\mathcal{F}_\tau\right]^k\right]. \tag{80}$$

Note that $y_1^{t,x}(\tau)\cdot y_1^{\tau,1,1}(T)$ is just X_T.

$$= \mathbb{E}_{t,x}\left[I_{\tau\leq T}\sum_{k=0}^{\infty} p_k \mathbb{E}_{t,x}\left[g(X_T)|\mathcal{F}_\tau\right]^k\right]. \tag{81}$$

Using the definition of v and the fact that v it is a martingale gives the desired result.

$$= \mathbb{E}_{t,x}\left[I_{\tau\le T}\sum_{k=0}^{\infty}p_k\mathbb{E}_{t,x}\left[v(\tau,X_\tau|\mathcal{F}_\tau\right]^k\right], \tag{82}$$

$$= \mathbb{E}_{t,x}\left[I_{\tau\le T}\sum_{k=0}^{\infty}p_k v(\tau,X_\tau)^k\right]. \tag{83}$$

Adding the two terms together gives:

$$u(t,X_t) = \mathbb{E}_{t,x}\left[I_{\tau>T}f(X_T)\right] + \mathbb{E}_{t,x}\left[I_{\tau\le T}\sum_{k=0}^{\infty}p_k v^k(\tau,X_\tau)\right]. \tag{84}$$

This is the stochastic representation of (69), thus $\hat{u}$ solves the PDE.

4.5 Marked Branch and Bound

The marked branch and bound approach, as described by [?] puts no restrictions on the coefficients in the approximating polynomial. This allows for greater flexibility in choosing the polynomial, however a finite polynomial should be used. This is not really a limitation since it has to be implemented, so a finite order polynomial will be used anyway. Determining the polynomial to be used and possible bounds on the error introduced by this approximating polynomial are given in Section 5. Start with the following type I PDE:

$$\frac{\partial u}{\partial t}(x,t) + rX_t\frac{\partial^x}{\partial u^x}1(x,t) + \frac{1}{2}\sigma^2X_t^2\frac{\partial^2 u}{\partial x^2}(x,t) + \beta\left(\sum_{k=0}^{m}a_k u^k(x,t) - u(x,t)\right) = 0, \tag{85}$$

$$u(T,x) = f(x),$$

$$\beta \ge 0.$$

Where $\sum_{k=0}^{m} a_k u^k(x,t) \approx F_I(u)$. Now choose probabilities such that

$$p_k = \frac{|a_k|\cdot||g||_\infty^k}{\sum_{i=0}^{\infty}|a_i|\cdot||g||_\infty^i}. \tag{86}$$

The above choice of p_k's is optimal as it will minimize the variance of $\tilde{u}$ [Henry-Labordére et al., 2013]. Equation (85) can be written in a stochastic form using a Feynman Kac like argument:

$$u(t,x) = \mathbb{E}_{t,x}\left[f(X_T)\right]e^{-\beta(T-t)} + \mathbb{E}_{t,x}\left[I_{\tau<T}\sum_{k=0}^{m}a_k\hat{u}^k(\tau,X_\tau)\right]. \tag{87}$$

Define a branching particle process z^{t,X_t} as before using the p_k's from (86) and the following dynamics:

$$\mathrm{d}z_i(t) = rz_i(t)\mathrm{d}t + \sigma z_i(t)\mathrm{d}W_t^i. \tag{88}$$

Now consider the following expectation $\hat{u}(t,x)$ similar to the previous section:

$$\hat{u}(t,x) = \mathbb{E}_{t,x}\left[\prod_{i=1}^{n(z^{t,x})(T)} g(z_i^{t,x}(T))\prod_{j=1}^{m}\left(\frac{a_j}{p_j}\right)^{\#_j(z^{t,x})(T)}\right]. \tag{89}$$

Where $\#_j(z^{t,x})(T)$ equals the total number of events, leaving j new particles, of the whole branching particle process $z^{t,x}$ up to and including time T. Assume $||g||_{L_\infty} \le 1$ to guarantee boundedness and split the expectation where τ is the time of death of the original particle.

$$= \mathbb{E}_{t,x}\left[I_{\tau>T} \prod_{i=1}^{n(z^{t,x})(T)} g(z_i^{t,x}(T)) \prod_{j=1}^{m} \left(\frac{a_j}{p_j}\right)^{\#_j(z^{t,x})(T)} \right] + \mathbb{E}_{t,x}\left[I_{\tau\leq T} \prod_{i=1}^{n(z^{t,x})(T)} g(z_i^{t,x}(T)) \prod_{j=1}^{m} \left(\frac{a_j}{p_j}\right)^{\#_j(z^{t,x})(T)} \right]. \quad (90)$$

Using a similar argument as before the left part of (89):

$$\mathbb{E}_{t,x}\left[I_{\tau>T} \prod_{i=1}^{n(z^{t,x})(T)} g(z_i^{t,x}(T)) \right] = \mathbb{E}_{t,x}\left[g(X_T)\right] e^{-\beta(T-t)}. \quad (91)$$

For the second part use the tower property and take $I_{\tau\leq T}$ out of the inner expectation:

$$\mathbb{E}_{t,x}\left[I_{\tau\leq T} \prod_{i=1}^{n(z^{t,x})(T)} g(z_i^{t,x}(T)) \prod_{j=1}^{m} \left(\frac{a_j}{p_j}\right)^{\#_j(z^{t,x})(T)} \right] = \mathbb{E}_{t,x}\left[I_{\tau\leq T}\mathbb{E}_{t,x}\left[\prod_{i=1}^{n(z^{t,x})(T)} g(z_i^{t,x}(T)) \prod_{j=1}^{m} \left(\frac{a_j}{p_j}\right)^{\#_j(z^{t,x})(T)} |\mathcal{F}_\tau \right]\right]. \quad (92)$$

The first particle dies at time $\tau \leq T$ and leaves k descendants:

$$= \mathbb{E}_{t,x}\left[I_{\tau\leq T} \sum_{k=0}^{m} p_k \frac{a_k}{p_k} \mathbb{E}_{t,x}\left[\prod_{j=1}^{k} \prod_{i=1}^{n(z^{\tau,x,j})(T)} \left(g(z_1^{t,x}(\tau)\cdot z_i^{\tau,1,j}(T)) \prod_{l=1}^{m} \left(\frac{a_l}{p_l}\right)^{\#_l(z^{\tau,1,j})(T)} \right) |\mathcal{F}_\tau \right]\right]. \quad (93)$$

The braces over the term of the second product are added to indicate that the third product is part of it. Where $z^{\tau,1,j}$ is the branching particle process corresponding with the j-th descendant of the first particle, starting at time τ and location 1. Note the following regarding an event at time τ leaving k particles:

$$\#_l(z^{t,x})(T) = \sum_{j=1}^{k} \#_l(z^{\tau,1,j})(T) + I_{k=l}, \quad \forall l \in \{0,1,\ldots,k\}. \quad (94)$$

Note that all the k descendant particle processes are i.i.d. with the same distribution as $z^{s,1}(T)$, where $s=\tau$, and are independent of $\mathcal{F}_\tau$. The conditional expectation can now be replaced by an expectation in a point as before.

$$= \mathbb{E}_{t,x}\left[I_{\tau\leq T} \sum_{k=0}^{m} a_k \mathbb{E}_{t,x}\left[\prod_{j=1}^{k} \prod_{i=1}^{n(z^{s,x,j})(T)} \left(g(z^*\cdot z_i^{s,1,j}(T)) \prod_{l=1}^{m} \left(\frac{a_l}{p_l}\right)^{\#_l(z^{s,1,j})(T)} \right)\right]_{z^*=z_1^{t,x}(\tau),s=\tau}\right]. \quad (95)$$

Al the terms of the product $\prod_{j=1}^{k}\ldots$ i.i.d thus the expectation of the product equals the product of expectations.

$$= \mathbb{E}_{t,x}\left[I_{\tau\leq T} \sum_{k=0}^{m} a_k \mathbb{E}_{t,x}\left[\prod_{i=1}^{n(z^{s,x,j})(T)} \left(g(z^*\cdot z_i^{s,1,1}(T)) \prod_{l=1}^{m} \left(\frac{a_l}{p_l}\right)^{\#_l(z^{s,1,1})(T)} \right)\right]^k_{z^*=z_1^{t,x}(\tau),s=\tau}\right]. \quad (96)$$

Using the definition of $\hat{u}(t,x)$ gives:

$$\begin{aligned} &= \mathbb{E}_{t,x}\left[I_{\tau\leq T} \sum_{k=0}^{m} a_k \hat{u}(s,z^*)^k_{z^*=z_1^{t,x}(\tau),s=\tau}\right], \quad (97)\\ &= \mathbb{E}_{t,x}\left[I_{\tau\leq T} \sum_{k=0}^{m} a_k \hat{u}(\tau,z_1^{t,x}(\tau))^k\right]. \end{aligned}$$

Adding the two terms together gives:

$$u(t,x) = \mathbb{E}_{t,x}\left[g(X_T)\right]e^{-\beta(T-t)} + \mathbb{E}_{t,x}\left[I_{\tau<T}\sum_{k=0}^{m} a_k \hat{u}^k(\tau, X_\tau)\right]. \tag{98}$$

This is the stochastic form of (85), thus $\hat{u}(t,x)$ is the solution for the type I PDE. Now consider the type II PDE:

$$\frac{\partial u}{\partial t}(x,t) + rX_t\frac{\partial u}{\partial x}(x,t) + \frac{1}{2}\sigma^2 X_t^2\frac{\partial^2 u}{\partial x^2}(x,t) + \beta\left(\sum_{k=0}^{m} a_k v^k(x,t) - u(x,t)\right) = 0, \tag{99}$$
$$u(T,x) = g(x),$$
$$\beta \geq 0.$$

Where $\sum_{k=0}^{m} a_k v^k(x,t) \approx F_{II}(v)$. With v the solution of:

$$\frac{\partial v}{\partial t}(x,t) + rX_t\frac{\partial v}{\partial x}(x,t) + \frac{1}{2}\sigma^2 X_t^2\frac{\partial^2 v}{\partial x^2}(x,t) = 0, \tag{100}$$
$$v(T,x) = g(x). \tag{101}$$

Now choose the same probabilities as for the Type I PDE. Than (99) can be written in a stochastic form using a Feynman Kac like argument:

$$u(t,x) = \mathbb{E}_{t,x}\left[g(X_T)\right]e^{-\beta(T-t)} + \mathbb{E}_{t,x}\left[I_{\tau<T}\sum_{k=0}^{m} a_k v^k(\tau, X_\tau)\right]. \tag{102}$$

Define a branching particle process y^{t,X_t}, with immortal descendants, as before using the p_k's from (86) and the following dynamics:

$$\mathrm{d}y_i(t) = ry_i(t)\mathrm{d}t + \sigma y_i(t)\mathrm{d}W_t^i. \tag{103}$$

Now consider the following expectation $\hat{u}(t,x)$:

$$\hat{u}(t,x) = \mathbb{E}_{t,x}\left[\prod_{i=1}^{n(y^{t,x})(T)} g(y_i^{t,x}(T))\prod_{j=1}^{m}\left(\frac{a_j}{p_j}\right)^{\#_j(y^{t,x})(T)}\right]. \tag{104}$$

Where $\#_j(y^{t,x})(T)$ equals the total number of events, leaving j new particles, of the whole branching particle process $y^{t,x}$ up to and including time T. Assume $||g||_{L_\infty} \leq 1$ to guarantee boundedness and split the expectation where τ is the time of death of the original particle.

$$= \mathbb{E}_{t,x}\left[I_{\tau>T}\prod_{i=1}^{n(y^{t,x})(T)} g(y_i^{t,x}(T))\prod_{j=1}^{m}\left(\frac{a_j}{p_j}\right)^{\#_j(y^{t,x})(T)}\right] + \mathbb{E}_{t,x}\left[I_{\tau\leq T}\prod_{i=1}^{n(y^{t,x})(T)} g(y_i^{t,x}(T))\prod_{j=1}^{m}\left(\frac{a_j}{p_j}\right)^{\#_j(y^{t,x})(T)}\right]. \tag{105}$$

Using a similar argument as before the left part of (104):

$$\mathbb{E}_{t,x}\left[I_{\tau>T}\prod_{i=1}^{n(y^{t,x})(T)} g(y_i^{t,x}(T))\right] = \mathbb{E}_{t,x}\left[g(X_T)\right]e^{-\beta(T-t)}. \tag{106}$$

For the second part use the tower property and take $I_{\tau\leq T}$ out of the inner expectation:

$$\mathbb{E}_{t,x}\left[I_{\tau\leq T}\prod_{i=1}^{n(y^{t,x})(T)} g(y_i^{t,x}(T))\prod_{j=1}^{m}\left(\frac{a_j}{p_j}\right)^{\#_j(y^{t,x})(T)}\right]=\mathbb{E}_{t,x}\left[I_{\tau\leq T}\mathbb{E}_{t,x}\left[\prod_{i=1}^{n(y^{t,x})(T)} g(y_i^{t,x}(T))\prod_{j=1}^{m}\left(\frac{a_j}{p_j}\right)^{\#_j(y^{t,x})(T)}|\mathcal{F}_\tau\right]\right]. \tag{107}$$

The first particle dies at time $\tau\leq T$ and leaves k immortal descendants:

$$=\mathbb{E}_{t,x}\left[I_{\tau\leq T}\sum_{k=0}^{m} p_k\frac{a_k}{p_k}\mathbb{E}_{t,x}\left[\prod_{j=1}^{k} g(y_1^{t,x}(\tau)\cdot y_1^{\tau,1,j}(T))|\mathcal{F}_\tau\right]\right]. \tag{108}$$

Proceeding as before by i.i.d. property and martingale property of v.

$$=\mathbb{E}_{t,x}\left[I_{\tau\leq T}\sum_{k=0}^{m} a_k v(\tau,X_\tau)^k\right]. \tag{109}$$

Adding the two terms together gives:

$$\hat{u}(t,x)=\mathbb{E}_{t,x}\left[g(X_T)\right]e^{-\beta(T-t)}+\mathbb{E}_{t,x}\left[I_{\tau<T}\sum_{k=0}^{m} a_k v^k(\tau,X_\tau)\right]. \tag{110}$$

The above equation is identical to (102), thus $\hat{u}$ solves the type II PDE (99).

4.6 Convergence Considerations and Conditions

The condition on $g: ||g||_\infty\leq 1$ can easily be relaxed by considering $\tilde{u}\frac{u}{||g||_\infty}$ with $\tilde{u}(T,X_T)=|\frac{g(X_T)}{||g||_\infty}|\leq 1$. This implies that any bounded payoff g can be used (possible rescaled) by the branch and bound algorithms. In practice this means that bounding the payoff of a derivate by some large number N and possibly rescaling guarantees functionality of the algorithm.

5 Polynomial Approximation

The branch and bound approaches discussed in the previous section are valid approximations given that x^+ can be approximated well by a polynomial $P_n(x)$ on an interval $[-a, a]$:

$$P_n(x) = \sum_{i=0}^{n} c_i x^i. \tag{111}$$

This section will deal with finding such an approximating polynomial as well as lower and upper bounds for the error made by the approximation. Several animations are included in the digital version. Readers of a printed version are directed to appendix C for snapshots of the frames.

5.1 Interval

To find the optimal polynomial P_n the $L_2[-a, a]$ norm of the difference is minimized. Note that minimizing $L_2^2[-a, a]$ is equivalent since a norm is always greater or equal to zero and $f(x) = x^2$ is nondecreasing on $\mathbb{R}^+$. Suppose that P_n is the optimal polynomial on $[-1, 1]$, using $aP_n(\frac{x}{a})$ to approximate x^+ on $[-a, a]$:

$$\begin{aligned}
\min_{c_0 \cdots c_n} ||aP_n(\frac{x}{a}) - x^+||^2_{L_2[-a,a]} &= \min_{c_0 \cdots c_n} \int_{-a}^{a} \left(aP_n\left(\frac{x}{a}\right) - a\left(\frac{x}{a}\right)^+ \right)^2 \mathrm{d}x, \\
&= a^2 \min_{c_0 \cdots c_n} \int_{-a}^{a} \left(P_n\left(\frac{x}{a}\right) - \left(\frac{x}{a}\right)^+ \right)^2 \mathrm{d}x, \\
&= a^2 \min_{c_0 \cdots c_n} \int_{-1}^{1} \left(P_n(x) - x^+ \right)^2 \frac{1}{a} dx, \\
&= a \min_{c_0 \cdots c_n} \int_{-1}^{1} \left(P_n(x) - x^+ \right)^2 dx, \\
&= a \min_{c_0 \cdots c_n} ||P_n(x) - x^+||^2_{L_2[-1,1]}.
\end{aligned} \tag{112}$$

The above shows that it is sufficient to have the polynomial approximation P_n on $[-1, 1]$, since $aP_n(\frac{x}{a})$ can be used on $[-a, a]$.

5.2 Example

The minimization can be solved exactly by working out the integral and solving for all partial derivatives equal to zero. Consider the following worked out example for P_4:

$$\begin{aligned}
||x^+ - P_4(x)||^2_{L_2[-1,1]} &= ||x^+ - (c_0 + c_1 x + c_2 x^2 + c_3 x^3 + c_4 x^4)||_{L_2(-1,1)}, \\
&= \int_{-1}^{1} (x^+ - (c_0 + c_1 x + c_2 x^2 + c_3 x^3 + c_4 x^4))^2 \mathrm{d}x, \\
&= \int_{-1}^{0} (c_0 + c_1 x + c_2 x^2 + c_3 x^3 + e x^4)^2 \mathrm{d}x + \int_{0}^{1} (c_0 + (c_1 - 1)x + c_2 x^2 + c_3 x^3 + c_4 x^4)^2 \mathrm{d}x, \\
&= \frac{1}{3} - c_0 + 2c_0^2 - \frac{2}{3}c_1 + \frac{2}{3}c_1^2 - \frac{1}{2}c_2 + \frac{2}{5}c_2^2 - \frac{2}{5}c_3 + \frac{2}{7}c_3^2 - \frac{1}{3}c_4 + \frac{2}{9}c_4^2 + \frac{4}{3}c_0 c_2 + \frac{4}{5}c_0 c_4 + \frac{4}{5}c_1 c_3 + \frac{4}{7}c_2 c_4.
\end{aligned} \tag{113}$$

Taking partial derivatives equal to zero gives a solvable system of five linear equations and five unknowns :

$$\begin{aligned}
\frac{\partial}{\partial c_0} &= -1 + 4c_0 + \frac{4}{3}c_2 + \frac{4}{5}c_4 = 0 && \Rightarrow c_0 = \frac{15}{256}, \\
\frac{\partial}{\partial c_1} &= -\frac{2}{3} + \frac{4}{3}c_1 + \frac{4}{5}c_3 = 0 && \Rightarrow c_1 = \frac{1}{2}, \\
\frac{\partial}{\partial c_2} &= -\frac{1}{2} + \frac{4}{5}c_2 + \frac{4}{3}c_0 + \frac{4}{7}c_4 = 0 && \Rightarrow c_2 = \frac{105}{128}, \\
\frac{\partial}{\partial c_3} &= -\frac{2}{5} + \frac{4}{7}c_3 + \frac{4}{5}c_1 = 0 && \Rightarrow c_3 = 0, \\
\frac{\partial}{\partial c_4} &= -\frac{1}{3} + \frac{4}{9}c_4 + \frac{4}{5}c_0 + \frac{4}{7}c_2 = 0 && \Rightarrow c_4 = -\frac{105}{256}.
\end{aligned} \tag{114}$$

Generalizing this for any n, up to $n = 25$, gives the polynomial approximation given in Figure 1. Note that higher order approximation can easily be constructed, however problems occur due to machine precision, especially when the approximation is scaled to a larger sized interval.

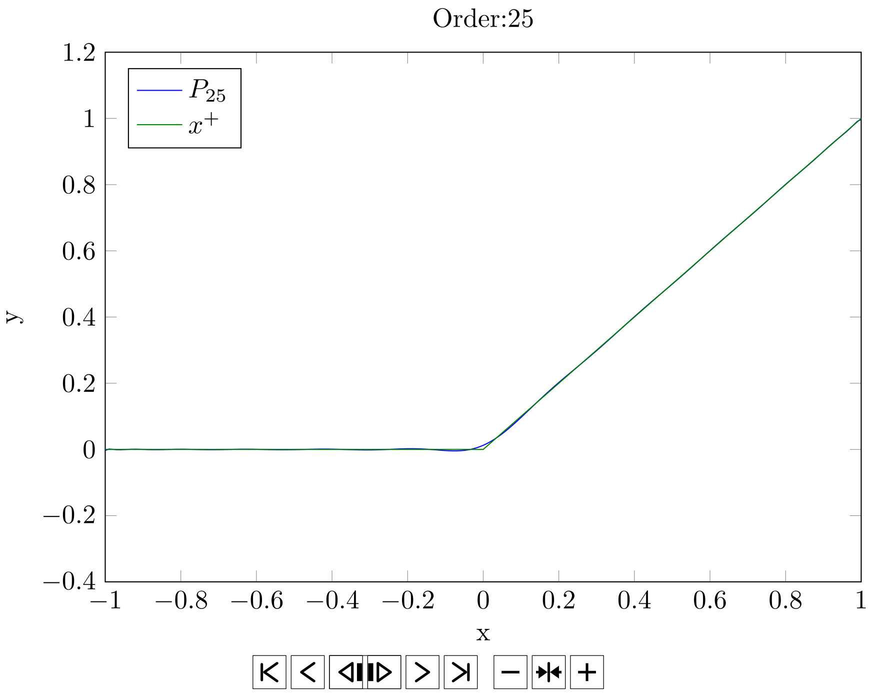


Figure 1: $P_n(x)$ approximation of x^+ on $[-1, 1]$

The polynomial approximation performs well in general, however it struggles around $x = 0$ because of the discontinuity in the derivative of x^+ at zero. Note that most functions can be approximated this way. [Weierstrass, 1885] gives convergence in L_∞ norm on [-1,1] for all continuous functions, this implies L_2 norm convergence on a [-1,1].

5.2.1 Two Sided Payoffs

If the underlying derivative has a two sided payoff x^+ is no longer sufficient and the following function will need to be approximated:

$$f(x) = Rx^+ - x^-. \tag{115}$$

If the counterparty defaults we are expected to pay the full value of the derivative if it has a negative value from our perspective, while we expect to receive a portion of the money owned to us if the counterparty defaults. Note that $x^- = (-x)^+$ thus if P_n is the polynomial used to approximate x^+ then

$$\tilde{f}_n = RP_n(x) - P_n(-x). \tag{116}$$

Note that $\tilde{f}_n$ is optimal if P_n is optimal:

$$\begin{aligned}
||\tilde{f}_n - Rx^+ - x^-||^2_{L_2[-1,1]} &= \int_{-1}^{0} \left(RP_n(x) - P_n(-x) - x\right)^2 \mathrm{d}x + \int_0^1 \left(RP_n(x) - P_n(-x) - Rx\right)^2 \mathrm{d}x, \qquad (117)\\
&= \int_{-1}^{0} R^2P_n^2(x) + P_n^2(-x) + x^2 - 2RP_n(x)P_n(-x) - 2xRP_n(x) + 2xP_n(-x)\mathrm{d}x\\
&+ \int_0^1 R^2P_n^2(x) + P_n^2(-x) + R^2x^2 - 2xR^2P_n(x)P_n(-x) - 2xRP_n(x) + 2xRP_n(-x)\mathrm{d}x,\\
&= R^2\int_{-1}^{0} P_n^2(x)\mathrm{d}x + \int_{-1}^{0} P_n^2(-x) + x^2 + 2xP_n(-x)\mathrm{d}x\\
&+ R^2\int_0^1 P_n^2(x) + x^2 - 2xP_n(x)\mathrm{d}x + \int_0^1 P_n^2(-x)\mathrm{d}x,\\
&= (R^2+1)\left(\int_{-1}^{0} P_n^2(x)\mathrm{d}x + \int_0^{-1} P_n^2(x) + x^2 - 2xP_n(x)\mathrm{d}x\right),\\
&= (R^2+1)\int_{-1}^{1}\left(P_n(x) - x^+\right)^2 dx,\\
&= (R^2+1)||P_n(x) - x^+||^2_{L_2[-1,1]}.
\end{aligned}$$

An animation of the approximations and errors made, using $R = 0.4$, is given below:

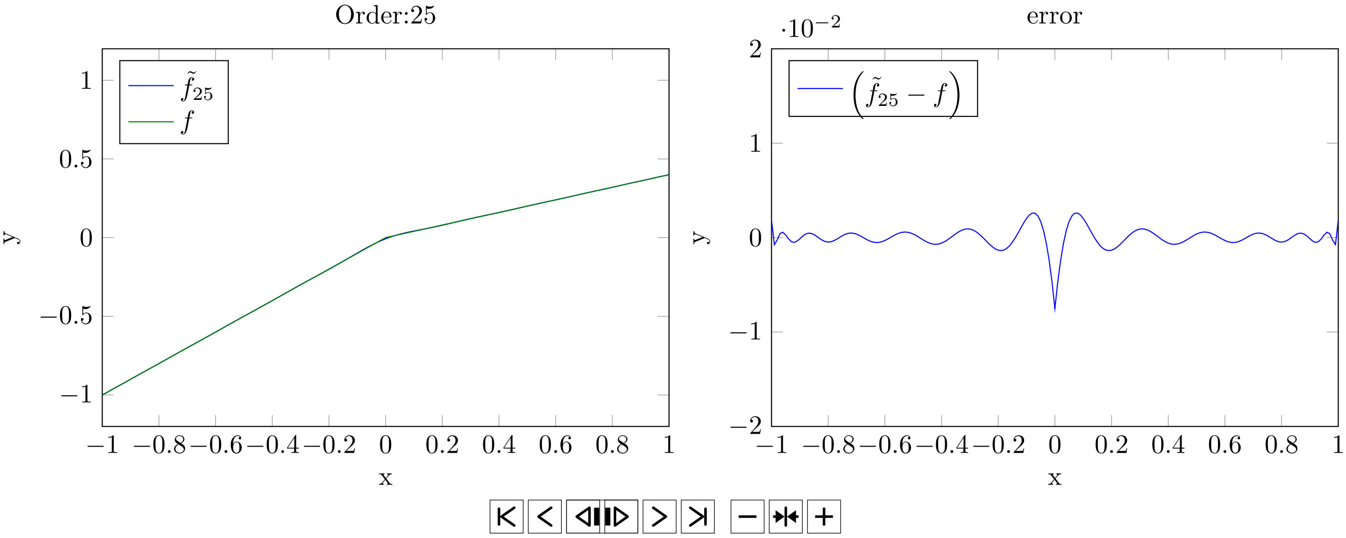


Figure 2: $\tilde{f}_n$ approximation of $Rx^+ - x^-$ on $[-1,1]$

5.3 Bounds

To get an estimate of the error made using the polynomial approximation in the previous section, u^+ or any other function can also be approximated from below and above by polynomials $\underline{P}_n$ and $\overline{P}_n$, by minimizing the distance as before and adding constraints.

$$\min ||x^+ - \overline{P}_n)||^2_{L_2[-1,1]} \quad \text{subject to} \int_{-1}^{1} I_{\overline{P}_n < x^+}\mathrm{d}x = 0, \qquad (118)$$

$$\min ||x^+ - \underline{P}_n)||^2_{L_2[-1,1]} \quad \text{subject to} \int_{-1}^{1} I_{\underline{P}_n > x^+}\mathrm{d}x = 0. \qquad (119)$$

[Weierstrass, 1885] guarantees convergence again. If $f(x) = Rx^+ - x^-$ can be approximated to within ϵ in L_∞ norm on [-1,1] by f_n given n sufficiently large. Than $f_n - \epsilon$ is a lower bound with error max 2ϵ in L_∞ norm on [-1,1] for any ϵ. Approximations from above can be made similarly. An analytic solution of the minimization is no longer available, however if the problem is discretized it can be rewritten in a more manageable form. Instead of working with $L_2[-1,1]$, l_2 used on the discretized, $x_0, x_1, \cdots, x_m$ interval [-1,1]. For the approximation from below this results in:

$$\min \sum_{i=0}^{m}(x_i^+ - \underline{P}_n(x_i))^2, \quad \text{subject to } \underline{P}_n(x_i) \leq x_i^+, \forall i \in \{0,1,\cdots,m\}, \tag{120}$$

$$\min \sum_{i=0}^{m}(x_i^+ - (\underline{c}_0 + \underline{c}_1 x_i + \underline{c}_2 x_i^2 + \cdots \underline{c}_n x_i^n))^2, \quad \text{subject to } \underline{c}_0 + \underline{c}_1 x_i + \underline{c}_2 x_i^2 + \cdots \underline{c}_n x_i^n \leq x_i^+, \forall i \in \{0,1,\cdots,m\}.$$

Matrix notation:

$$\min \sum_{i=0}^{m} \begin{bmatrix} \underline{c}_0 & \underline{c}_1 & \underline{c}_2 & \cdots & \underline{c}_n \end{bmatrix} \begin{bmatrix} 1 & x_i & x_i^2 & \cdots & x_i^n \\ x_i & x_i^2 & x_i^3 & \cdots & x_i^{n+1} \\ x_i^2 & x_i^3 & x_i^4 & \cdots & x_i^{n+2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_i^n & x_i^{n+1} & x_i^{n+2} & \cdots & x_i^{2n} \end{bmatrix} \begin{bmatrix} \underline{c}_0 \\ \underline{c}_1 \\ \underline{c}_2 \\ \vdots \\ \underline{c}_n \end{bmatrix} + 2 \begin{bmatrix} -x_i^+ \\ -(x_i^+)^2 \\ -(x_i^+)^3 \\ \vdots \\ -(x_i^+)^n \end{bmatrix}^T \begin{bmatrix} \underline{c}_0 \\ \underline{c}_1 \\ \underline{c}_2 \\ \vdots \\ \underline{c}_n \end{bmatrix} + (x_i^+)^2, \tag{121}$$

$$\text{subject to } \begin{bmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^n \\ 1 & x_2 & x_2^2 & \cdots & x_2^n \\ 1 & x_3 & x_3^2 & \cdots & x_3^n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_m & x_m^2 & \cdots & x_m^n \end{bmatrix} \begin{bmatrix} \underline{c}_0 \\ \underline{c}_1 \\ \underline{c}_2 \\ \vdots \\ \underline{c}_n \end{bmatrix} \leq \begin{bmatrix} x_0^+ \\ x_1^+ \\ x_2^+ \\ \vdots \\ x_m^+ \end{bmatrix}.$$

The last term to be minimized is independent of $\underline{\mathbf{c}}$ so can be ignored. Dividing the objective function by 2 and summing the matrices results in the following form for $\underline{P}_n$ and $\overline{P}_n$:

$$\min \frac{1}{2}\underline{\mathbf{c}}^T \mathbf{H}\underline{\mathbf{c}} + \mathbf{v}^T\underline{\mathbf{c}}, \quad \text{subject to } \mathbf{A}\underline{\mathbf{c}} \leq \mathbf{b}, \tag{122}$$

$$\min \frac{1}{2}\overline{\mathbf{c}}^T \mathbf{H}\overline{\mathbf{c}} + \mathbf{v}^T\overline{\mathbf{c}}, \quad \text{subject to } -\mathbf{A}\overline{\mathbf{c}} \leq -\mathbf{b}.$$

Where:

$$\mathbf{H}_{j,k} = \mathbf{H}_{k,j} = \sum_{i=0}^{m} x_i^{j+k-2}, \tag{123}$$

$$\mathbf{A}_{j,k} = x_j^{k-1}.$$

The form in (122), with $\mathbf{H}$ symmetric, is well known and can be efficiently approximated by *quadprog* in Matlab for example, see [Mehrotra, 1992]. The resulting approximations are given below:

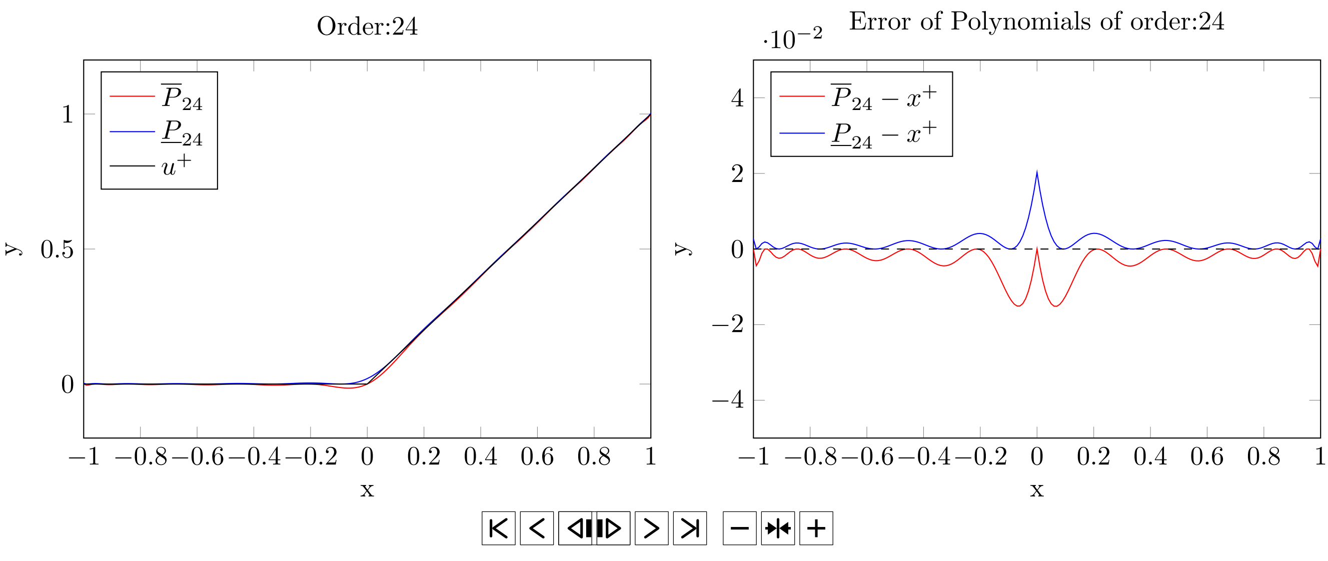


Figure 3: $\underline{P}_n$ and $\overline{P}_n$ approximation of x^+ on $[-1, 1]$

Figure 3 shows the errors made by the approximating polynomials P_n, $\underline{P}_n$ and $\overline{P}_n$. The smoothness of these polynomials guarantees that the constraints are not violated to0 much in between discretization points.

5.3.1 Bounds on Two Sided Payoffs

If $\underline{P}_n(x)$ is an approximation from below of x^+, then $-\underline{P}_n(-x)$ is an approximation from above of x^-, therefore $\tilde{f}_n$ can be bounded as follows:

$$\begin{aligned} \underline{f}_n(x) &= R\underline{P}_n(x) - \overline{P}_n(-x), \\ \overline{f}_n(x) &= R\overline{P}_n(x) - \underline{P}_n(-x). \end{aligned} \tag{124}$$

A bound on the error made using the polynomial approximation $\tilde{f}_n$ can be obtained by implementing a branch and bound algorithm for both of the above bounds and examining the difference between the two.

6 Interest Rates

In this section the marked branch and bound algorithm will be extended to include a stochastic rate process. A product of payoff functions taking discounting into account will be used and shown to solve type I and type II PDE's including stochastic interest rates.

Consider the following type I PDE, similar to (13):

$$\frac{\partial u}{\partial t}(t,X_t) + r(t)X_t\frac{\partial u}{\partial X}(t,X_t) + \sigma^2 X_t^2 \frac{1}{2}\frac{\partial^2 u}{\partial X^2}(t,X_t) + \beta(F_I(u(t,X_t)) - u(t,X_t)) - r(t)u(t,X_t) = 0, \quad u(T,X_T) = g(x). \tag{125}$$

Now approximate F_I by a polynomial as before:

$$\frac{\partial u}{\partial t}(t,X_t) + r(t)X_t\frac{\partial u}{\partial X}(t,X_t) + \sigma^2 X_t^2 \frac{1}{2}\frac{\partial^2 u}{\partial X^2}(t,X_t) + \beta\left(\sum_{k=1}^{m} a_k u^k(t,X_t) - u(t,X_t)\right) - r(t)u(t,X_t) = 0, \quad u(T,X_T) = g(x). \tag{126}$$

The solution to (126) can be represented stochastically by using a Feynman Kac type derivation.

$$u(t,X_t) = \mathbb{E}_{t,x}\left[g(X_T)e^{-\int_t^T r(s)\mathrm{d}s}\right]e^{-\beta(T-t)} + \mathbb{E}_{t,x}\left[I_{\tau<T}e^{-\int_t^\tau r(s)\mathrm{d}s}\sum_{k=0}^{m} a_k u^k(\tau,X_\tau)\right]. \tag{127}$$

The above can intuitively be interpreted as the value of the derivative u equals the expected discounted payoff times the probability that the counterparty does not default before maturity T plus a discounted payoff at default times the probability of the counterparty defaulting at that time. The marked branch and bound algorithm is extended to use the following expected product of discounted payoffs:

$$\hat{u}(t,X_t) = \mathbb{E}_{t,x}\left[\prod_{i=1}^{n(z^{t,x_t,r_t})(T)} g(z_i^{t,x_t,r_t}(T))\prod_{j=1}^{m}\left(\frac{a_j}{p_j}\right)^{\#_j(z^{t,x_t,r_t})(T)} e^{-\int_t^T \sum_{h=1}^{n(z^{t,x_t,r_t})(s)} r_h(z^{t,x_t,r_t})(s)\mathrm{d}s}\right]. \tag{128}$$

Where $r_h(z^{t,x_t,r_t})(s)$ represents the rate process corresponding with the h-th particle of the particle process z^{t,x_t,r_t} at time s. Using p_k's as defined in Section 4.5, a branching particle process z^{t,x_t,r_t} with the following dynamics for z_i:

$$\mathrm{d}z_i(t) = r_i(t)z_i(t)\mathrm{d}t + \sigma z_i(t)\mathrm{d}W_t^i, \tag{129}$$

and one of the following dynamics for the corresponding r_i as defined in [Brigo and Mercurio, 2006] and [Hull, 2012].

Constant	$\mathrm{d}r(t) = 0,$	$r(t_0) = r(t) = r_0 > 0,$	(130)
Hull White	$\mathrm{d}r(t) = \kappa(\theta - r(t))\mathrm{d}t + \eta\mathrm{d}W_r^{\mathbb{Q}},$	$r(t_0) = r_0 > 0,$	(131)
CIR	$\mathrm{d}r(t) = \kappa(\theta - r(t))\mathrm{d}t + \eta\sqrt{r(t)}\mathrm{d}W_r^{\mathbb{Q}},$	$r(t_0) = r_0 > 0.$	(132)

For the purpose of this thesis the stochastic rate processes above suffice. The aim is to show that the marked branch and bound algorithm can be extended to include a stochastic rate process and subsequent discounting. Note that the Hull White and CIR process can be extended to fit a whole market curve by introducing a time dependency in the long term mean, $\theta(t)$. Proceeding as in Section 4.5, assume $||g||_{L_\infty} \leq 1$, the expectation in (128) is split.

$$\hat{u}(t,X_t) = \mathbb{E}_{t,x}\left[I_{\tau>T}\prod_{i=1}^{n(z^{t,x_t,r_t})(T)} g(z_i^{t,x}(T))\prod_{j=1}^{m}\left(\frac{a_j}{p_j}\right)^{\#_j(z^{t,x})(T)} e^{-\int_t^T\sum_{h=1}^{n(z^{t,x_t,r_t})(s)} r_h(z^{t,x_t,r_t})(s)\mathrm{d}s}\right], \tag{133}$$
$$+\mathbb{E}_{t,x}\left[I_{\tau\le T}\prod_{i=1}^{n(z^{t,x_t,r_t})(T)} g(z_i^{t,x}(T))\prod_{j=1}^{m}\left(\frac{a_j}{p_j}\right)^{\#_j(z^{t,x})(T)} e^{-\int_t^T\sum_{h=1}^{n(z^{t,x_t,r_t})(s)} r_h(z^{t,x_t,r_t})(s)\mathrm{d}s}\right].$$

Using a similar argument as before the left part of (128) can be simplified considerably since it deals with only a single particle:

$$\mathbb{E}_{t,x}\left[I_{\tau>T}\prod_{i=1}^{n(z^{t,x_t,r_t})(T)} g(z_i^{t,x_t,r_t}(T))\prod_{j=1}^{m}\left(\frac{a_j}{p_j}\right)^{\#_j(z^{t,x_t,r_t})(T)} e^{-\int_t^T\sum_{h=1}^{n(z^{t,x_t,r_t})(s)} r_h(z^{t,x_t,r_t})(s)\mathrm{d}s}\right], \tag{134}$$
$$= \mathbb{E}_{t,x}\left[g(X_T)e^{-\int_t^T r(s)\mathrm{d}s}\right]e^{-\beta(T-t)}.$$

For the second part use the tower property and take $I_{\tau\le T}$ out of the inner expectation:

$$\mathbb{E}_{t,x}\left[I_{\tau\le T}\mathbb{E}_{t,x}\left[\prod_{i=1}^{n(z^{t,x_t,r_t})(T)} g(z_i^{t,x_t,r_t}(T))\prod_{j=1}^{m}\left(\frac{a_j}{p_j}\right)^{\#_j(z^{t,x_t,r_t})(T)} e^{-\int_t^T\sum_{h=1}^{n(z^{t,x_t,r_t})(s)} r_h(z^{t,x_t,r_t})(s)\mathrm{d}s}|\mathcal{F}_\tau\right]\right]. \tag{135}$$

The first particle dies at time $\tau \le T$ and leaves k descendants:

$$\mathbb{E}_{t,x}\left[I_{\tau\le T}\sum_{k=0}^{m} p_k\frac{a_k}{p_k}e^{-\int_t^\tau r(z^{t,x_t,r_t})(s)\mathrm{d}s}\,.\right.$$
$$\left.\mathbb{E}_{t,x}\left[\prod_{j=1}^{k}\prod_{i=1}^{n(z^{\tau,x,j})(T)}\left(g(z_1^{t,x_t,r_t}(\tau)\cdot z_i^{\tau,1,j}(T))\prod_{l=1}^{m}\left(\frac{a_l}{p_l}\right)^{\#_l(z^{\tau,1,j})(T)} e^{-\int_\tau^T\sum_{h=1}^{n(z^{\tau,1,j})(s)} r_h(z^{\tau,1,j})(s)\mathrm{d}s}\right)|\mathcal{F}_\tau\right]\right]. \tag{136}$$

Where $z^{\tau,1,j}$ represents the j-th descendant of z_1^{t,x_t,r_t} defaulting at time τ, with a corresponding rate process $r_j(z^{t,1,j})$ starting with the value of the parent particle at default $r_j(z^{t,1,j})(\tau) = r_1(z^{t,x_t,r_t})(\tau)$. Note the following regarding the discounting term, given that the first particle dies at time τ and leaves k descendants:

$$e^{-\int_t^T\sum_{h=1}^{n(z^{t,x_t,r_t})(s)} r_h(z^{t,x_t,r_t})(s)\mathrm{d}s} = e^{-\int_t^\tau r_1(z^{t,x_t,r_t})(s)\mathrm{d}s}\prod_{j=1}^{k}e^{-\int_\tau^T\sum_{h=1}^{n(z^{\tau,1,j})(s)} r_h(z^{\tau,1,j})(s)\mathrm{d}s}. \tag{137}$$

Up to time τ there is only a single particle, while after time τ multiple particle processes exist, possibly descending new particles. Now note that all the k descendant particle processes are i.i.d. with the same distribution as $z^{s,1,r_s}(T)$, where $s=\tau$ and are independent of $\mathcal{F}_\tau$. The conditional expectation can now be replaced by an expectation in a point as before.

$$\mathbb{E}_{t,x}\left[I_{\tau\le T}\sum_{k=0}^{m} a_k e^{-\int_t^\tau r(z^{t,x_t,r_t})(s)\mathrm{d}s}\,.\right.$$
$$\left.\mathbb{E}_{t,x}\left[\prod_{j=1}^{k}\prod_{i=1}^{n(z^{s,x,j})(T)}\left(g(z^*\cdot z_i^{s,1,j}(T))\prod_{l=1}^{m}\left(\frac{a_l}{p_l}\right)^{\#_l(z^{s,1,j})(T)} e^{-\int_\tau^T\sum_{h=1}^{n(z^{s,1,j})(s)} r_h(z^{s,1,j})(s)\mathrm{d}s}\right)\right]_{z^*=z_1^{t,x_t,r_t}(\tau),s=\tau}\right]. \tag{138}$$

Al the terms of the product $\prod_{j=1}^{k}\dots$ above are i.i.d thus the expectation of the product equals the product of expectations.

$$\mathbb{E}_{t,x}\left[I_{\tau\leq T}\sum_{k=0}^{m}a_k e^{-\int_t^\tau r(z^{t,x_t,r_t})(s)\mathrm{d}s}\cdot\right.$$

$$\left.\mathbb{E}_{t,x}\left[\prod_{i=1}^{n(z^{s,x,1})(T)}\left(g(z^*\cdot z_i^{s,1,1}(T))\prod_{l=1}^{m}\left(\frac{a_l}{p_l}\right)^{\#_l(z^{s,1,1})(T)}e^{-\int_\tau^T\sum_{h=1}^{n(z^{s,1,1})(s)}r_h(z^{s,1,1})(s)\mathrm{d}s}\right)\right]^k_{z^*=z_1^{t,x_t,r_t}(\tau),s=\tau}\right]. \tag{139}$$

Using the definition of $\hat{u}(t,x)$ gives:

$$\begin{aligned}&=\mathbb{E}_{t,x}\left[I_{\tau\leq T}\sum_{k=0}^{m}a_k e^{-\int_t^\tau r(z^{t,x_t,r_t})(s)\mathrm{d}s}\hat{u}(s,z^*)^k_{z^*=z_1^{t,x_t,r_t}(\tau),s=\tau}\right], &(140)\\&=\mathbb{E}_{t,x}\left[I_{\tau\leq T}\sum_{k=0}^{m}a_k e^{-\int_t^\tau r(z^{t,x_t,r_t})(s)\mathrm{d}s}\hat{u}(\tau,z_1^{t,x}(\tau))^k\right].\end{aligned}$$

Adding the two terms together gives:

$$u(t,X_t)=\mathbb{E}_{t,x}\left[g(X_T)e^{-\int_t^T r(s)\mathrm{d}s}\right]e^{-\beta(T-t)}+\mathbb{E}_{t,x}\left[I_{\tau<T}e^{-\int_t^\tau r(s)\mathrm{d}s}\sum_{k=0}^{m}a_k u^k(\tau,X_\tau)\right]. \tag{141}$$

This is the stochastic form of (146), thus $\hat{u}(t,x)$ is the solution for the type I PDE (126). Now consider the type II PDE extended in a similar way to (125):

$$\begin{aligned}\frac{\partial u}{\partial t}(t,X_t)+r(t)X_t\frac{\partial u}{\partial X}(t,X_t)+\sigma^2X_t^2\frac{1}{2}\frac{\partial^2 u}{\partial X^2}(t,X_t)+\beta(F_{II}(v(t,X_t))-u(t,X_t))-r(t)u(t,X_t)=0,&\quad(142)\\u(T,X_T)=g(X).&\end{aligned}$$

Where v is the solution to the Black and Scholes equation:

$$\begin{aligned}\frac{\partial v}{\partial t}(t,X_t)+r(t)X_t\frac{\partial v}{\partial X}(t,X_t)+\sigma^2X_t^2\frac{1}{2}\frac{\partial^2 v}{\partial X^2}(t,X_t)-r(t)v(t,X_t)=0,&\quad(143)\\v(T,X_T)=g(X).&\end{aligned}$$

v can be represented stochastically, by Feynman Kac as :

$$v(t,x)=\mathbb{E}_{t,x}\left[g(X_T)e^{-\int_t^T r(s)\mathrm{d}s}\right]. \tag{144}$$

Approximate F_{II} by a polynomial as before:

$$\begin{aligned}\frac{\partial u}{\partial t}(t,X_t)+r(t)X_t\frac{\partial u}{\partial X}(t,X_t)+\sigma^2X_t^2\frac{1}{2}\frac{\partial^2 u}{\partial X^2}(t,X_t)+\beta\left(\sum_{k=1}^{m}a_k v^k(t,X_t)-u(t,X_t)\right)-r(t)u(t,X_t)=0,&\quad(145)\\u(T,X_T)=g(x).&\end{aligned}$$

The solution to (145) can be represented stochastically by using a Feynman Kac type derivation.

$$u(t,X_t)=\mathbb{E}_{t,x}\left[g(X_T)e^{-\int_t^T r(s)\mathrm{d}s}\right]e^{-\beta(T-t)}+\mathbb{E}_{t,x}\left[I_{\tau<T}e^{-\int_t^\tau r(s)\mathrm{d}s}\sum_{k=0}^{m}a_k v^k(\tau,X_\tau)\right]. \tag{146}$$

Now choose probabilities as for the Type I PDE and define a branching particle process y^{t,x_t,r_t}, with immortal descendants, as before using the p_k's from (86) and the following dynamics for the particles:

$$\mathrm{d}y_i(t) = r(t)y_i(t)\mathrm{d}t + \sigma y_i(t)\mathrm{d}W_t^i. \tag{147}$$

Where $r_i(t)$ behaves according to (130), (131) or (132). Consider the following expectation $\hat{u}(t,x)$:

$$\hat{u}(t,X_t) = \mathbb{E}_{t,x}\left[\prod_{i=1}^{n(y^{t,x_t,r_t})(T)} g(y_i^{t,x_t,r_t}(T)) \prod_{j=1}^{m}\left(\frac{a_j}{p_j}\right)^{\#_j(y^{t,x_t,r_t})(T)} e^{-\int_t^T \sum_{h=1}^{n(y^{t,x_t,r_t})(s)} r_h(y^{t,x_t,r_t})(s)\mathrm{d}s}\right]. \tag{148}$$

Assume $||g||_{L_\infty} \leq 1$ to guarantee boundedness and split the expectation where τ is the time of death of the original particle.

$$\begin{aligned}\hat{u}(t,X_t) &= \mathbb{E}_{t,x}\left[I_{\tau>T}\prod_{i=1}^{n(y^{t,x_t,r_t})(T)} g(y_i^{t,x_t,r_t}(T)) \prod_{j=1}^{m}\left(\frac{a_j}{p_j}\right)^{\#_j(y^{t,x_t,r_t})(T)} e^{-\int_t^T \sum_{h=1}^{n(y^{t,x_t,r_t})(s)} r_h(y^{t,x_t,r_t})(s)\mathrm{d}s}\right]\\ &+ \mathbb{E}_{t,x}\left[I_{\tau\leq T}\prod_{i=1}^{n(y^{t,x_t,r_t})(T)} g(y_i^{t,x_t,r_t}(T)) \prod_{j=1}^{m}\left(\frac{a_j}{p_j}\right)^{\#_j(y^{t,x_t,r_t})(T)} e^{-\int_t^T \sum_{h=1}^{n(y^{t,x_t,r_t})(s)} r_h(y^{t,x_t,r_t})(s)\mathrm{d}s}\right].\end{aligned} \tag{149}$$

Using a similar argument as before the left part equals:

$$\mathbb{E}_{t,x}\left[g(X_T)e^{-\int_t^T r(s)\mathrm{d}s}\right]e^{-\beta(T-t)}.$$

For the second part use the tower property and take $I_{\tau\leq T}$ out of the inner expectation:

$$\mathbb{E}_{t,x}\left[I_{\tau\leq T}\mathbb{E}_{t,x}\left[\prod_{i=1}^{n(y^{t,x_t,r_t})(T)} g(y_i^{t,x_t,r_t}(T)) \prod_{j=1}^{m}\left(\frac{a_j}{p_j}\right)^{\#_j(y^{t,x_t,r_t})(T)} e^{-\int_t^T \sum_{h=1}^{n(y^{t,x_t,r_t})(s)} r_h(z^{t,x_t,r_t})(s)\mathrm{d}s}|\mathcal{F}_\tau\right]\right]. \tag{150}$$

The first particle dies at time $\tau \leq T$ and leaves k immortal descendants:

$$\begin{aligned}&\mathbb{E}_{t,x}\left[I_{\tau\leq T}\sum_{k=0}^{m} p_k\frac{a_k}{p_k}e^{-\int_t^\tau r(y^{t,x_t,r_t})(s)\mathrm{d}s}\,.\right.\\ &\left.\mathbb{E}_{t,x}\left[\prod_{j=1}^{k}\prod_{i=1}^{n(y^{\tau,x,j})(T)}\left(g(y_1^{t,x_t,r_t}(\tau)\cdot y_i^{\tau,1,j}(T))\prod_{l=1}^{m}\left(\frac{a_l}{p_l}\right)^{\#_l(y^{\tau,1,j})(T)} e^{-\int_\tau^T \sum_{h=1}^{n(y^{\tau,1,j})(s)} r_h(y^{\tau,1,j})(s)\mathrm{d}s}\right)|\mathcal{F}_\tau\right]\right].\end{aligned} \tag{151}$$

These descendants produce no more particles themselves, $n(y^{\tau,x,j})(T) = 1$.

$$\mathbb{E}_{t,x}\left[I_{\tau\leq T}\sum_{k=0}^{m} p_k\frac{a_k}{p_k}e^{-\int_t^\tau r(y^{t,x_t,r_t})(s)\mathrm{d}s}\mathbb{E}_{t,x}\left[\prod_{j=1}^{k} g(y_1^{t,x_t,r_t}(\tau)\cdot y_i^{\tau,1,j}(T))e^{-\int_\tau^T r_1(y^{\tau,1,1})(s)\mathrm{d}s}|\mathcal{F}_\tau\right]\right]. \tag{152}$$

Proceeding as before by i.i.d. property and martingale property of v.

$$= \mathbb{E}_{t,x}\left[I_{\tau\leq T}\sum_{k=0}^{m} a_k e^{-\int_t^\tau r(y^{t,x_t,r_t})(s)\mathrm{d}s}v(\tau,X_\tau)^k\right]. \tag{153}$$

Adding the two terms together gives:

$$\hat{u}(t, X_t) = \mathbb{E}_{t,x}\left[g(X_T)e^{-\int_t^T r(s)\mathrm{d}s}\right]e^{-\beta(T-t)} + \mathbb{E}_{t,x}\left[I_{\tau<T}e^{-\int_t^\tau r(s)\mathrm{d}s}\sum_{k=0}^{m} a_k v^k(\tau, X_\tau)\right]. \tag{154}$$

Identical to (146), thus $\hat{u}$ solves the type II PDE (142) and a product expression is found incorporating discounting to solve PDE's (125) and (142) of type I and type II.

7 Analytical Results

In this section analytical solutions of u as function of v, with u and v as in the previous section, will be derived in the case of one sided payoffs, $g \geq 0$ for both type I and type II. These can be used in combination with analytic solutions of v when available to get exact analytical solutions for type I and type II problems to evaluate the results of the branching algorithms. Note that if $g \leq 0$, a one sided negative payoff at maturity T, $u = v$.

7.1 Type I

For type I PDE's with one sided positive payoffs the polynomial approximation of F_I can be simplified to Ru, since $Ru = -u^- + Ru$ if $u \geq 0$. Consider the stochastic representation of the type I PDE (125):

$$u(t, X_t) = \mathbb{E}_{t,x}\left[g(X_T)e^{-\int_t^T r(s)\mathrm{d}s}\right]e^{-\beta(T-t)} + \mathbb{E}_{t,x}\left[I_{\tau<T}e^{-\int_t^\tau r(s)\mathrm{d}s}F_I(u_\tau)\right]. \tag{155}$$

Recognize the first part of the above as the stochastic representation of v as defined in (142).

$$u(t, X_t) = v(t, X_t)\mathbb{E}_{t,X_t}\left[I_{\tau_1>T}\right] + \mathbb{E}_{t,X_t}\left[I_{\tau_1<T}e^{-\int_t^{\tau_1} r(s)\mathrm{d}s}Ru(\tau_1, X_{\tau_1})\right]. \tag{156}$$

Where $\tau_i = t + \sum_{j=1}^i \tau_i'$ with τ_i' i.i.d. with an exponential distribution with parameter β. Continuing with the iteration first used in Section 3 and using the martingale property of discounted v results in:

$$\begin{aligned}
&= v(t, X_t)\mathbb{E}_{t,X_t}\left[I_{\tau_1>T}\right] + R\mathbb{E}_{t,X_t}\left[I_{\tau_1<T}e^{-\int_t^{\tau_1} r(s)\mathrm{d}s}u(\tau_1, X_{\tau_1})\right], \qquad (157)\\
&= v(t, X_t)\mathbb{E}_{t,X_t}\left[I_{\tau_1>T}\right] + R\mathbb{E}_{t,X_t}\left[I_{\tau_1<T}e^{-\int_t^{\tau_1} r(s)\mathrm{d}s}\left(v(\tau_1, X_{\tau_1})\mathbb{E}_{\tau_1,X_{\tau_1}}\left[I_{\tau_2>T}\right] + R\mathbb{E}_{\tau_1,X_{\tau_1}}\left[I_{\tau_2<T}e^{-\int_{\tau_1}^{\tau_2} r(s)\mathrm{d}s}u(\tau_2, X_{\tau_2})\right]\right)\right],\\
&= v(t, X_t)\mathbb{E}_{t,X_t}\left[I_{\tau_1>T}\right] + Rv(t, X_t)\mathbb{E}_{t,X_t}\left[I_{\tau_2>T}\right] + R^2\mathbb{E}_{t,X_t}\left[I_{\tau_2>T}u(\tau_2, X_{\tau_2})\right],\\
&= \sum_{i=0}^{\infty} R^i v(t, X_t)\mathbb{E}_{t,X_t}\left[I_{\tau_i<T<\tau_{i+1}}\right].
\end{aligned}$$

Note that $\mathbb{E}_{t,X_t}\left[I_{\tau_i<T<\tau_{i+1}}\right]$ equals a Poisson distribution with parameter $\beta(T-t)$.

$$u(t, X_t) = v(t, X_t)\mathbb{E}\left[R^p\right]. \tag{158}$$

Where p is Poisson distributed with parameter β. Rewriting gives the definition of the characteristic function ϕ_P evaluated at $-i\log(R)$. Using the known characteristic function of a Poisson variable gives:

$$\begin{aligned}
\mathbb{E}\left[R^p\right] &= \mathbb{E}\left[e^{\log(R)p}\right], \qquad (159)\\
&= \phi_p(-i\log(R)),\\
&= e^{\beta(T-t)\left(e^{i(-i\log(R))}\right)},\\
&= e^{\beta(T-t)(R-1)}.
\end{aligned}$$

Applying the above gives the following if g is a nonnegative payoff function at maturity:

$$u(t, X_t) = e^{\beta(T-t)(R-1)}v(t, X_t). \tag{160}$$

7.2 Type II

Similarly to the previous section, type II PDE's with one sided positive payoffs allow the polynomial approximation of F_{II} to be simplified to Rv, since $Rv = -v^- + Rv$ if $u \geq 0$. Consider the stochastic representation of the type II PDE (142):

$$u(t, X_t) = \mathbb{E}_{t,x}\left[g(X_T)e^{-\int_t^T r(s)\mathrm{d}s}\right]e^{-\beta(T-t)} + \mathbb{E}_{t,x}\left[I_{\tau<T}e^{-\int_t^\tau r(s)\mathrm{d}s}F_{II}(v_\tau)\right]. \tag{161}$$

Recognize the first part as the stochastic representation of v as defined in (142) and integrate against the density of τ in the second part.

$$\begin{aligned} u(t, X_t) &= \mathbb{E}_{t,x}\left[g(X_T)e^{-\int_t^T r(s)\mathrm{d}s}\right]e^{-\beta(T-t)} + \mathbb{E}_{t,x}\left[I_{\tau<T}e^{-\int_t^\tau r(s)\mathrm{d}s}Rv(\tau, X_\tau)\right], \\ &= v(t, X_t)e^{-\beta(T-t)} + R\mathbb{E}_{t,x}\left[\int_t^T \beta e^{-\beta(z-s)}e^{-\int_t^z r(s)\mathrm{d}s}v(s, X_s)\mathrm{d}z\right]. \end{aligned} \tag{162}$$

Assuming g is bounded as before, integration and expectation can be interchanged by boundedness of the inner term.

$$\begin{aligned} u(t, X_t) &= v(t, X_t)e^{-\beta(T-t)} + R\int_t^T \mathbb{E}_{t,x}\left[\beta e^{-\beta(z-s)}e^{-\int_t^z r(s)\mathrm{d}s}v(s, X_s)\right]\mathrm{d}z, \\ &= v(t, X_t)e^{-\beta(T-t)} + R\int_t^T \beta e^{-\beta(z-s)}\mathbb{E}_{t,x}\left[e^{-\int_t^z r(s)\mathrm{d}s}v(s, X_s)\right]\mathrm{d}z. \end{aligned} \tag{163}$$

Now use the martingale property of $e^{-\int_t^z r(s)\mathrm{d}s}v(s, X_s)$:

$$\begin{aligned} u(t, X_t) &= v(t, X_t)e^{-\beta(T-t)} + R\int_t^T \beta e^{-\beta(z-s)}v(t, X_t)\mathrm{d}z, \\ &= v(t, X_t)e^{-\beta(T-t)} + Rv(t, X_t)\int_t^T \beta e^{-\beta(z-s)}v(t, X_t)\mathrm{d}z, \\ &= v(t, X_t)e^{-\beta(T-t)} + Rv(t, X_t)(1 - e^{-\beta(T-t)}), \\ &= v(t, X_t)\left(e^{-\beta(T-t)} + R(1 - e^{-\beta(T-t)})\right). \end{aligned} \tag{164}$$

7.3 Exact Solutions for the Constant and Hull White Interest Rate Models

The analytic expressions of u as a function of v found in the previous section are only useful if v can be found analytically or via an efficient simulation. In the case of a constant or Hull White rate process v has an exact solution for several common derivatives. These exact solutions are given below. Where a binary call pays out 0 or 1 at maturity, $I_{S>K}$:

$$\begin{aligned} \text{Call:} \quad v^C(t_0, S_0, K) &= S_0\mathcal{N}(d_1) - Ke^{-r(T-t_0)}\mathcal{N}(d_2), \\ \text{Put:} \quad v^P(t_0, S_0, K) &= Ke^{-r(T-t_0)}\mathcal{N}(-d_2) - S_0\mathcal{N}(-d_1), \\ \text{Binary Call:} \quad v^{BC}(t_0, S_0, K) &= e^{-r(T-t_0)}\mathcal{N}(d_2), \\ \text{Binary Put:} \quad v^{BP}(t_0, S_0, K) &= e^{-r(T-t_0)}\mathcal{N}(-d_2). \end{aligned} \tag{165}$$

Where:

$$\begin{aligned} d_1 &= \frac{\log(S/K) + r}{\sigma\sqrt{T - t_0}} + \frac{1}{2}\sigma\sqrt{T - t_0}, \\ d_2 &= \frac{\log(S/K) + r}{\sigma\sqrt{T - t_0}} - \frac{1}{2}\sigma\sqrt{T - t_0}, \\ &= d_1 - \sigma\sqrt{T - t_0}. \end{aligned} \tag{166}$$

Hull White:

$$
\begin{aligned}
\text{Call:} \quad v^C(t_0, S_0, K) &= S_0\mathcal{N}(d_1) - KP(r_0, t_0)\mathcal{N}(d_2), \\
\text{Put:} \quad v^P(t_0, S_0, K) &= KP(r_0, t_0)\mathcal{N}(-d_2) - S_0\mathcal{N}(-d_1), \\
\text{Binary Call:} \quad v^{BC}(t_0, S_0, K) &= P(r_0, t_0)\mathcal{N}(d_2), \\
\text{Binary Put:} \quad v^{BP}(t_0, S_0, K) &= P(r_0, t_0)\mathcal{N}(-d_2).
\end{aligned}
\tag{167}
$$

Where:

$$
\begin{aligned}
d_1 &= \frac{\log(S/K) - \log(P(r_0, t_0))}{\sqrt{\sigma^2 + \eta^2}\sqrt{T - t_0}} + \frac{1}{2}\sqrt{\sigma^2 + \eta^2}\sqrt{T - t_0}, \\
d_2 &= d_1 - \sqrt{\sigma^2 + \eta^2}\sqrt{T - t_0}, \\
A(t_0, T) &= \exp\left(\theta(t_0 - T) - \frac{\eta^2}{2\kappa^2}\left((t_0 - T) - \frac{2}{\kappa}(e^{\kappa(t_0 - T)} - 1) + \frac{1}{2\kappa}(e^{2\kappa(t_0 - T)} - 1)\right)\right), \\
B(t_0, T) &= \frac{1 - e^{\kappa(t_0 - T)}}{\kappa}, \\
P(r_0, t_0) &= A(t_0, T)e^{-(r_0 - \theta)B(t_0, T)}.
\end{aligned}
\tag{168}
$$

Note that even if an analytic solution is not available it is often easier to find an expression for v than for u, therefore the results in this section should be used to compute u in the case of a non negative payoff function g for types I and II.

8 Implementation

This section should be considered as a short note on the implementation without going into to much detail. The choice for Matlab and C++ is motivated as well as some of the challenges faced in the implementation.

programming structure
multithreading
whole pricing surface at once
bounding of g to a and approximating u at interval should be done, current implementation not effectet by this because of small probability of mass ending up in region, propability decreases exponentialy while payoff increases polynominally.

8.1 Matlab

The estimation of the coefficients of the approximating polynomials $P_n, \underline{P}_n$ and $\overline{P}_n$ are implemented in Matlab. because of the results in Section **??** this only has be computed once to get the correct coefficients after that these can be scaled for any interval needed. This is done from an ease of implementation perspective. Matlab can easily compute and solve the algebraic equations needed in (112) while its matrix and quadprog implementations allow for the minimizations to be computed fast and correct.

8.2 C++

The remainder of the implementation has been done in C++. The reasons for this are threefold: Firstly yours truly wanted to expand his programming knowledge and learn the financial industry standard language. Secondly the object orientated nature of C++ lends itself well for the implementation of branch and bound type algorithms. Also the speed increase compared to Matlab is a welcome benefit since branch and bound type algorithms typically require large amounts of non trivial simulations. A disadvantage of C++ is its complexity and learning curve, especially grasping and applying the concepts of object based programming. However multiple sources exist to aid on during this process, for example [Koenig and Moo, 2000] [Lippman, 2002] and [Meyers, 2005].

8.3 Functionality

The Mersennene Twister *mt19937*, part of the standard library in C11, is used for random number generation. Also note that a whole pricing surface (even for multiple derivative payoffs) can be computed simultaneously by pausing the branching particle processes at the relevant strike times and evaluating the payoffs. Almost all implementations of the evolving random processes are implemented using an Euler forwards discretization technique [Seydel, 2009]. The OpenMP framework is used to implement the multi threaded Monte Carlo simulations.

9 Results

The extended branching diffusion algorithm from Section 6 is implemented for both type I and type II PDE's using a constant, Hull White or a CIR interest rate process. The derivatives used are a put, call, binary put, binary call and a forward agreement. If an analytic solution is available it is computed as well as a plot of the errors made in the pricing of the derivative. A plot of the standard deviation of the branching algorithm is given as well. A bound for the error introduced by the polynomial approximation P_n of F is estimated by computing and subtracting the results generated by the extended branching diffusion algorithm implemented with $\overline{P_n}$ and $\underline{P_n}$. The results using the Hull White interest rate model will be discussed in this section, while the results from the constant and CIR rate processes are given in Appendix B.

9.1 Parameters

All results are generated using approximating polynomials of order 15 on the interval [-3,3] and the following common parameters, similar to market parameters:

$$\begin{aligned} \beta &= 0.05, && \text{Default rate,} \\ R &= 0.4, && \text{Recovery rate,} \\ \delta t &= 0.025, && \text{Timestep used in Euler discretization,} \\ n &= 10^6, && \text{Number of Monte Carlo simulations of the extended marked branching diffusion process,} \\ t &= 0, && \text{Starting time,} \\ X_0 &= 1, && \text{Location of (normalized) underlying at inception,} \\ \sigma &= 0.2, && \sigma \text{ of underlying } X. \end{aligned} \tag{169}$$

9.2 Hull White Rates

The Hull White interest rate model is used to discuss the results because it is stochastic but still allows for the analytic solutions of Section 7, and all relevant findings are observable.

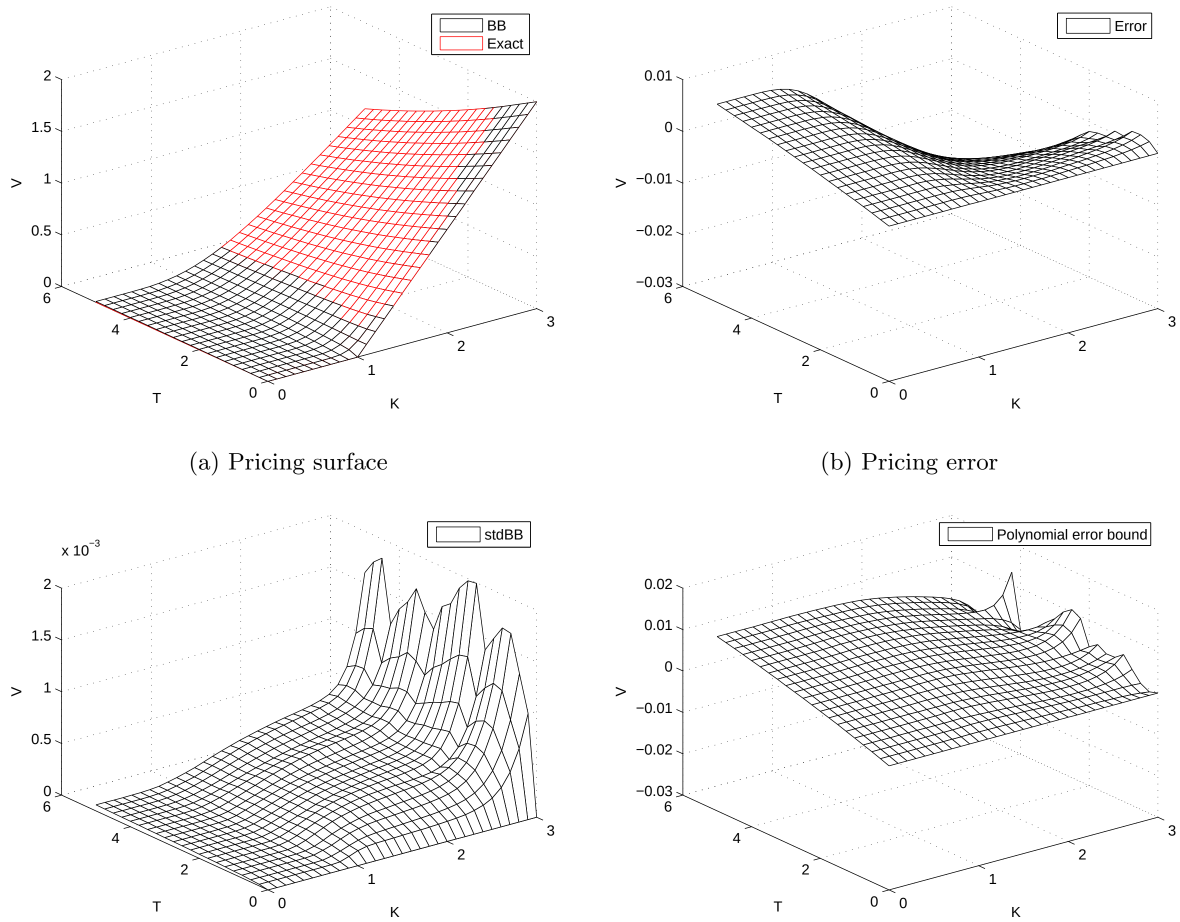
9.2.1 Parameters

The Hull White interest rate models has the following specific parameters:

$$\begin{aligned} r_0 &= 0.05 && \text{Interest rate at inception,} \\ \kappa &= 0.2 && \text{Mean reverting parameter,} \\ \theta &= 0.05 && \text{Long term mean,} \\ \sigma_r &= 0.05^{1.5} && \sigma \text{ used for the dynamics of the rate process.} \end{aligned} \tag{170}$$

9.2.2 Type I

First the results in the case of a type I default convention are examined. A pricing surface of a put with maturity T and strike value K is given below. The analytic solution and the extended marked branching diffusion solution are given. Next to that the error is plotted on the same grid. The third picture shows the standard deviation of the extended marked branching diffusion results. While the lower right plot is of the polynomial bounds introduced by the approximation of F_I. After that the same figure is give for a call in Figure 5.



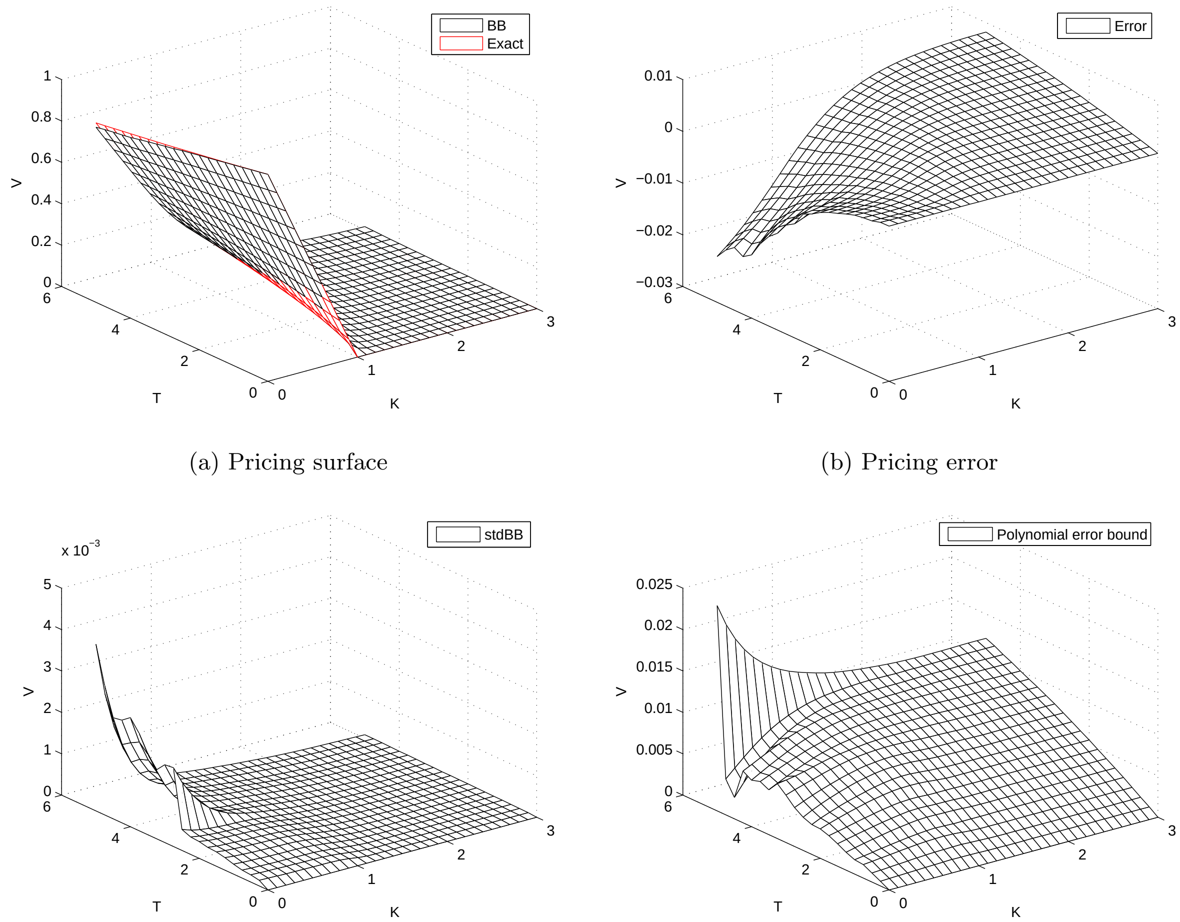
(a) Pricing surface

(b) Pricing error

(c) Standard deviation

(d) Max polynomial error

Figure 4: Put



(a) Pricing surface

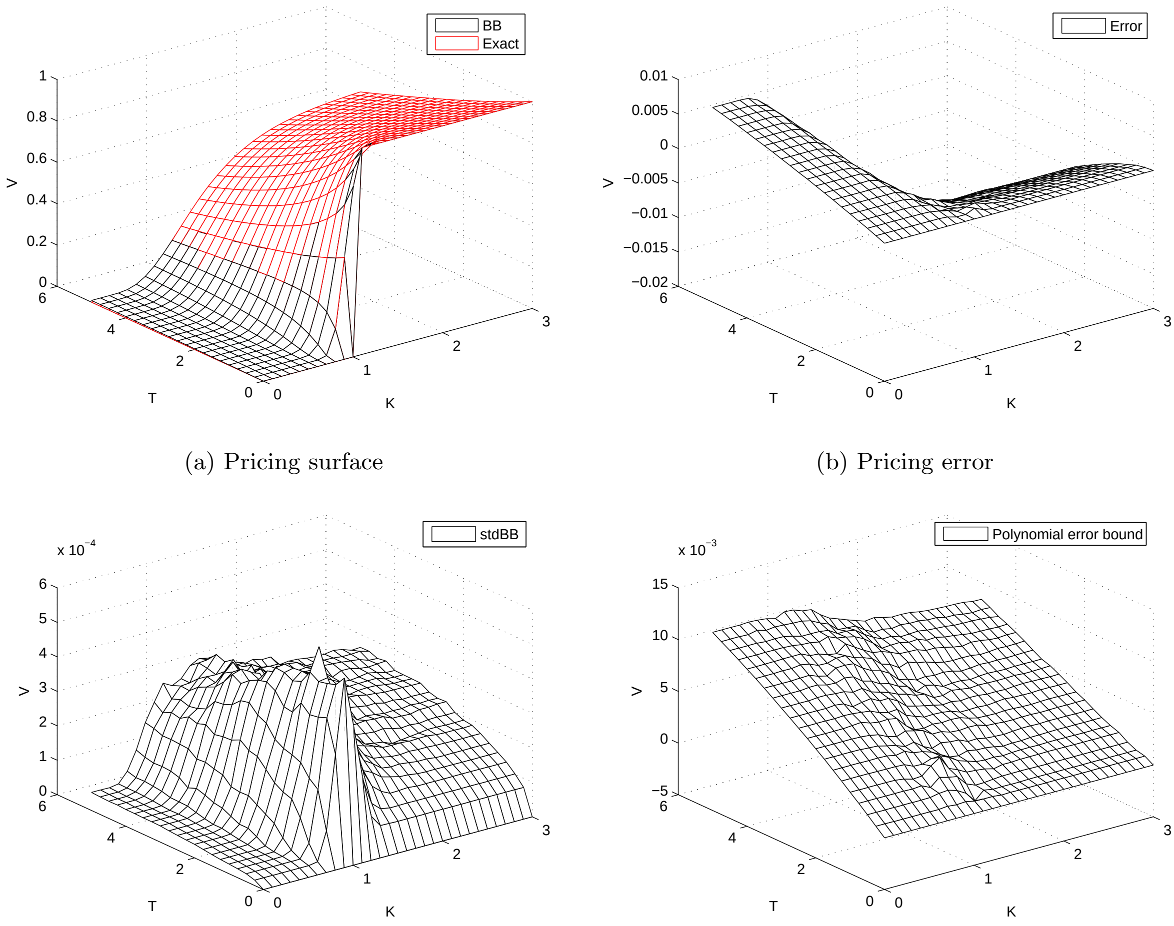
(b) Pricing error

(c) Standard deviation

(d) Max polynomial error

Figure 5: Call

The performance of the branching diffusion algorithm is not bad, notice how the pricing surface is estimated to within approximately 0.02. There is a consistent error when the pricing surface is close to zero. The standard deviation of the results increases a the price increases this is to be expected since the branching diffusion algorithm depends on evaluating a product of payoffs. Also notice that the polynomial bounds give a good upper bound for the error made if the standard deviation is not to high (since the lower and upper approximation suffer from a similar standard deviation. The next two figures are of a binary put and binary call



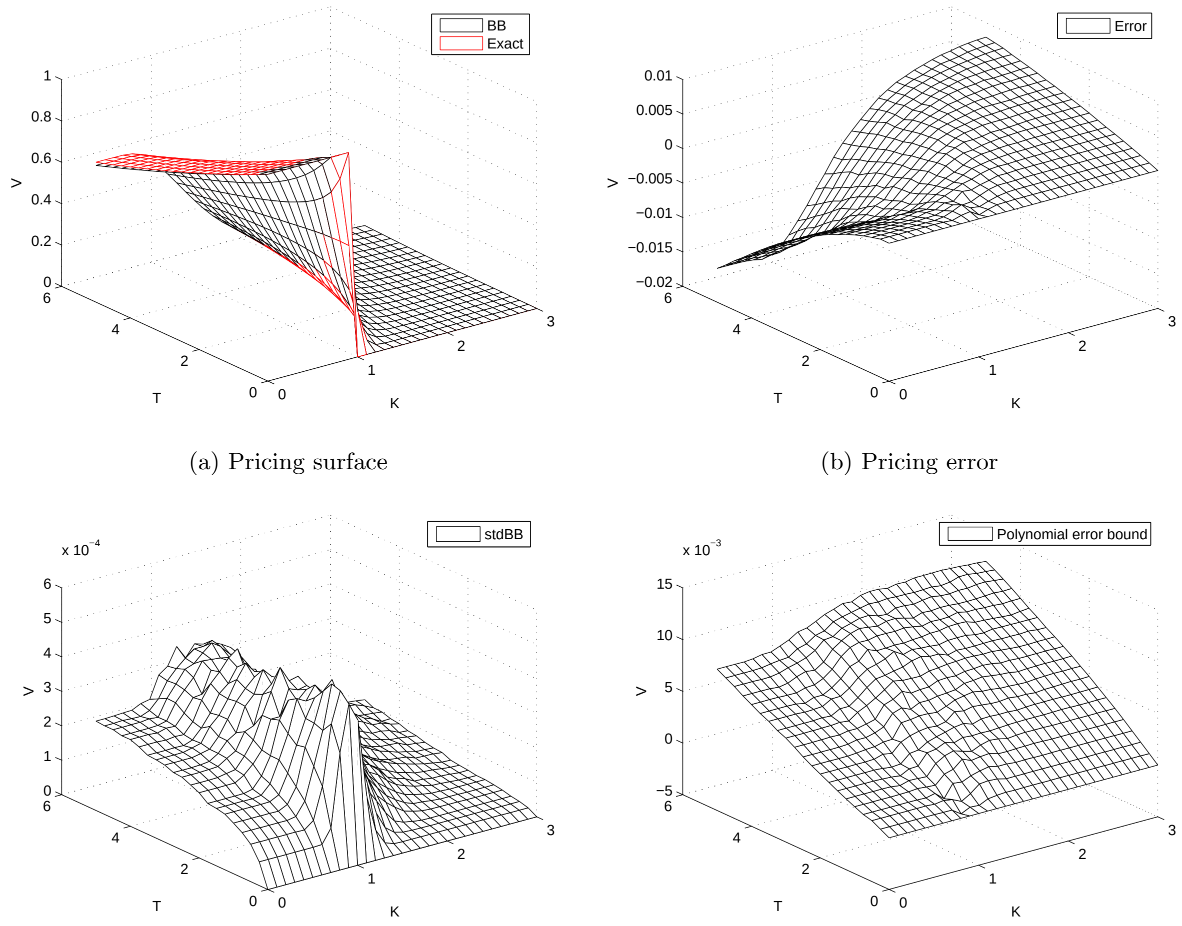
(a) Pricing surface

(b) Pricing error

(c) Standard deviation

(d) Max polynomial error

Figure 6: Binary Put



(a) Pricing surface

(b) Pricing error

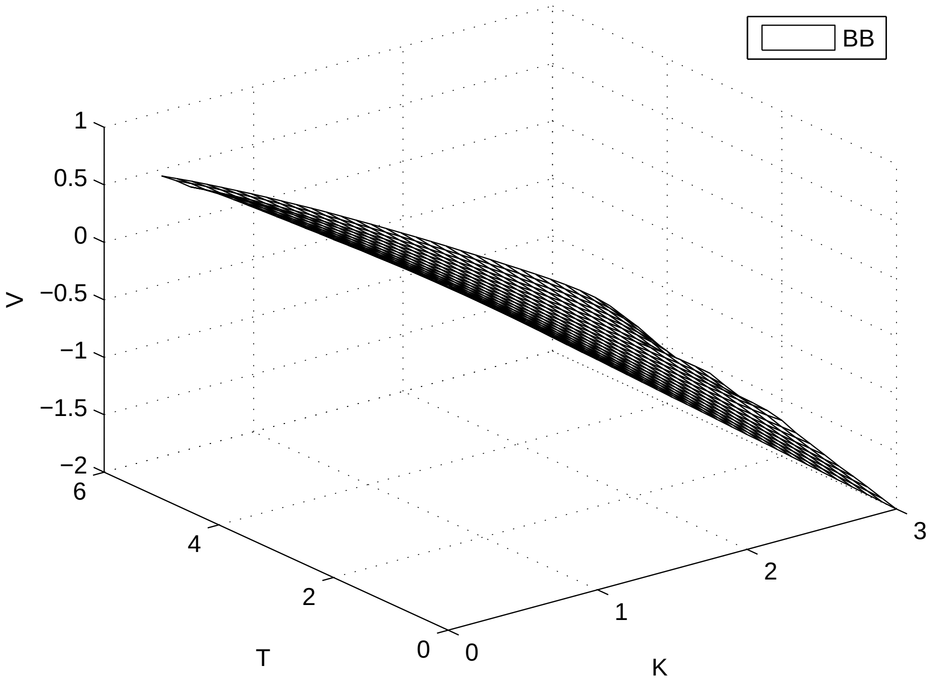
(c) Standard deviation

(d) Max polynomial error

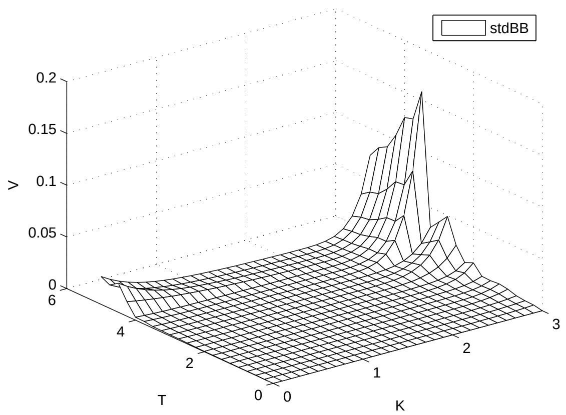
Figure 7: Binary Call

The performance for the binary call and binary put are even better than for the normal put and call. This is to be expected since their payoff is bounded by one. The error and the standard deviation of the branching diffusion algorithm are very low. The polynomial error is again an upper bound of the error made and is larger in the region where the price is close to zero.

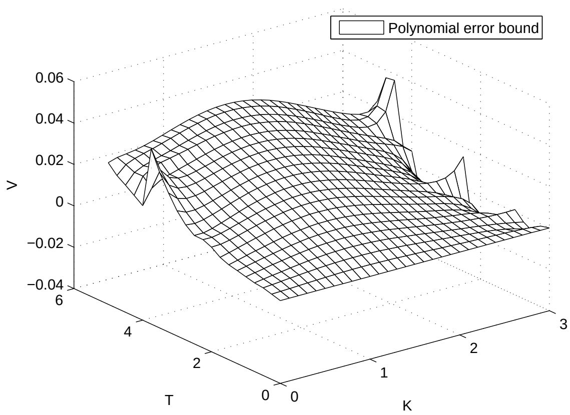
Results for the forward are given in figure 8. No analytic solution is known, however the performance seems similar to the results for the call and put. the polynomial error bound is of similar size and takes on its larges values around the area where the price is cloe to zero. Again the standard deviation of the estimates increases as the price increases in absolute value.



(a) Pricing surface



(b) Standard deviation

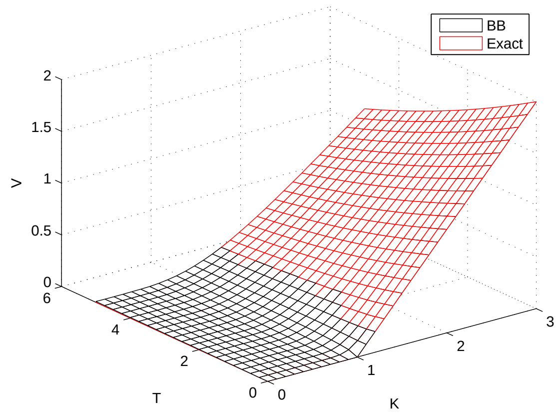


(c) Max polynomial error

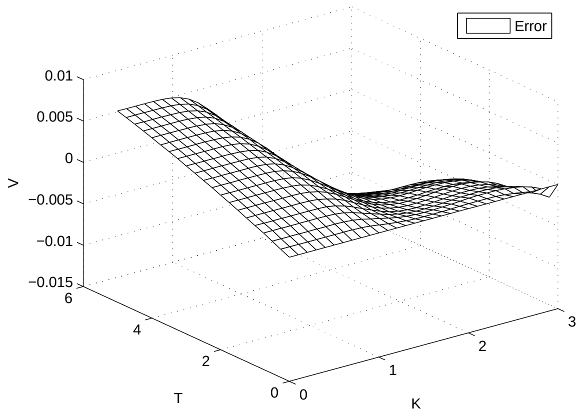
Figure 8: Forward

9.2.3 Type II

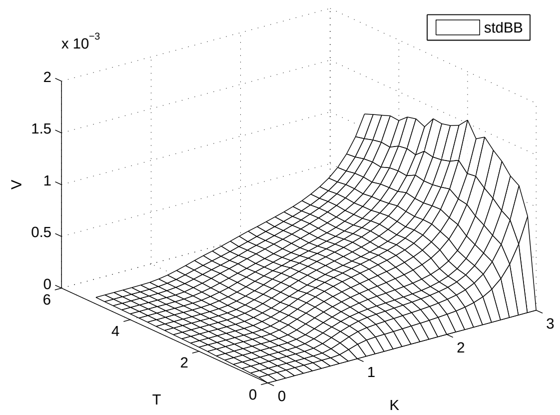
The results for the pricing of derivatives including a type II CVA are given below. The findings and errors are very similar to the type I results discussed before. The main difference compared to the type I results are the slightly higher valuations as expected, since a default nets a portion of a risk free derivative in stead of a risky derivative.



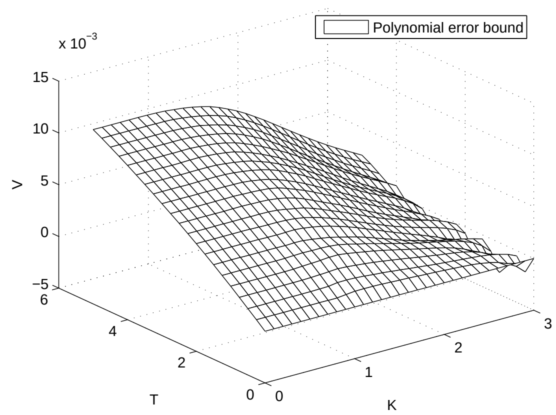
(a) Pricing surface



(b) Pricing error

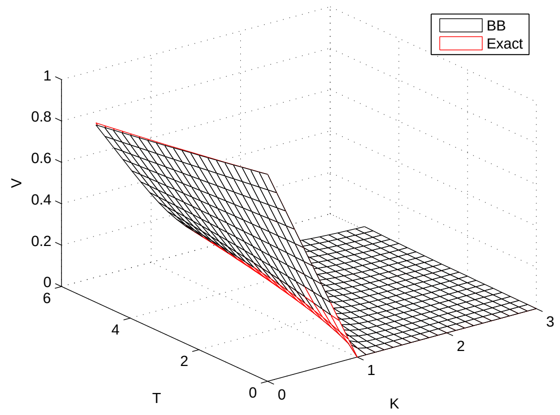


(c) Standard deviation

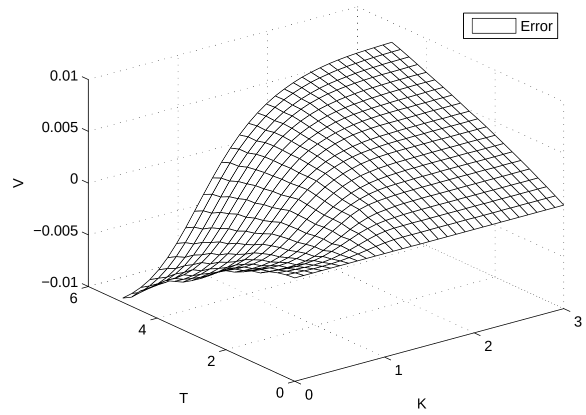


(d) Max polynomial error

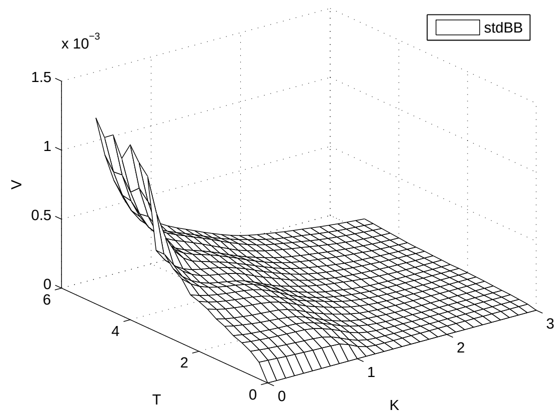
Figure 9: Put



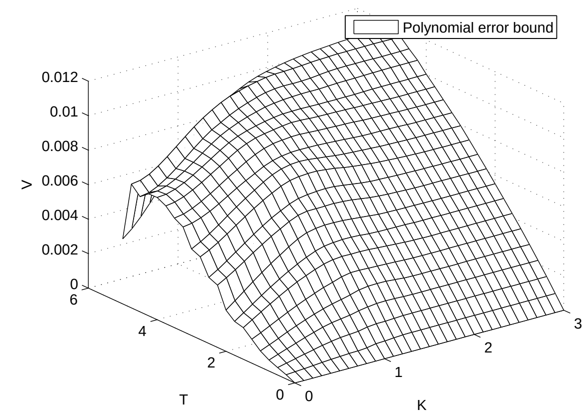
(a) Pricing surface



(b) Pricing error

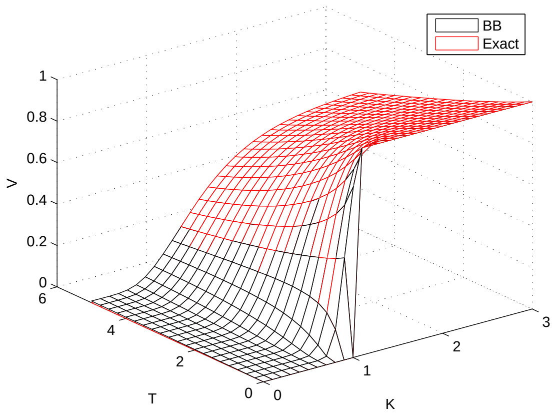


(c) Standard deviation

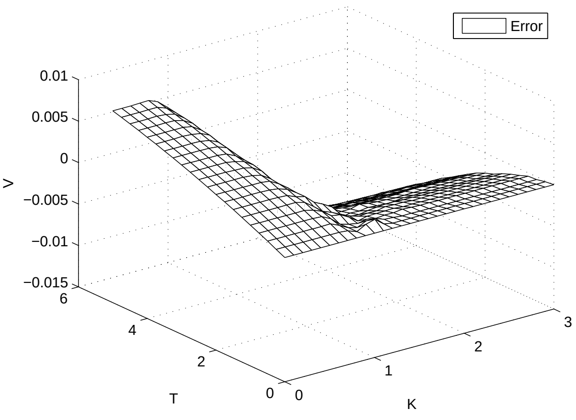


(d) Max polynomial error

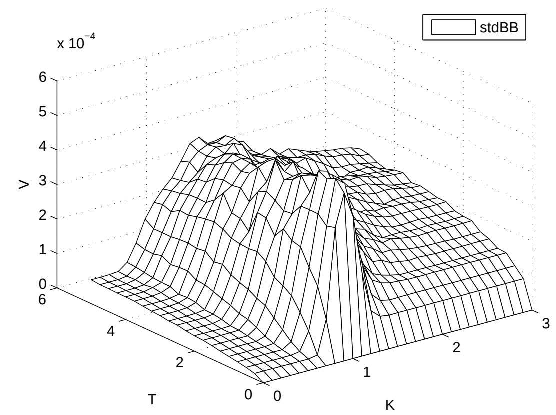
Figure 10: Call



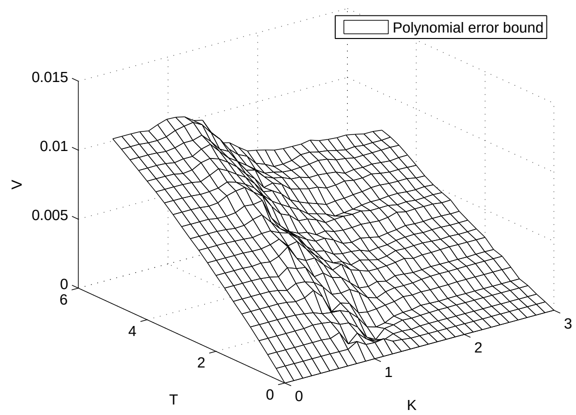
(a) Pricing surface



(b) Pricing error

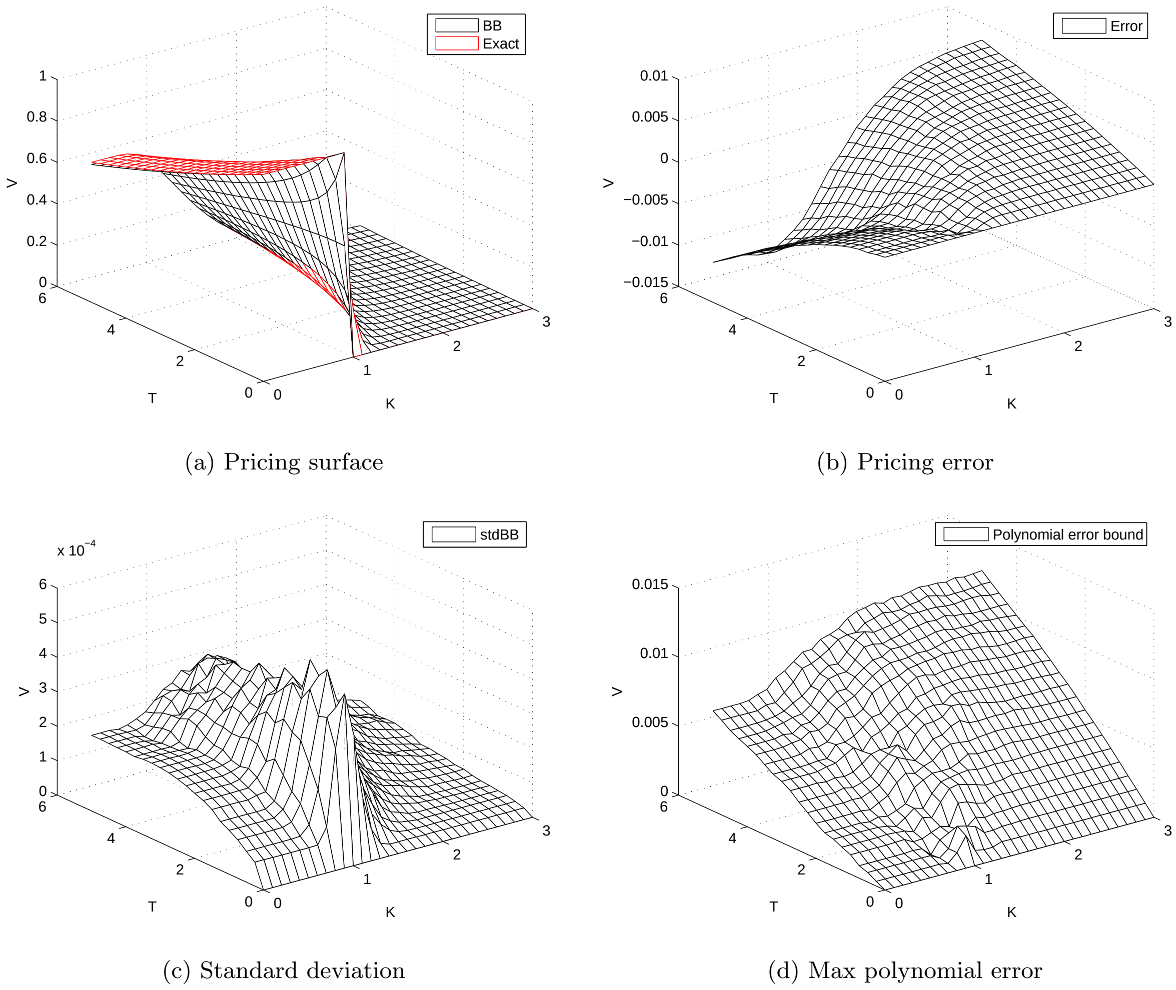


(c) Standard deviation



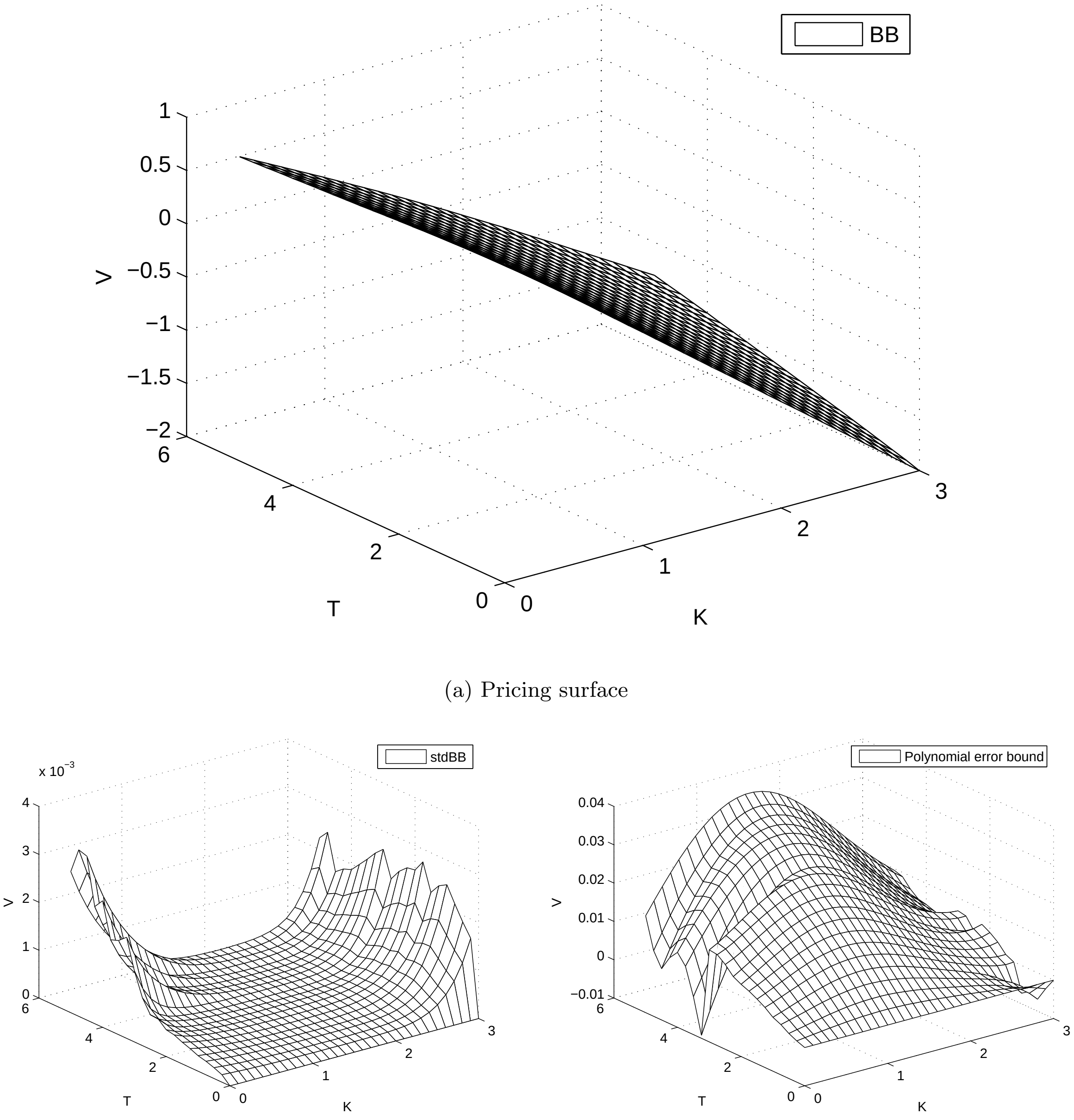
(d) Max polynomial error

Figure 11: Binary Put



(a) Pricing surface
(b) Pricing error
(c) Standard deviation
(d) Max polynomial error

Figure 12: Binary Call



(a) Pricing surface

(b) Standard deviation

(c) Max polynomial error

Figure 13: Forward

10 Conclusions

The inclusion of CVA when pricing OTC derivatives without collateral postings is needed in current markets. Using the type I and type II default agreements[Brigo and Morini, 2011] the marked branching diffusion algorithm [Henry-Labordére, 2012] is extended to include stochastic rate processes. The research questions from the introduction are repeated below and answered individually.

Can the marked branching diffusion approach be used to price deals including CVA?
Yes, if Type I or type II default assumptions are made the algorithm will generate credible derivative values. However if the payoff at maturity is nonnegative or nonpositive the analytical results from Section 7 should be used since finding the price of a derivative without CVA is often a much simpler task. Also great care should be taken when pricing derivatives with possible large sections of the pricing surface close to zero. The polynomial approximation struggles with the discontinuity of the derivative of x^+ at $x = 0$. This can lead to an incorrect increases in value of a derivative by adding CVA. An estimate of the maximum effect of this can be derived by looking at the difference in valuation with the use of a polynomial upper and lower approximation of the payoff at default. However these bounds are the result of branching diffusion process themselves so should only be used when the process has converged sufficiently.

Can the marked branching diffusion approach be extended to include stochastic interest rates?
Yes, the method can be extended to include stochastic rates by "equipping" each particle with its own stochastic rate process and adapting the product of payoff functions to include discounting.

What are the advantages and disadvantages of using the marked branching diffusion approach?
The marked branching approach allows for valuation of derivatives of Type I with a possible two sided payoff without needing nested Monte Carlo iterations or nested numerical integration. In Type II situations with a two sided payoff it is comparable to a Monte Carlo method. Also the method works best when the payoff range of a derivative is relatively small and bounded. This allows for a good polynomial approximation used for the payoff at default. The complexity of the marked branching diffusion approach is somewhat of a disadvantage. It is harder to implement and understand, thus more prone to errors, than a relatively straightforward Monte Carlo approach where paths are simulated and payoffs evaluated.

10.1 Future research

Further research could look into extensions and implementation to other asset classes. As well as determining calibration methods to get reliable parameters. The method can easily be expanded to include multiple stochastic underlying sources of risk. This would allow the algorithms to be used on a portfolio of derivatives based on multiple underlying stocks with a single counterparty to get a market to market valuation of the whole portfolio with this counterparty. Path dependent payoffs have not been considered in this thesis, these could be included by assuming that the particles alive at maturity have "walked" the path of their parent particles. The frame work can also be extended to include stochastic default rates and possible modeling of counterparty stock. By introducing a correlation structure between the default rate, counterparty stock and underlying stock, more advanced dynamics can be replicated. As mentioned before DVA and funding costs (for both parties) can be added as well.

A Personal Note

This thesis is the result of my internship at ING within the MRM quant team. I am thankful for their support and feedback and I will miss the weekly puzzles and riddles. Special thanks are due to Dmytro and Jasper, my supervisors at ING and TUDelft without whom I would have been lost in direction and C++ programming. I am grateful for Teun's critical evaluation of my drafts and motivation during the more difficult times. Lastly I want to thank my wife Marijn without whom I would have never finished my thesis nor achieve anything.

B Results Continued

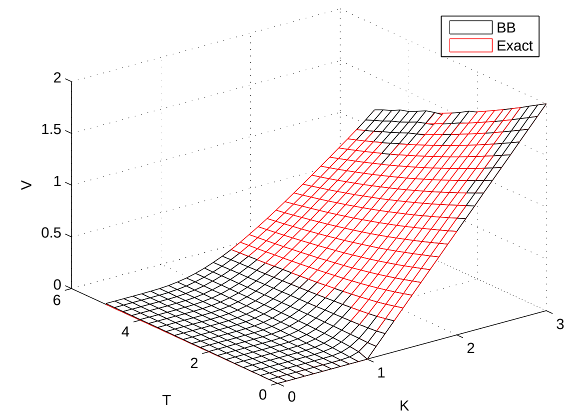
B.1 Constant Rates

B.1.1 Parameters

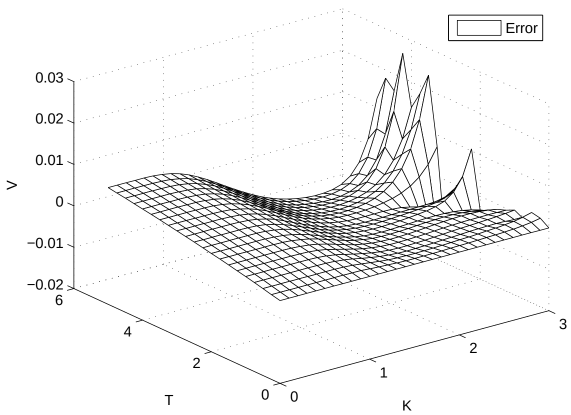
The constant interest rate models has the following specific parameters:

$$r_0 = 0.05 \qquad \text{Interest rate,} \qquad (171)$$
$$(172)$$

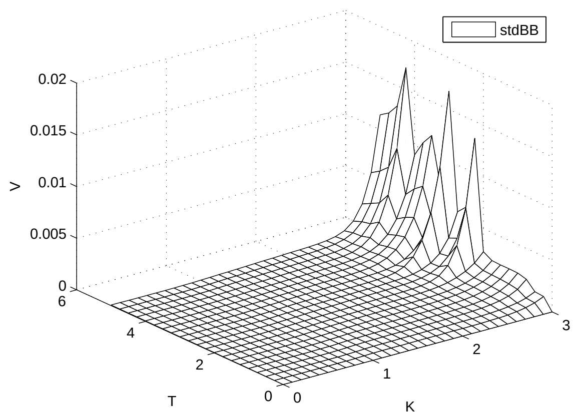
B.1.2 Type I



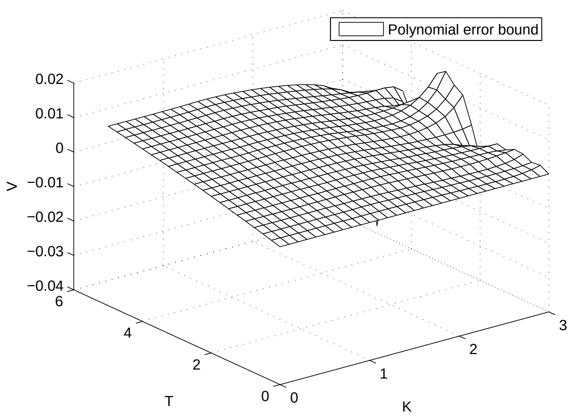
(a) Pricing surface



(b) Pricing error

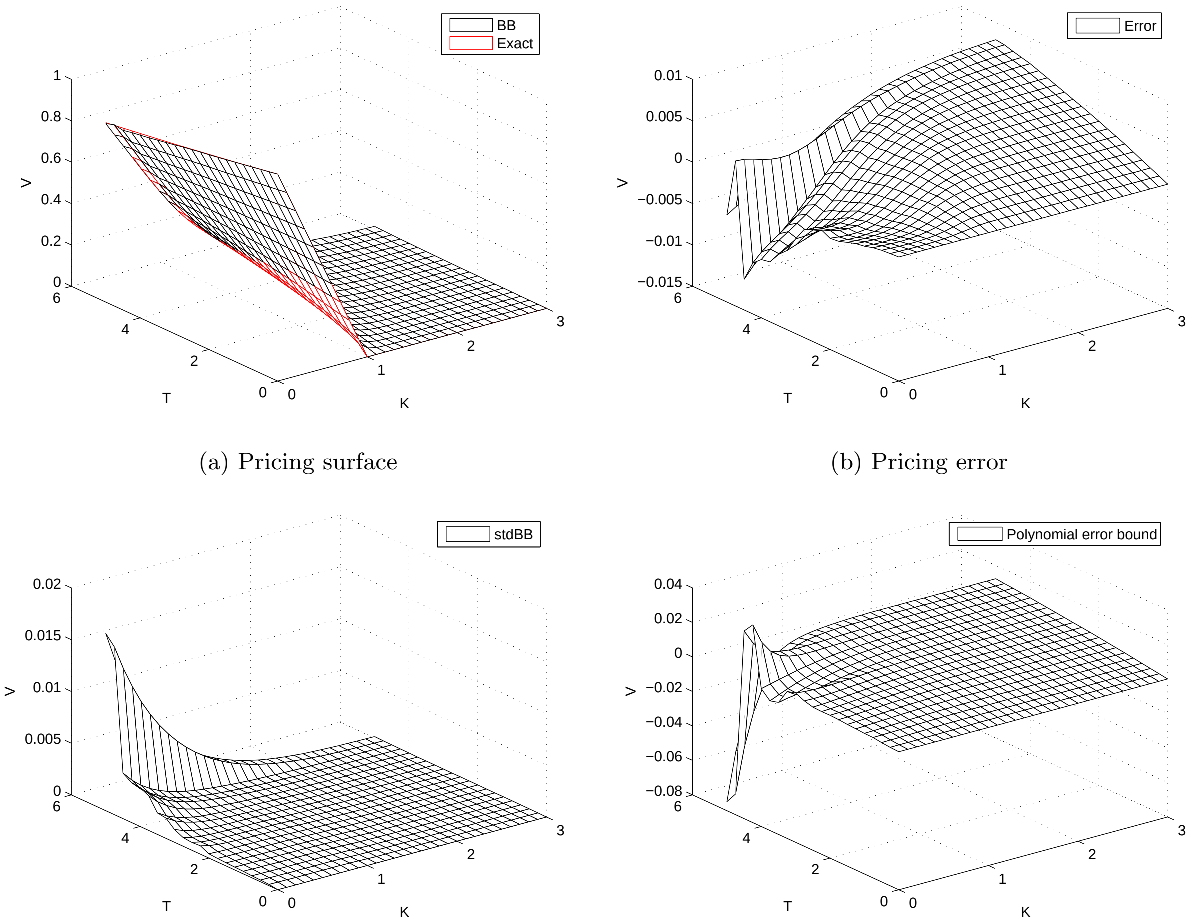


(c) Standard deviation



(d) Max polynomial error

Figure 14: Put



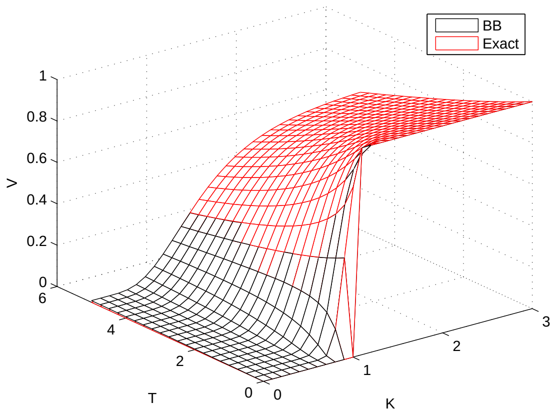
(a) Pricing surface

(b) Pricing error

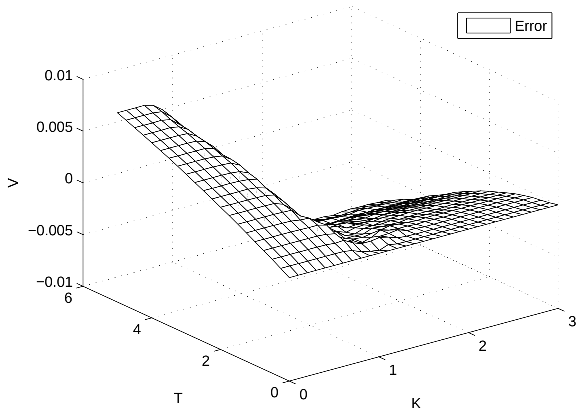
(c) Standard deviation

(d) Max polynomial error

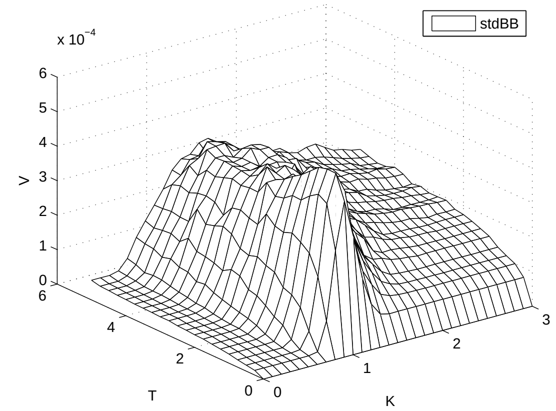
Figure 15: Call



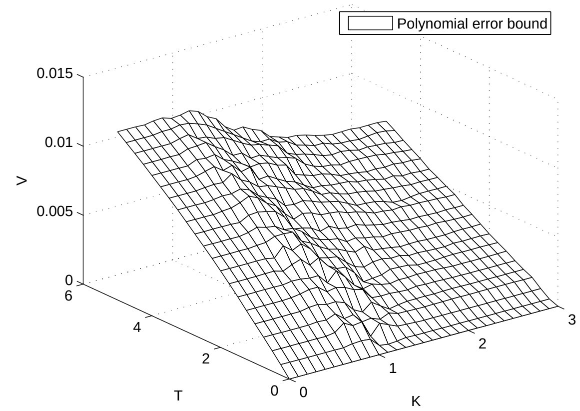
(a) Pricing surface



(b) Pricing error

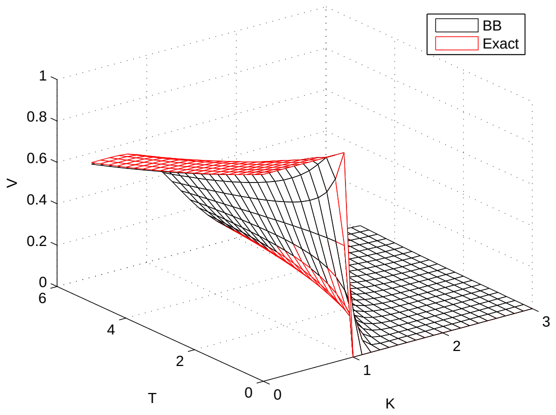


(c) Standard deviation

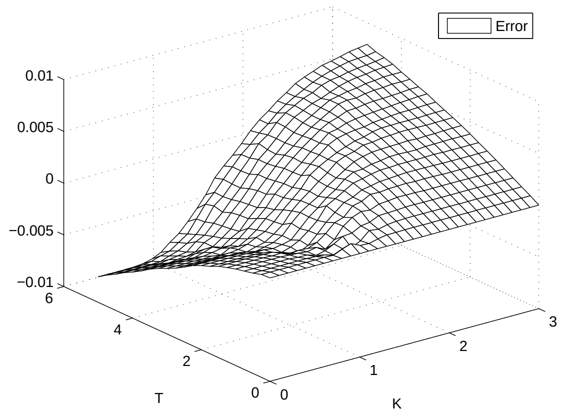


(d) Max polynomial error

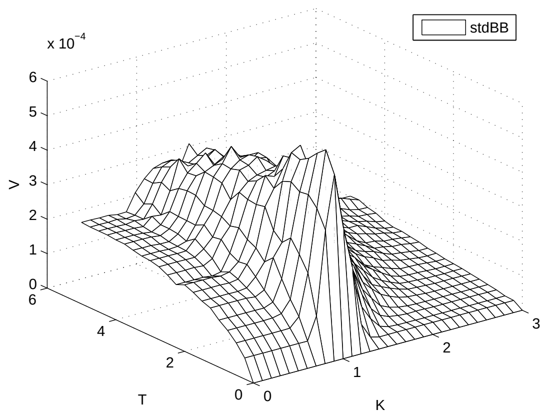
Figure 16: Binary Put



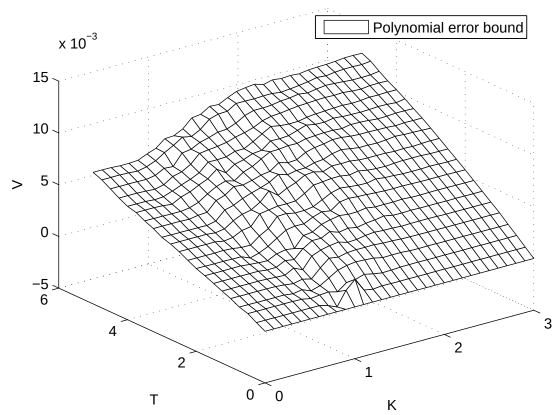
(a) Pricing surface



(b) Pricing error

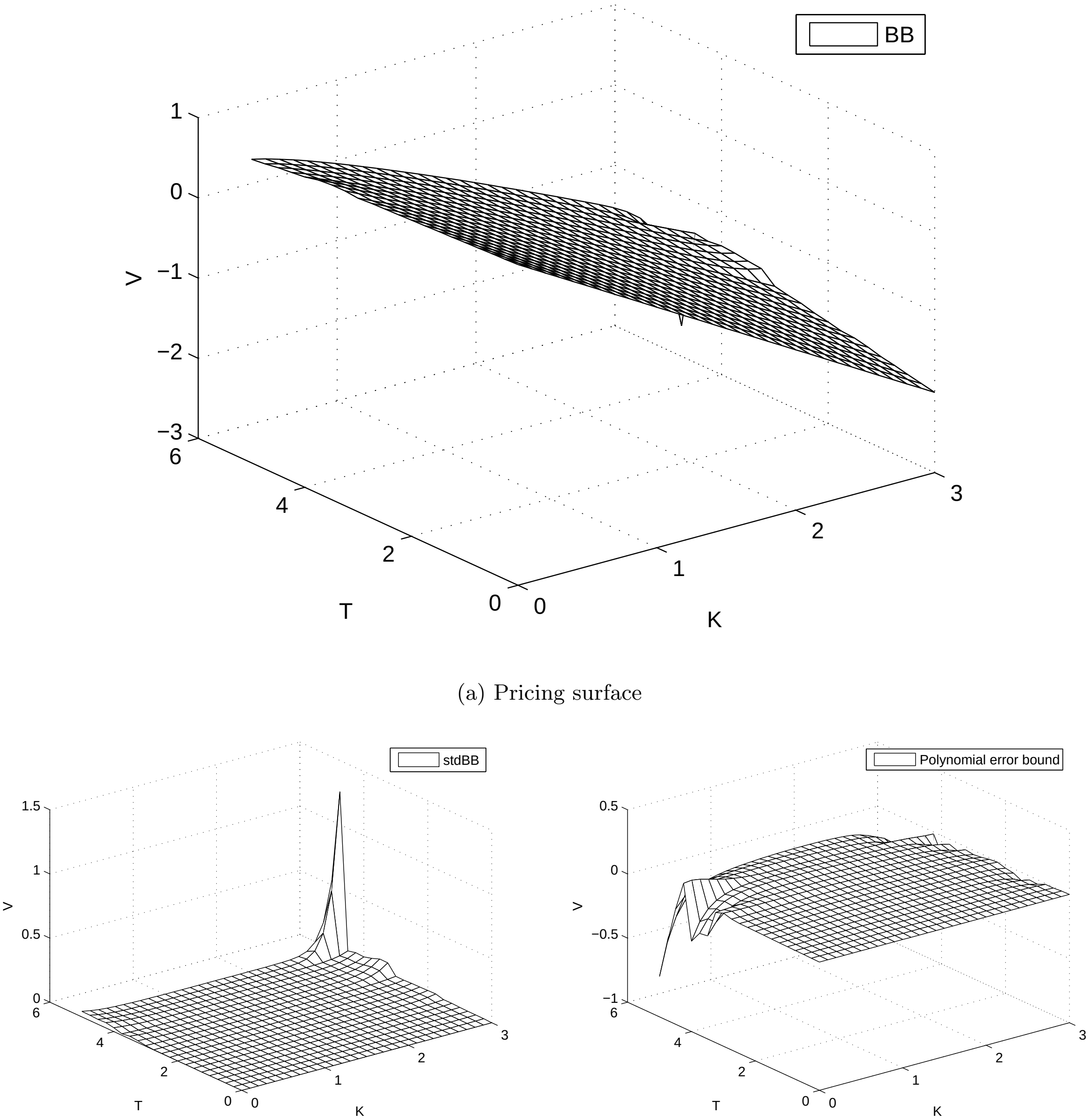


(c) Standard deviation



(d) Max polynomial error

Figure 17: Binary Call



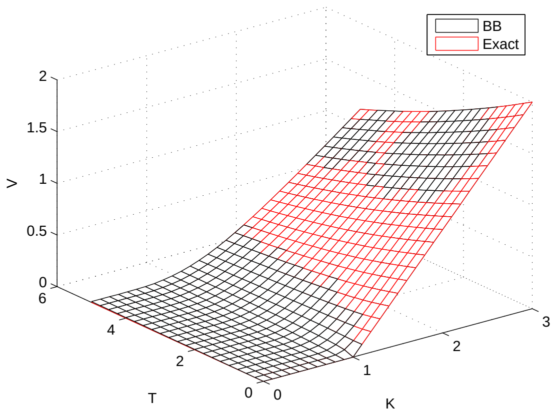
(a) Pricing surface

(b) Standard deviation

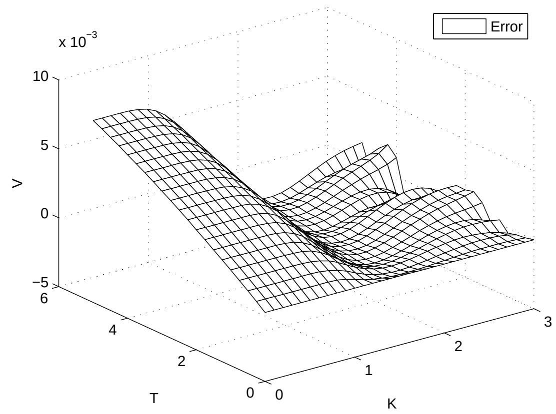
(c) Max polynomial error

Figure 18: Forward

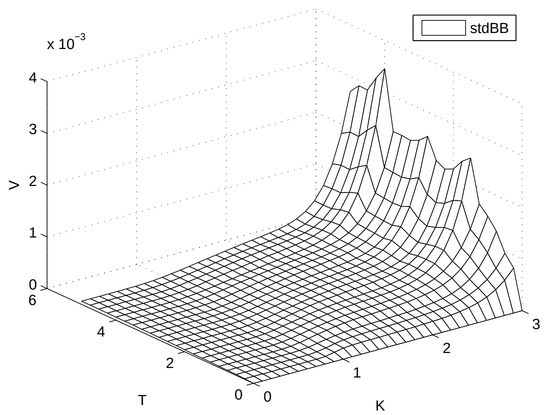
B.1.3 Type II



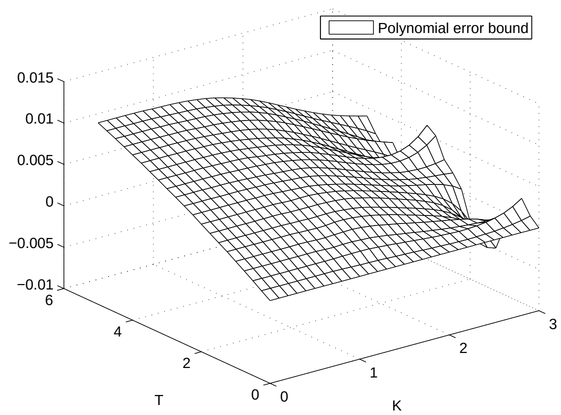
(a) Pricing surface



(b) Pricing error

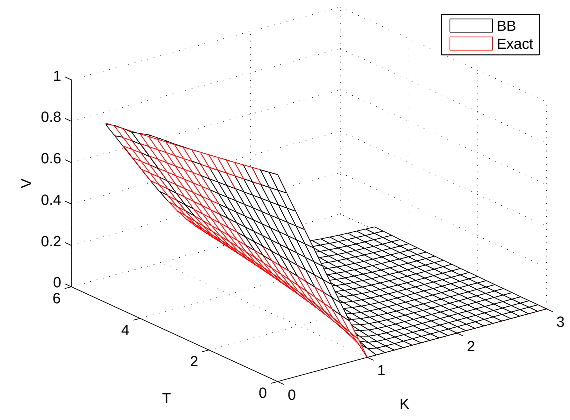


(c) Standard deviation

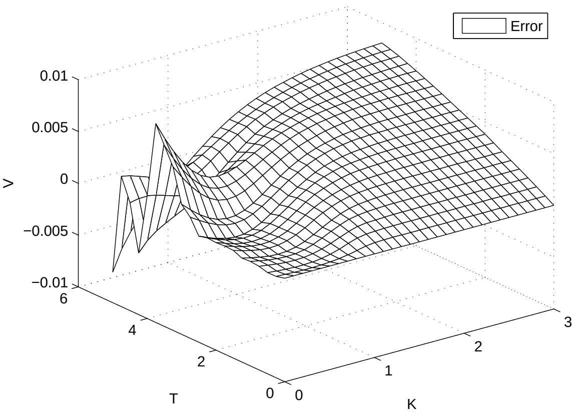


(d) Max polynomial error

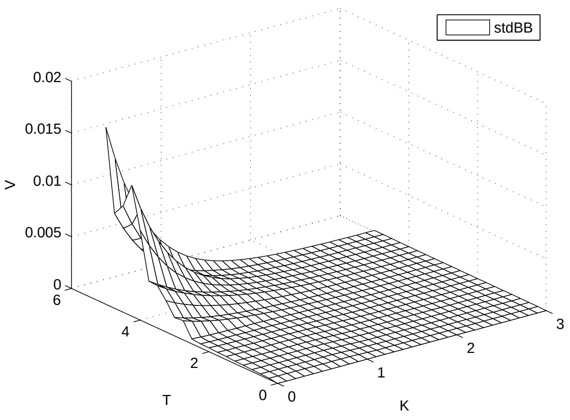
Figure 19: Put



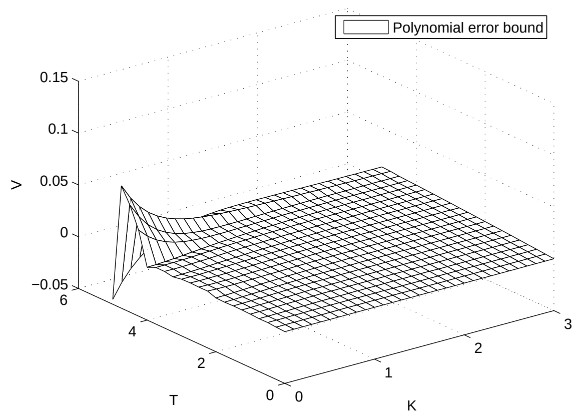
(a) Pricing surface



(b) Pricing error

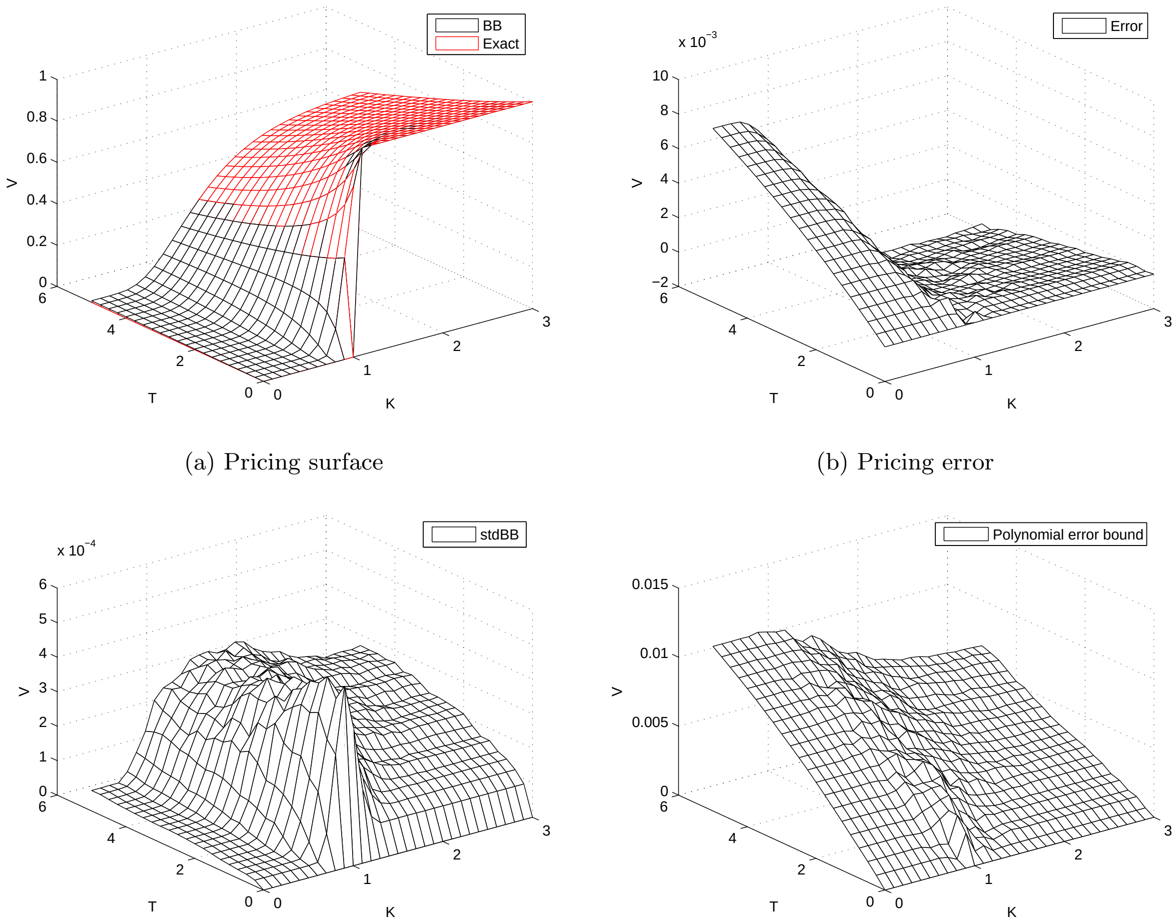


(c) Standard deviation



(d) Max polynomial error

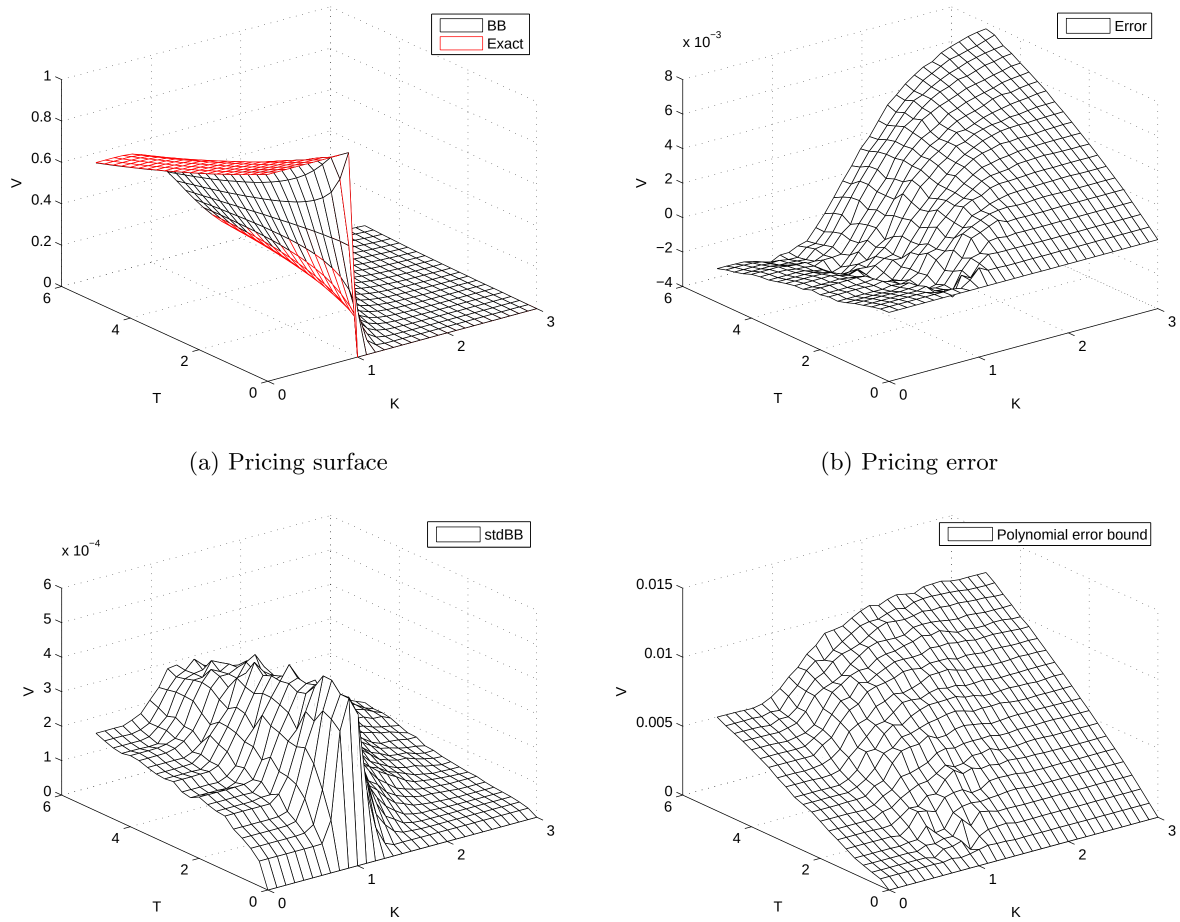
Figure 20: Call



(a) Pricing surface (b) Pricing error

(c) Standard deviation (d) Max polynomial error

Figure 21: Binary Put



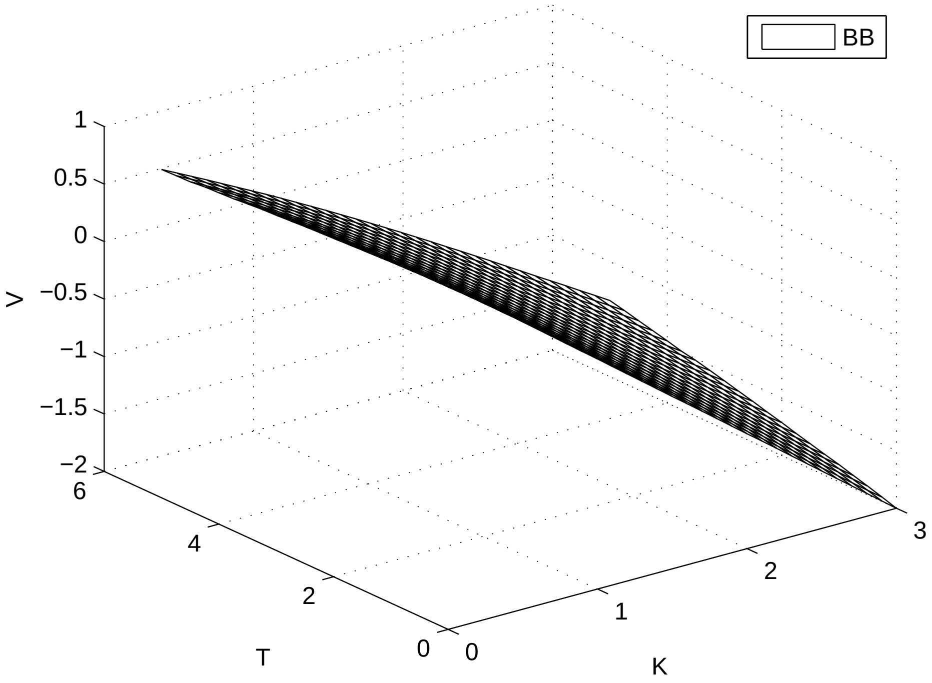
(a) Pricing surface

(b) Pricing error

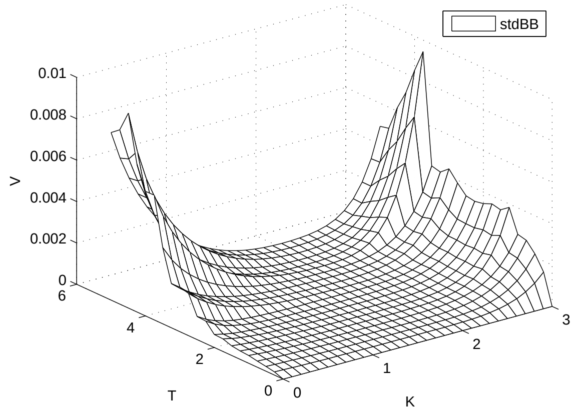
(c) Standard deviation

(d) Max polynomial error

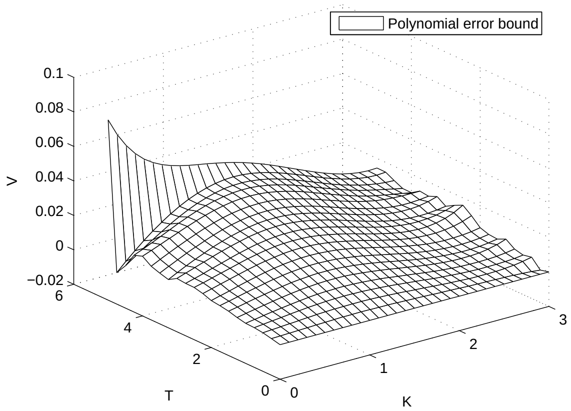
Figure 22: Binary Call



(a) Pricing surface



(b) Standard deviation



(c) Max polynomial error

Figure 23: Forward

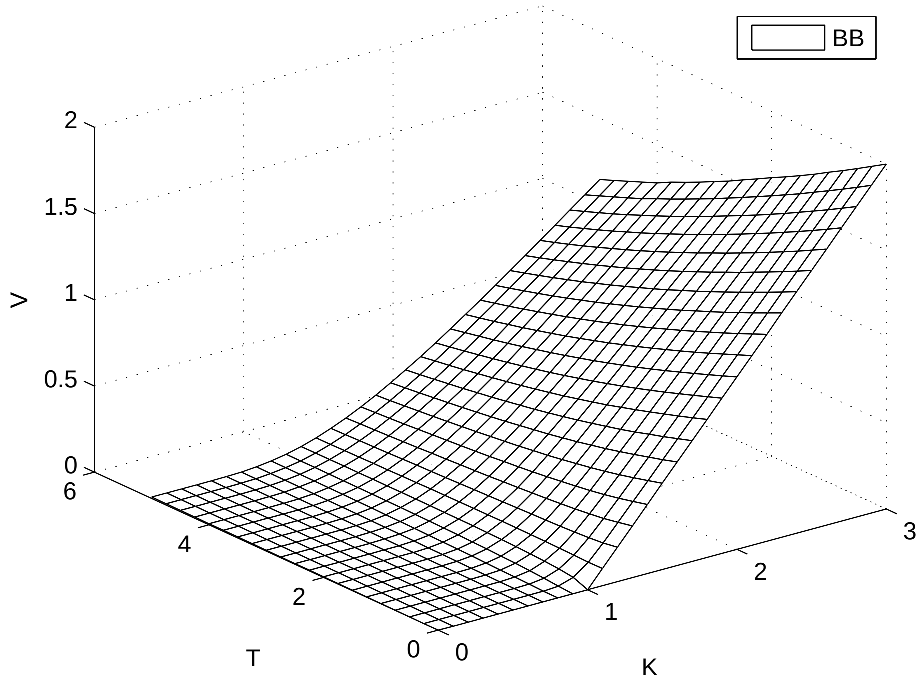
B.2 CIR Rates

B.2.1 Parameters

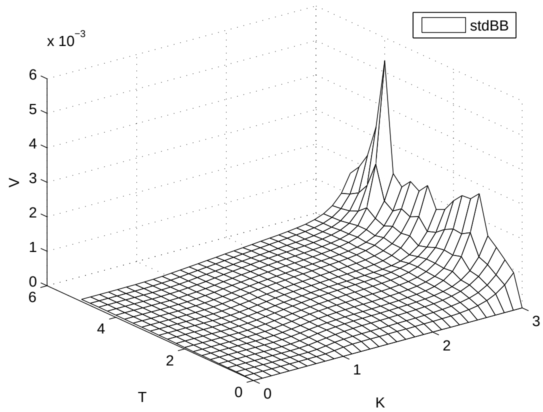
The CIR interest rate models has the following specific parameters:

$$
\begin{aligned}
r_0 &= 0.05 && \text{Interest rate at inception,} \\
\kappa &= 0.2 && \text{Mean reverting parameter,} \\
\theta &= 0.05 && \text{Long term mean,} \\
\sigma_r &= 0.05^{1.5} && \sigma \text{ used for the dynamics of the rate process.}
\end{aligned} \tag{173}
$$

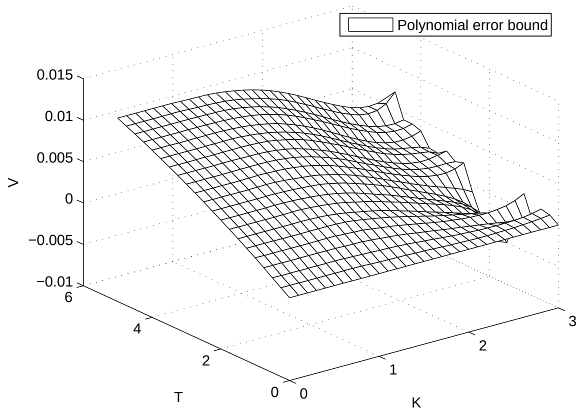
B.2.2 Type I



(a) Pricing surface

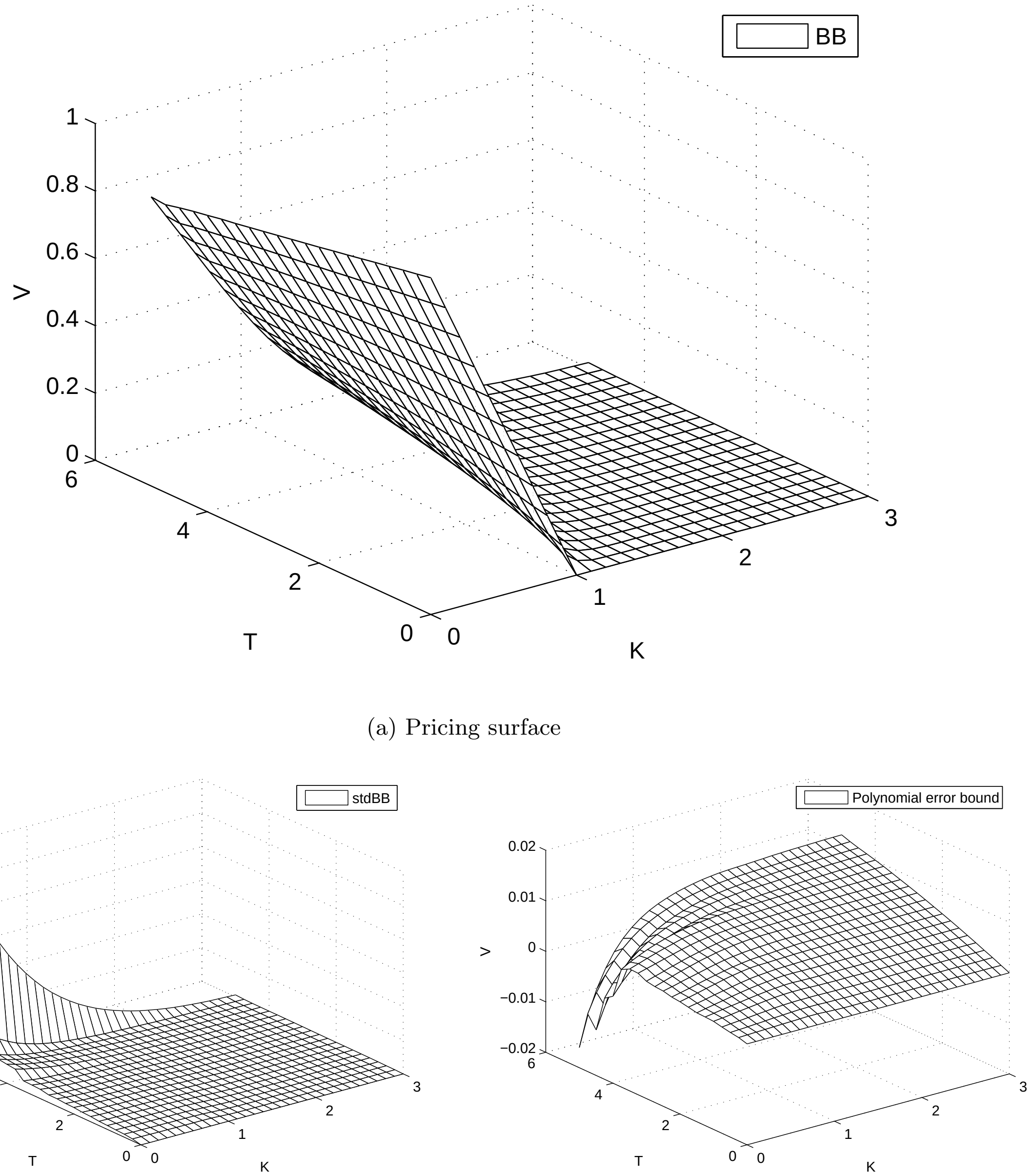


(b) Standard deviation



(c) Max polynomial error

Figure 24: Put

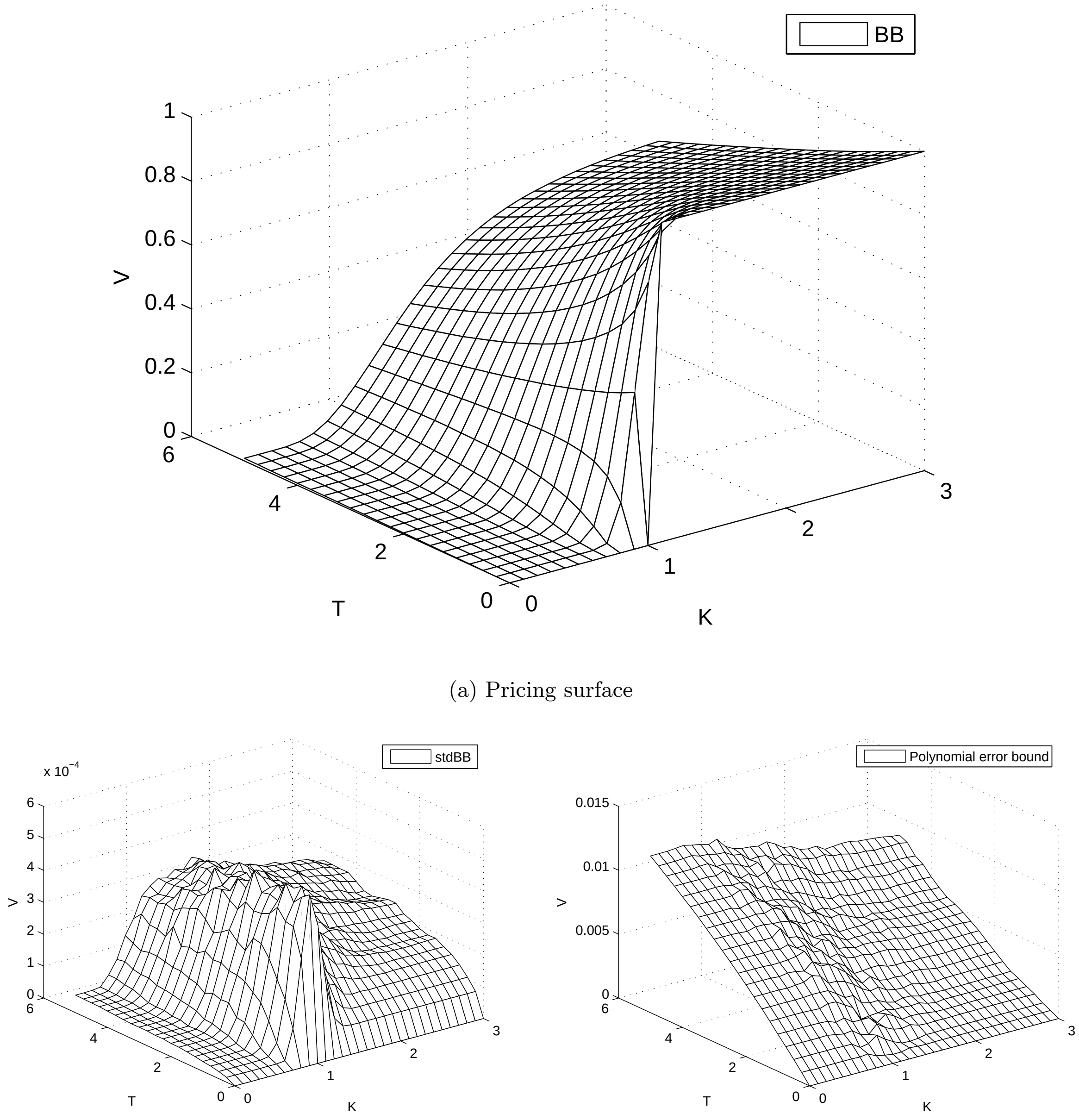


(a) Pricing surface

(b) Standard deviation

(c) Max polynomial error

Figure 25: Call

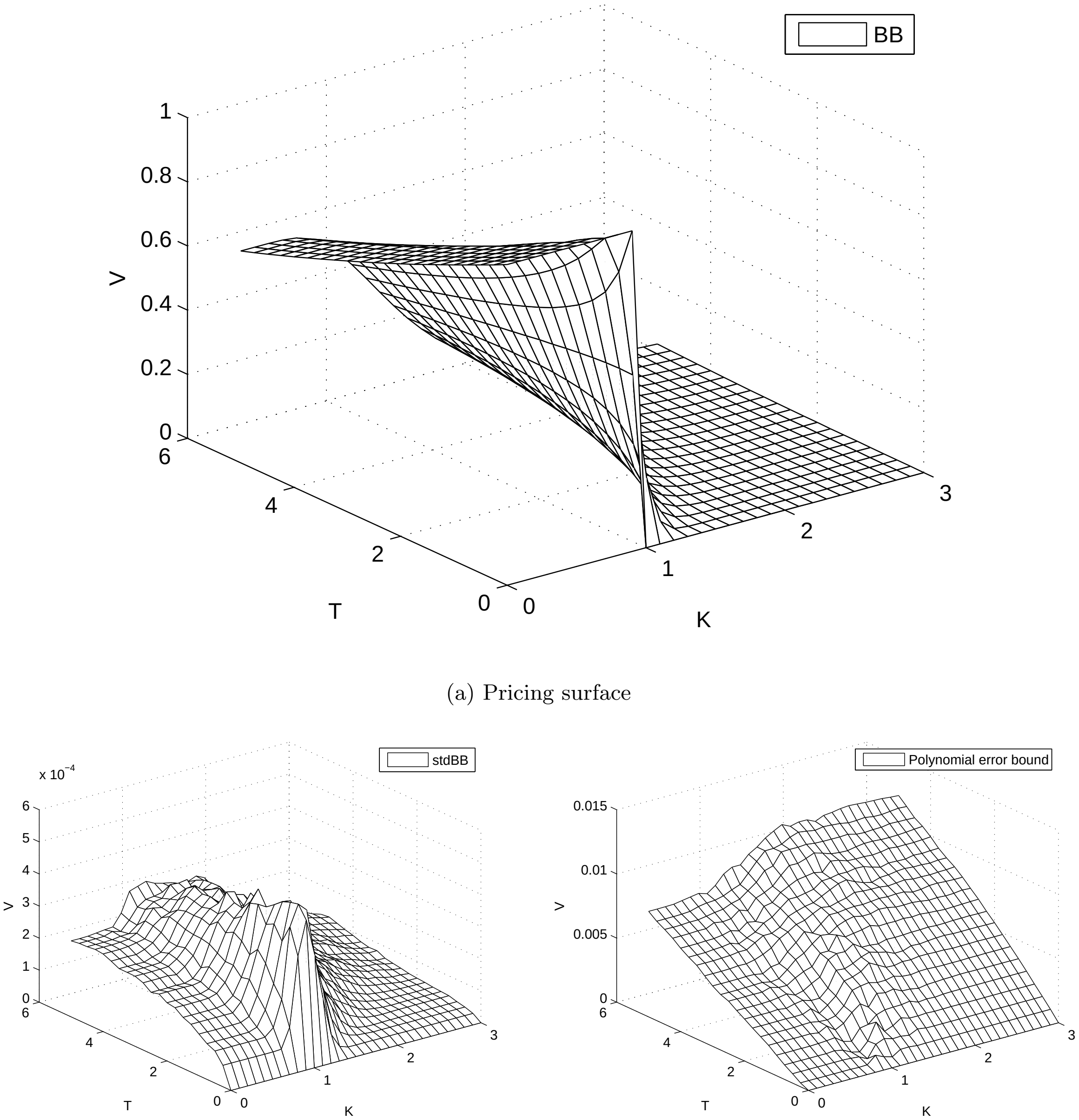


(a) Pricing surface

(b) Standard deviation

(c) Max polynomial error

Figure 26: Binary Put

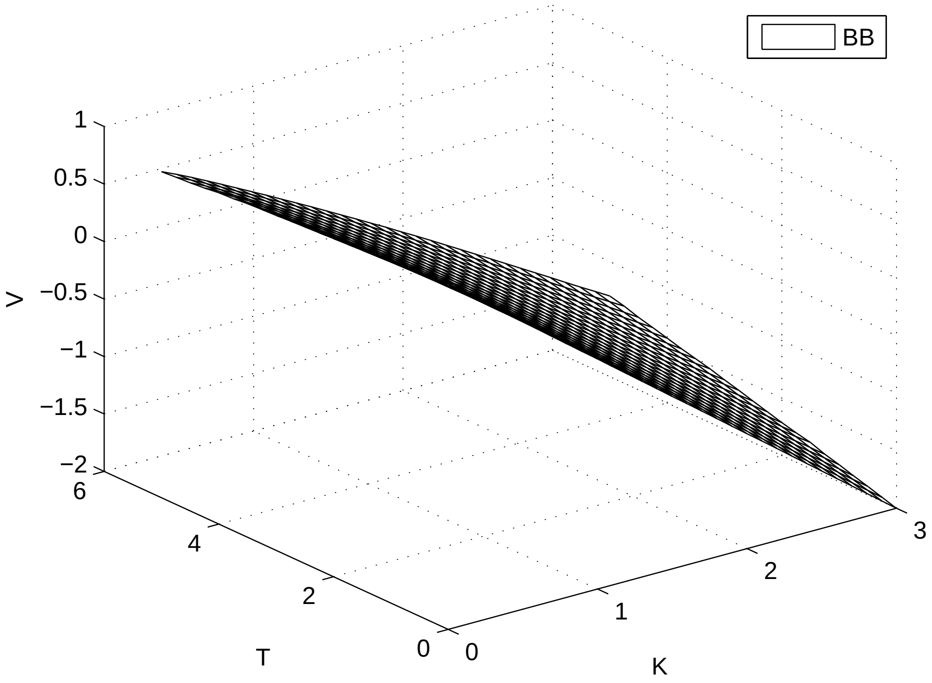


(a) Pricing surface

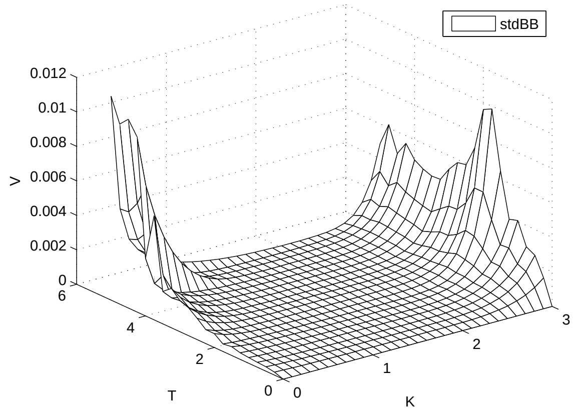
(b) Standard deviation

(c) Max polynomial error

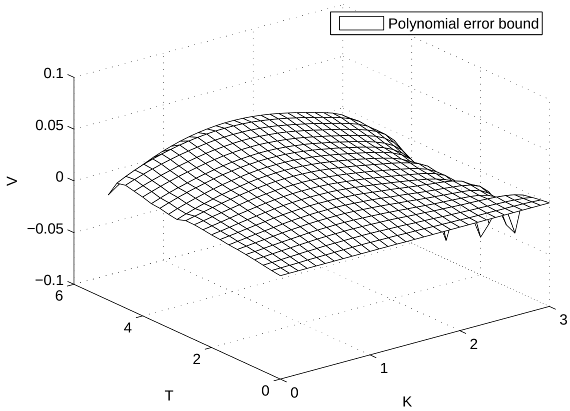
Figure 27: Binary Call



(a) Pricing surface



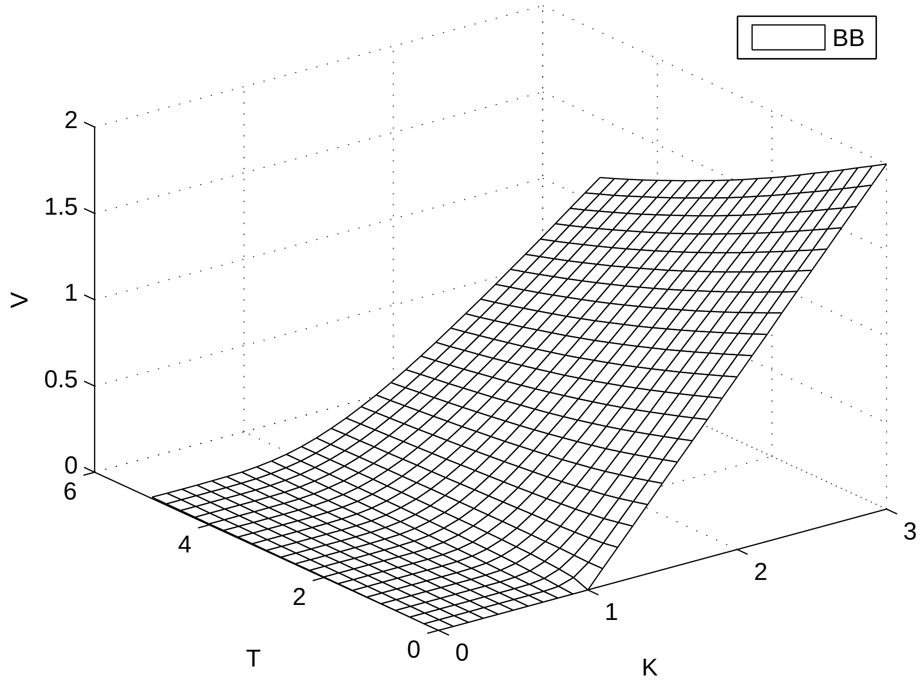
(b) Standard deviation



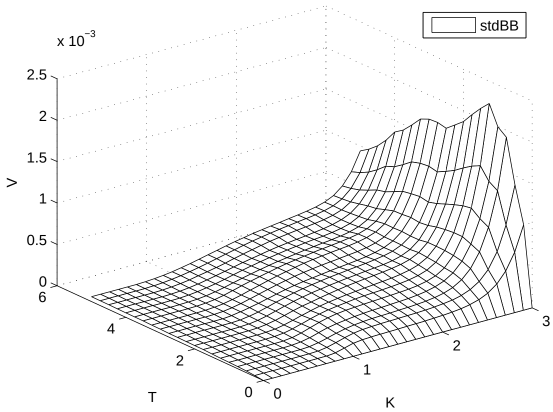
(c) Max polynomial error

Figure 28: Forward

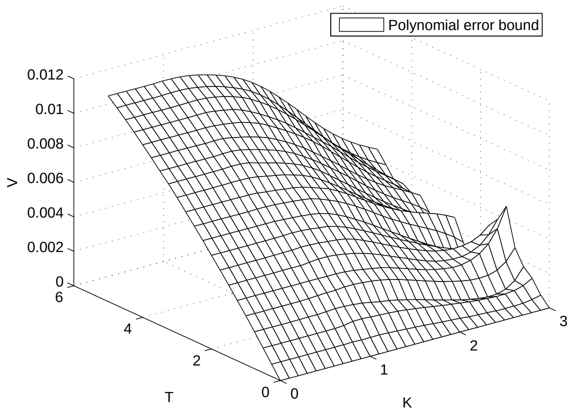
B.2.3 Type II



(a) Pricing surface

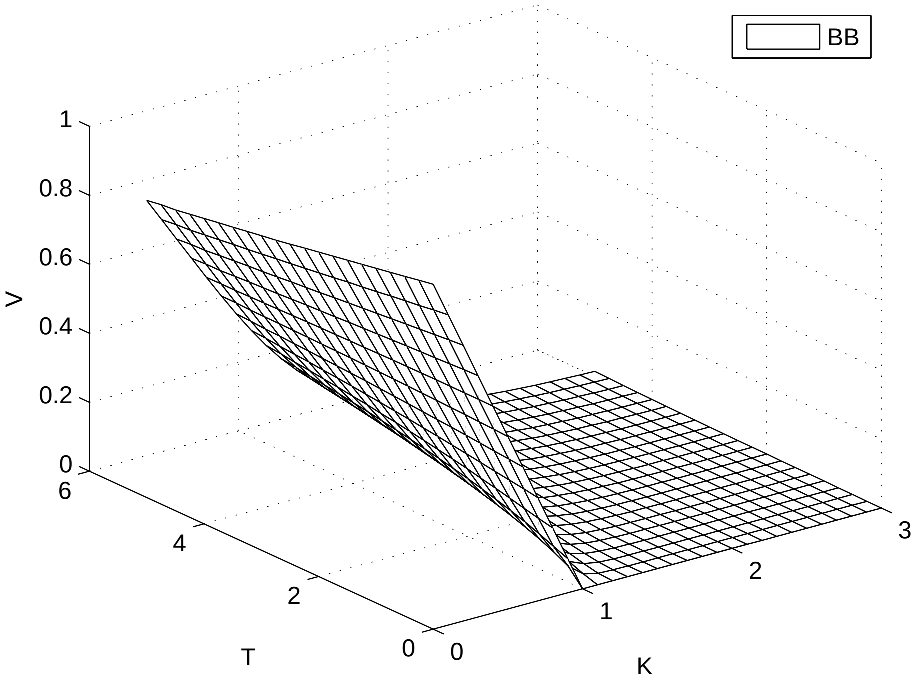


(b) Standard deviation

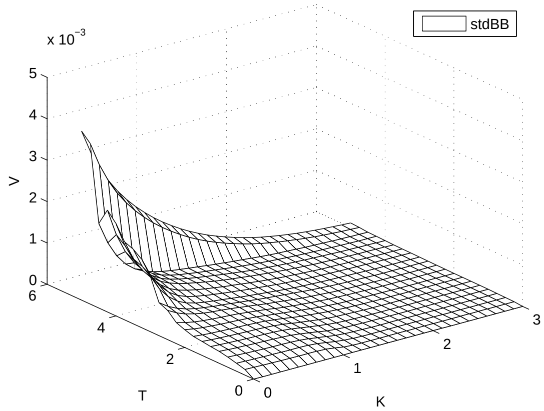


(c) Max polynomial error

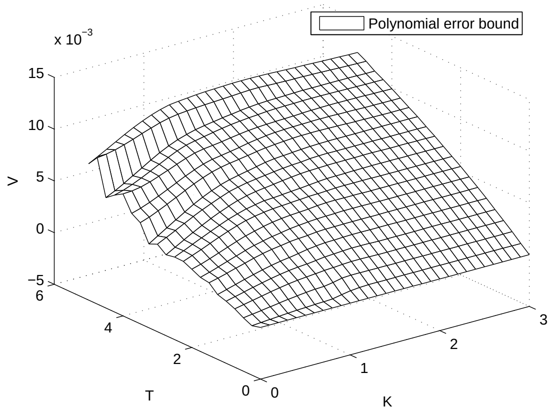
Figure 29: Put



(a) Pricing surface

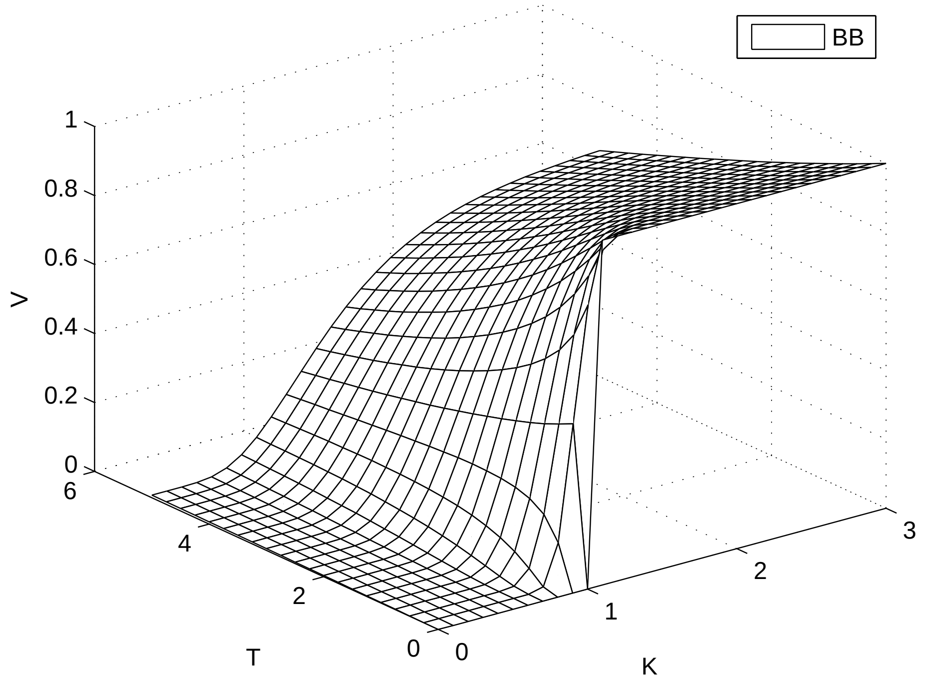


(b) Standard deviation

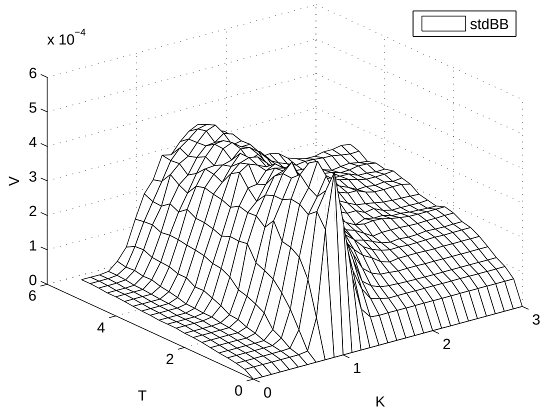


(c) Max polynomial error

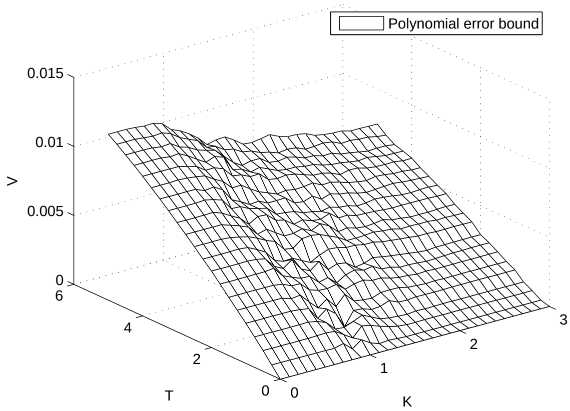
Figure 30: Call



(a) Pricing surface

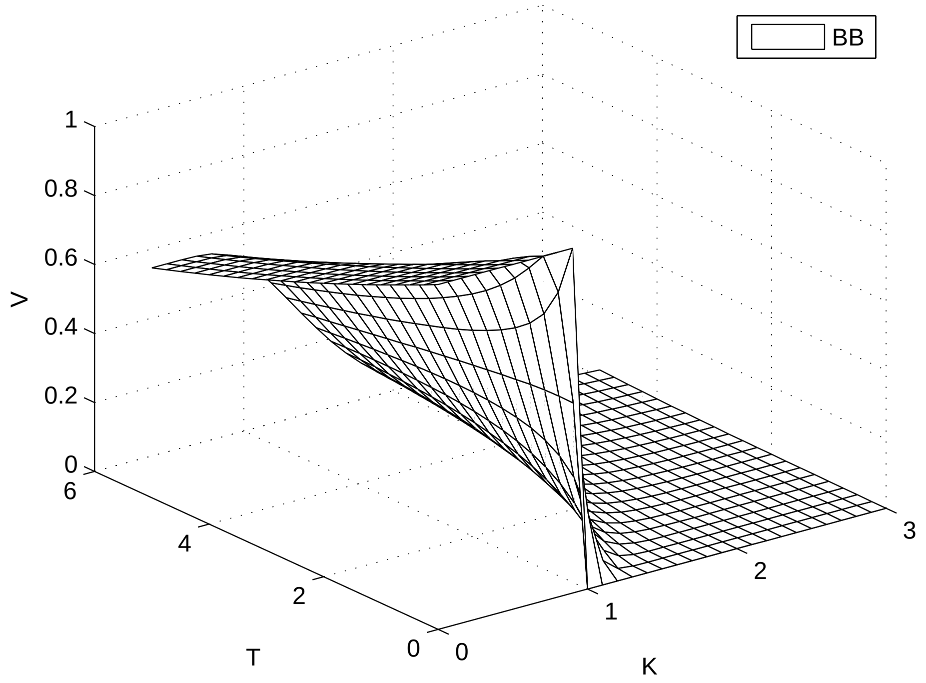


(b) Standard deviation

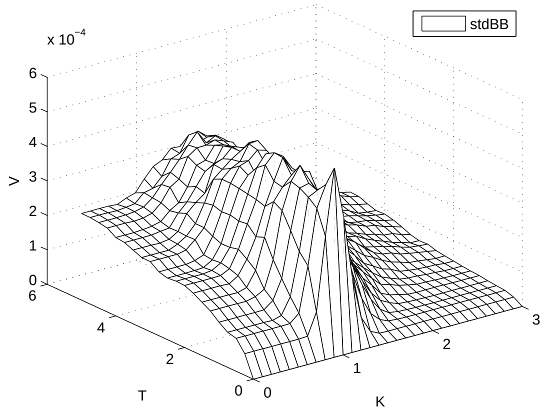


(c) Max polynomial error

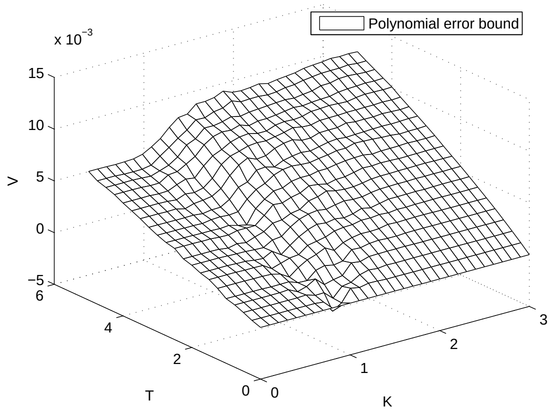
Figure 31: Binary Put



(a) Pricing surface

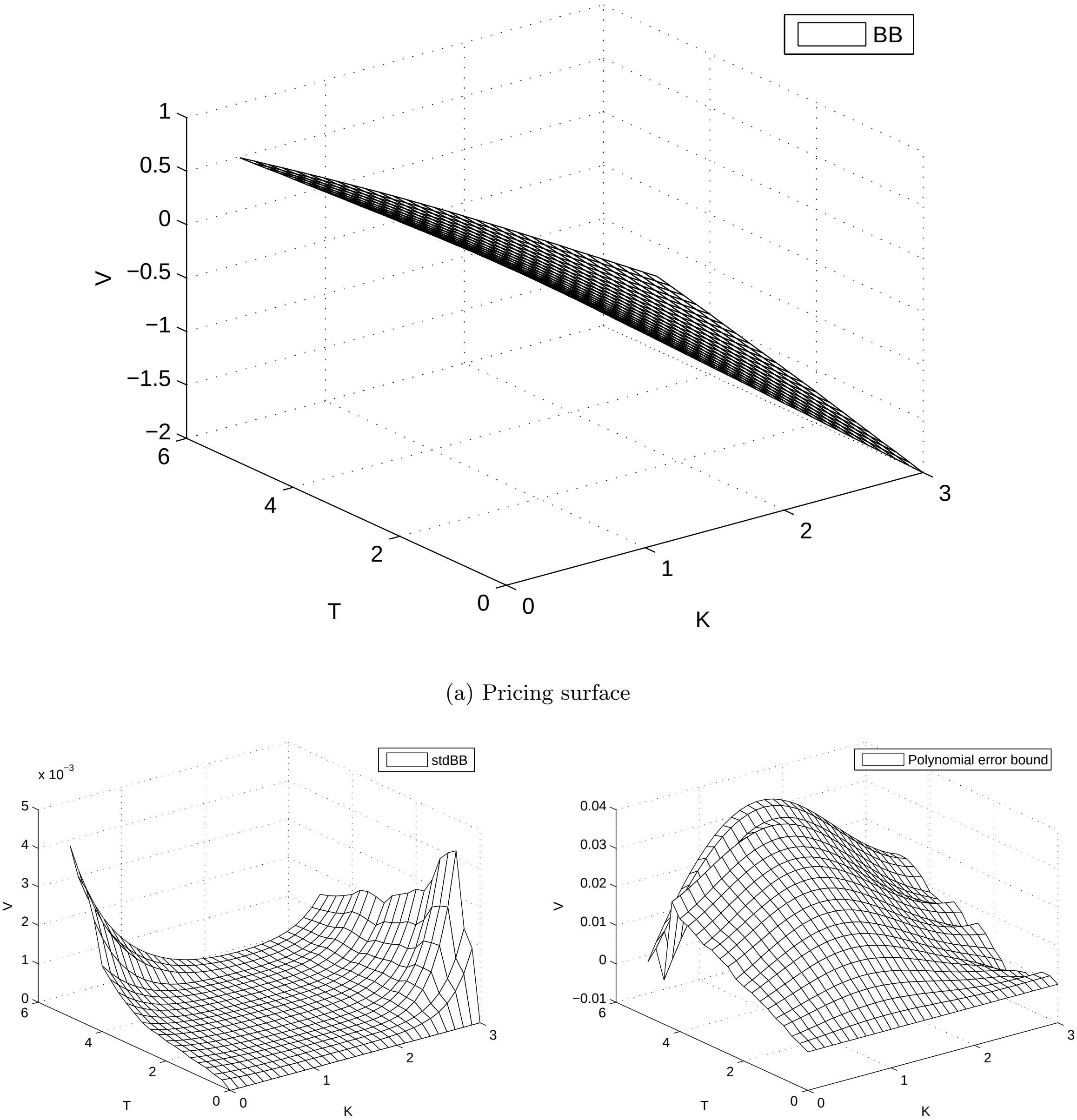


(b) Standard deviation



(c) Max polynomial error

Figure 32: Binary Call



(a) Pricing surface

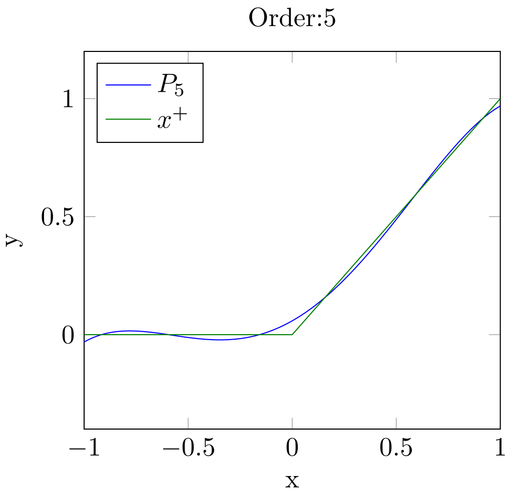
(b) Standard deviation

(c) Max polynomial error

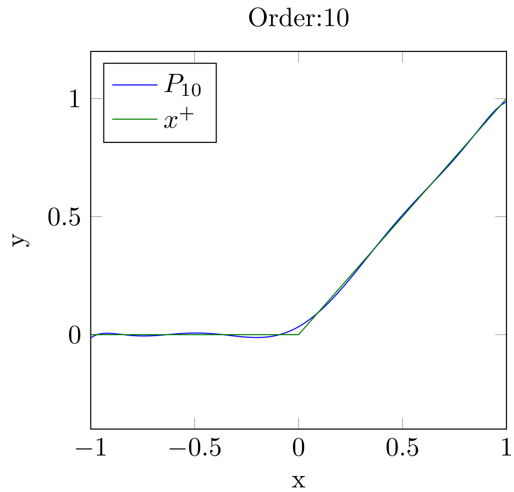
Figure 33: Forward

C Animation frames

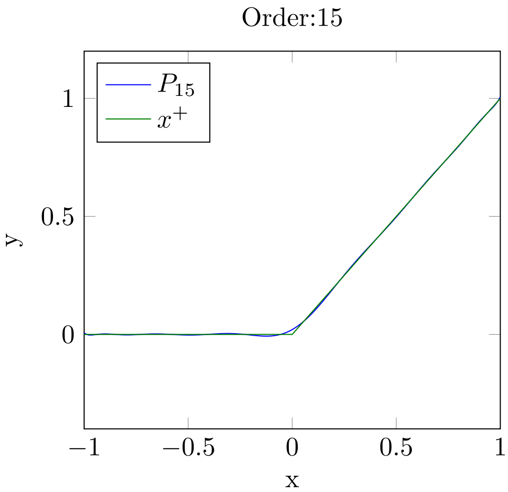
Snapshots of the animations in Sections 5 are given below:



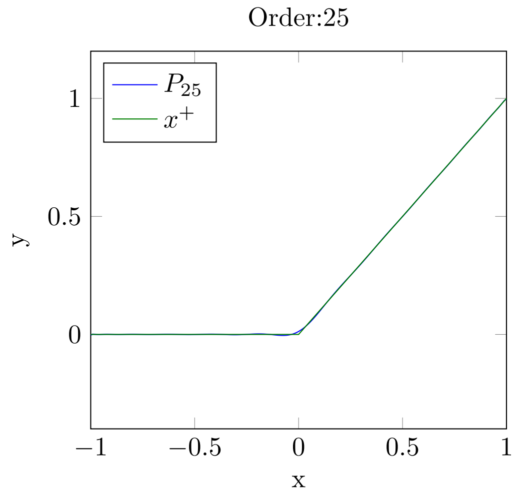
(a) P_5 approximation of x^+ on $[-1, 1]$



(b) P_{10} approximation of x^+ on $[-1, 1]$

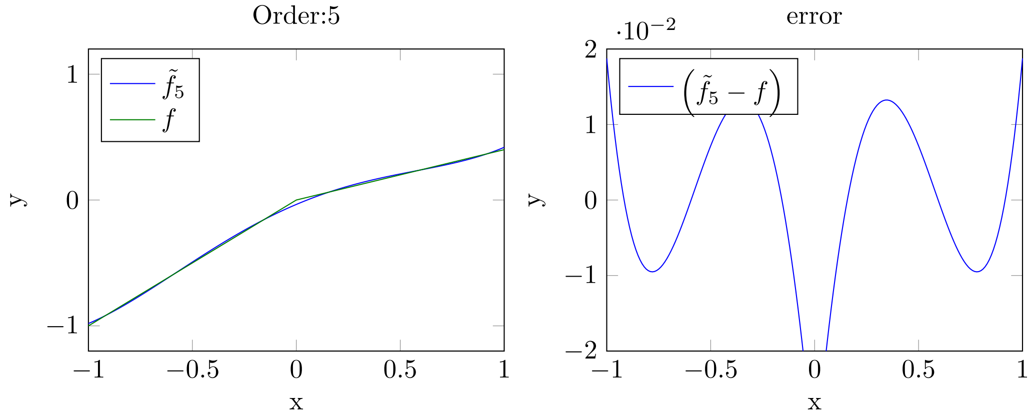


(c) P_{15} approximation of x^+ on $[-1, 1]$

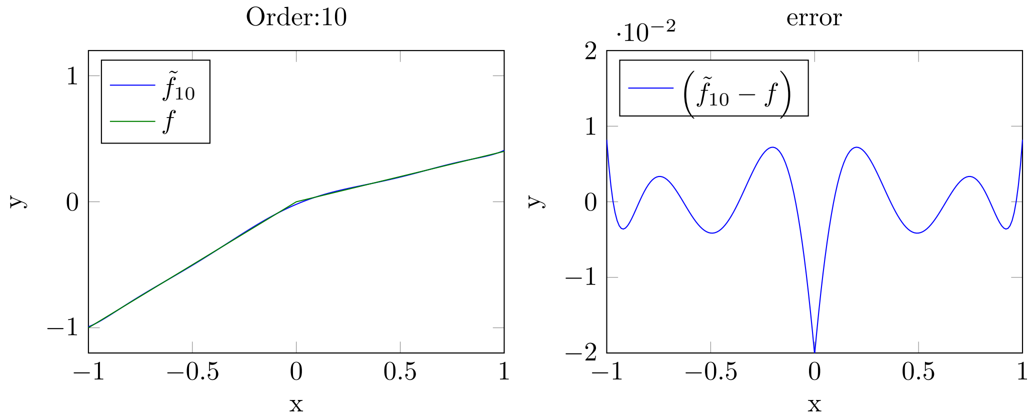


(d) P_{25} approximation of x^+ on $[-1, 1]$

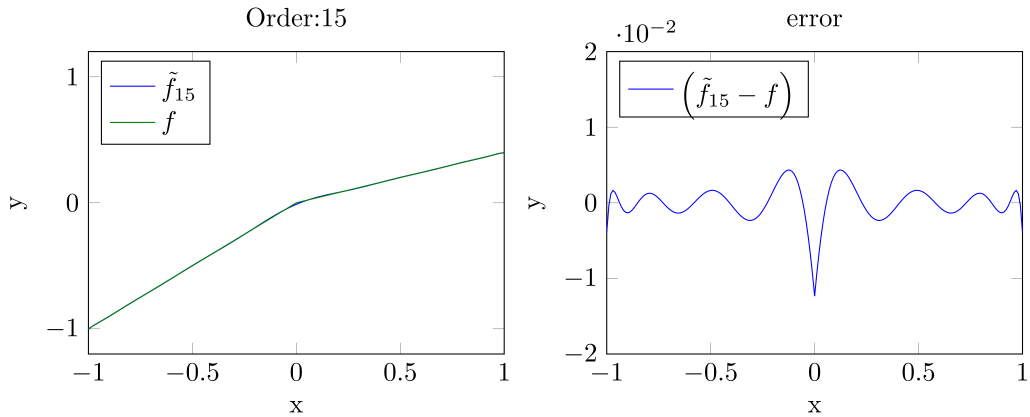
Figure 34: P_n approximation of x^+ on $[-1, 1]$



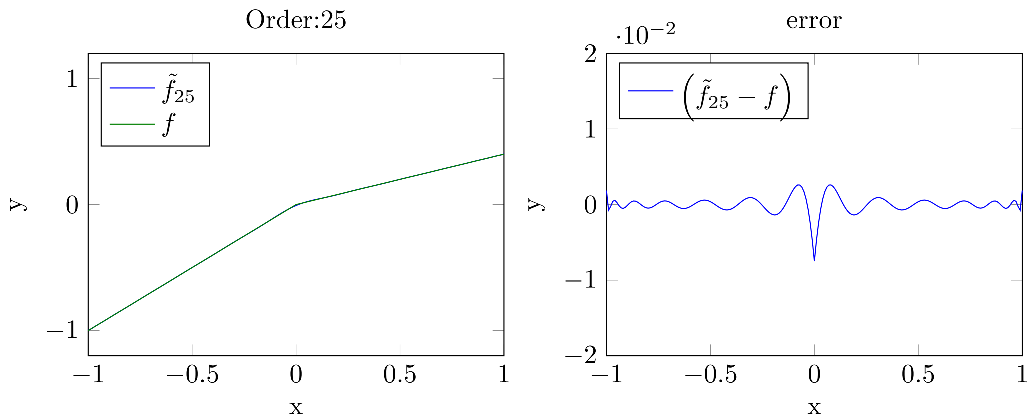
(a) $\tilde{f}_5$ approximation of x^+ on $[-1, 1]$



(b) $\tilde{f}_{10}$ approximation of x^+ on $[-1, 1]$

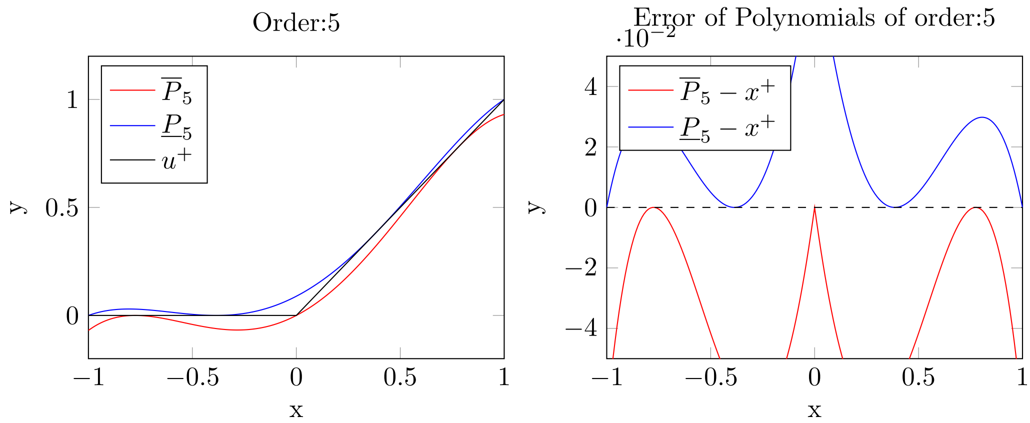


(c) $\tilde{f}_{15}$ approximation of x^+ on $[-1, 1]$

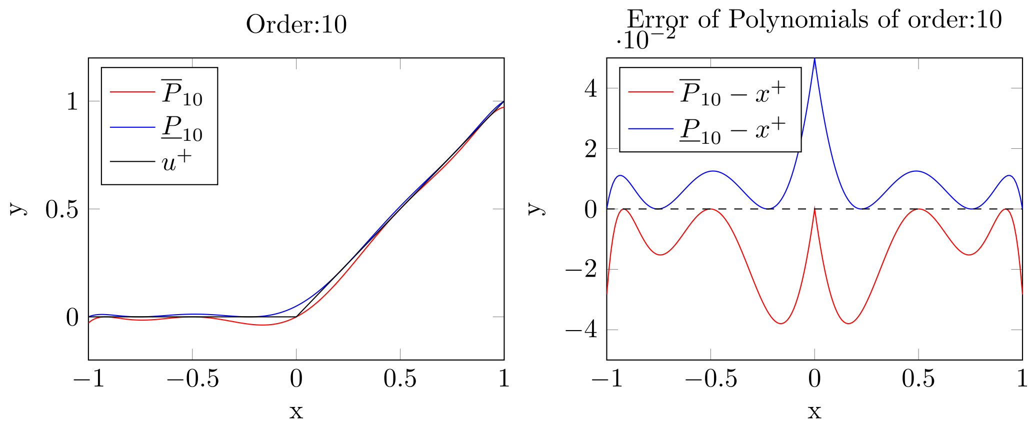


(d) $\tilde{f}_{25}$ approximation of x^+ on $[-1, 1]$

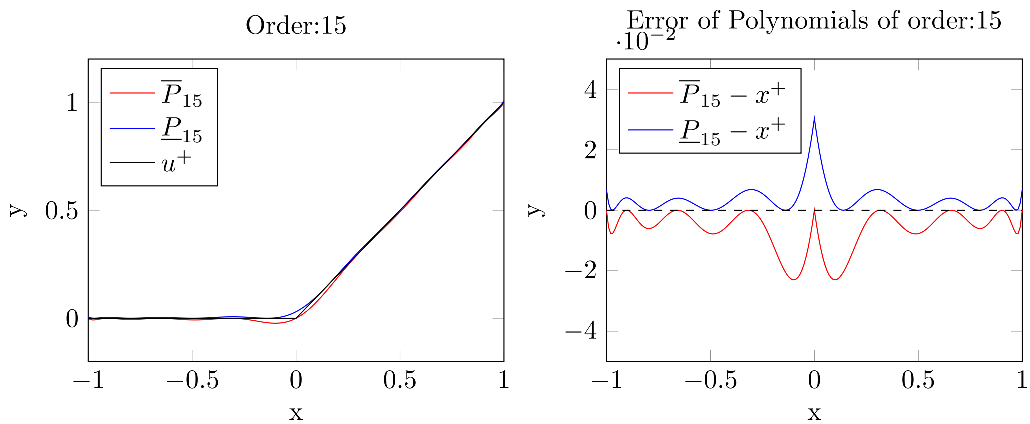
Figure 35: $\tilde{f}_n$ approximation of x^+ on $[-1, 1]$



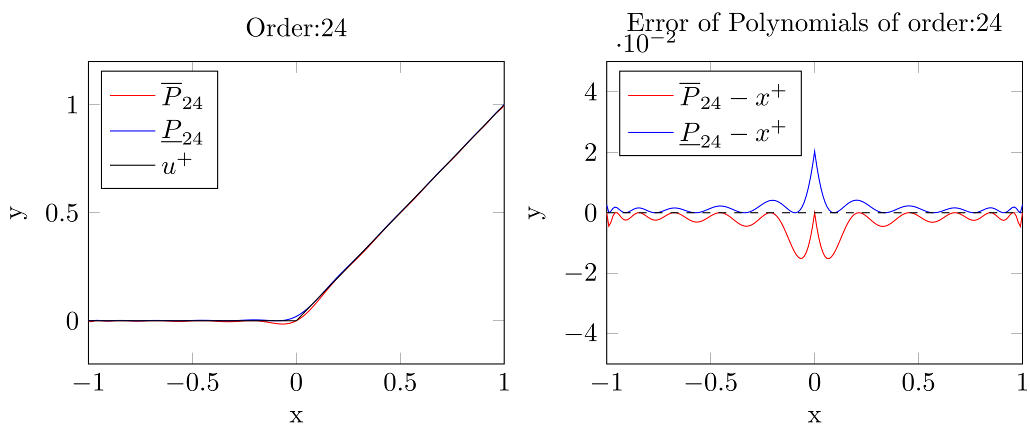
(a) $\underline{P}_5$ and $\overline{P}_5$ approximation of x^+ on $[-1, 1]$



(b) $\underline{P}_{10}$ and $\overline{P}_{10}$ approximation of x^+ on $[-1, 1]$



(c) $\underline{P}_{15}$ and $\overline{P}_5$ approximation of x^+ on $[-1, 1]$



(d) $\underline{P}_{24}$ and $\overline{P}_{24}$ approximation of x^+ on $[-1, 1]$

Figure 36: $\underline{P}_n$ and $\overline{P}_n$ approximation of x^+ on $[-1, 1]$

D P_n Coefficients

Coefficients of P_{15} used throughout the implementation.

c_n	P_n	$\underline{P}_n$	$\overline{P}_n$
c_0	0.0206	-0	0.0303
c_1	0.5000	0.5000	0.5000
c_2	2.4476	2.8876	2.2750
c_3	0	0	0
c_4	-15.5016	-19.7819	-13.8681
c_5	0	0	0
c_6	65.1065	84.7573	57.5044
c_7	0	0	0
c_8	-152.8010	-200.0238	-133.9469
c_9	0	0	0
c_{10}	198.0754	259.2135	172.6211
c_{11}	0	0	0
c_{12}	-132.5959	-172.9508	-114.9500
c_{13}	0	0	0
c_{14}	35.7550	46.3981	30.8409
c_{15}	0	0	0

Table 1: Coefficients of $P_n, \underline{P}_n$ and $\overline{P}_n$.

References

Damiano Brigo and Fabio Mercurio. *Interest Rate Models - Theory and Practice With Smile, Inflation and Credit.* Springer, 2006.

Damiano Brigo and Massimo Morini. Close-out convention tensions. *Risk*, December 2011.

Micheal G. Crandall, Hitoshi Ishii, and Pierre-Louis Lions. User's guide to viscosity solutions of second order partial differential equations. *American Mathematical Society*, 1992.

Ibrahim Ekren, Nizar Touzi, and Jianfeng Zhang. Viscosity solutions of fully nonlinear parabolic path dependent pdes: Part i. *arXiv:1210.0006v2*, 2013a.

Ibrahim Ekren, Nizar Touzi, and Jianfeng Zhang. Viscosity solutions of fully nonlinear parabolic path dependent pdes: Part ii. *arXiv:1210.0007v2*, 2013b.

Paul Glasserman. *Monte Carlo Methods in Financial Engineering.* Springer, 2003.

Jon Gregory. *Counterparty Credit Risk.* Wiley, 2010.

Julien Guyon and Pierre Henry-Labordére. *Nonlinear Option Pricing.* Chapman & Hall/CRC, 2014.

Pierre Henry-Labordére. *Analysis, geometry and Modeling in Finance, Advanced Methods in Option Pricing.* Chapman & Hall/CRC, 2009.

Pierre Henry-Labordére. Counterparty risk valuation: A marked branching diffusion approach. *arXiv:1203.2369v1*, 2012.

Pierre Henry-Labordére, Xiaolu Tan, and Nizar Touzi. A numerical algorithm for a class of bsdes via brancing process. *arXiv:1302.4624v3*, 2013.

John C. Hull. *Options, Futures and other Derivatives.* Pearson, 2012.

Andrew Koenig and Barbara E. Moo. *Accelerated C++ Practical Programming by Example.* Addison Wesley, 2000.

Stanley B. Lippman. *Essential C++.* Addison Wesley, 2002.

H. P. McKean. Application of brownian motion to the equation of kolmogorv-petrovskii-piskunov. *Communications on Pure and Applied Mathematics*, XXVIII, 1975.

Sanjay Mehrotra. On the implementation of a primal-dual interior point method. *SIAM Journal on optimization*, 1992.

Scott Meyers. *Effective C++ Third Edition 55 Specific Ways to Improve Your Programs and Designs.* Addison Wesley, 2005.

Sheldon M. Ross. *Stochastic Processes.* Wiley, 1996.

Rüdiger U. Seydel. *Tools for Computational Finance.* Springer, 2009.

Steven E. Shreve. *Stochastic Calculus for Finance II, Continuous - Time Models.* Springer, 2004.

J. Micheal Steele. *Stochastic Calculus and Financial Applications.* Springer, 2001.

Karl Weierstrass. On the analytic representability of so called arbitrary functions of a real variable. (uber die analytische darstellbarkeit sogenamter willk urlicher functionen einer reellen ver anderlichen). *Sitzungsberichte der Akademie zu Berlin*, 1885.