The Pricing of Double Barriers Knock-in Binary Put Option

Coursework for Derivatives 2

Ying Li (mr_liying@yahoo.com)
Nina Jhatakia (Nina.Jhatakia@gmail.com)
Alina Ma(alinama_919@yahoo.com)

MSc Mathematical Trading and Finance
Cass Business School
25th, March 2008

Abstract

As a coursework, we are required to price a double barriers knock-in binary put option. We used finite difference method in 24 ways and multinomial lattice in 12 ways. We also implemented analytic and Markov chain method. At the end, we compared these four methods and Monte Carlo method.

In this coursework, we discussed the speed, convergence rate and monotonicity of convergence for these methods. We also discussed whether extrapolation improves convergence.

Our Task

The subject is pricing of barrier option in the Black Scholes model. The option is a discretely (daily) monitored European style barrier knock-in option. The initial stock price is S0 = 100: Time to maturity is 40 trading days. The barriers are 105, 95, the binary strike price is 105.

Assume the logarithm of the stock price is a Gaussian process with constant drift and volatility, b = 0.02; c = 0.4 under the risk-neutral measure: Assume time is measured in years and one year has 250 (trading) days. There are no dividends.

The Option Features

1. Sensitive to barriers and strike price

For the option is a knock-in double barriers option, the barriers have big impact to the price. As a binary option, its payoff is not continues, so the strike price is a critical value also.

2. Barriers and strike price are near to the initial price.

For the barriers (105 and 95) and strike (105) are very near to the initial price (100), especially with the high volatility(40%), the possibility of hitting barriers are very high.

3. Discontinuous Payoff

A Binary Option has a discontinuous payoff. That means a continuous underlying price could generate a completely discontinuous option value. That is, in the payoff chart, there is a jump at the strike price. Discontinuous payoffs generate special oscillation problems.

Analytic Solution

Hui(1996) published closed-form formulas for the valuation of one-touch double-barrier binary options. A knock-in one-touch double-barrier pays off a cash amount K at maturity if the asset price touches the lower L or upper U barrier before expiration. The option pays off zero if the barriers are not hit during the lifetime of the option. Similarly, a knock-out pays out a predefined cash amount K at maturity if the lower or upper barriers are not hit during the lifetime of the option. If the asset price touches any of the barriers, the option vanishes. The formula for the knock-out variant is:

$$c = \sum_{i=1}^{\infty} \frac{2\pi i K}{Z^2} \left[\frac{\left(\frac{S}{L}\right)^{\alpha} - (-1)^i \left(\frac{S}{U}\right)^{\alpha}}{\alpha^2 + \left(\frac{i\pi}{Z}\right)^2} \right] \times \sin\left(\frac{i\pi}{Z} Ln\left(\frac{S}{L}\right)\right) e^{\frac{-1}{2}} \left[\left(\frac{i\pi}{Z}\right)^2 - \beta\right] \sigma^2 T$$

Where

$$Z = Ln\left({}^{U}/{}_{L}\right), \quad \alpha = \frac{-1}{2}\left(\frac{2b}{\sigma^2} - 1\right) \quad \beta = \frac{-1}{4}\left(\frac{2b}{\sigma^2} - 1\right)^2 - 2\,\frac{r}{\sigma^2}$$

The option we are pricing is just a knock-in one-touch double-barrier with K=1. The following table gives some pricing results related to our option. We use this as a benchmark to judge our pricing results of other methods. The value of a continuous monitoring corresponding option is 0.6024.

Double-Barrier Binary Option Value

(S=100, T= $\frac{40}{250}$, r=0.1, b=0.02, K=1, L=95, U=105)

	$\sigma = 0.1$	$\sigma = 0.2$	$\sigma = 0.3$	$\sigma = 0.4$
Binary put (K=105)	0.7868	0.6629	0.6205	0.6024
Knock Out	0.5352	0.0527	0.0010	0.0000
Knock In	0.2516	0.6102	0.6195	**0.6024**

Finite Difference solution

At fist, in order to compare each method thoroughly, we will not only price the discrete monitored option, which is our task, but also we will price the continuous monitored option.

In order to compare the different convergence speed and pricing accuracy, we will use three kinds of finite difference methods: standard Implicit Finite Difference Method (IFD), Standard Crank-Nicosoln Finite Difference Method (CN) and the Improved Crank-Nicosoln Difference Method (Improved). IFD is stable, so it is a good benchmark for us. CN converges quickly. While CN has some stability issues, we can use some techniques to improve it in the Improved. So, we choose these three.

With the three basic methods, we will use different finite grid construction methods. One method puts all the critical values on the nodes as some papers suggested. The other method puts those just between the nodes as other papers suggested.

Besides the direct pricing method, we will use indirect pricing method also. It will give us totally different features.

So, we have 3 (CN,IFD and Improved) x 2 (OnNodes and BetweenNodes) x 2 (Directly and Indirectly) x 2 (Continuous and Discrete) = 24 pricing ways.

Implementation

1. Continuous and discrete monitor frequency

In order to compare the difference between the continuous and discrete monitoring frequencies, we calculate the price for both situations. For the former, we use finer and finer grids in both the space and time directions. For the latter, we use finer and finer grids in the space direction only, leaving the time with daily divisions.

2. Direct and indirect pricing

In theory, the price of a double knock-in binary put is that of a normal binary minus that of a double knock out binary put. In short,

DKI_Binary_Put= Binary_Put – DKO_Binary_Put

So, instead of pricing the DKI_Binar_Put directly, we can price DKO_Binary_Put and then subtract it from a Binary_Put. In regards to our task, the standard binary put is a European option, which can be priced by a continuous BS model and the barriers have discrete monitored frequencies

3. Grid Construction Way

In order to capture the critical events, one way is to put the critical values on the grid nodes. Although there is a way to shift the grid to put the strike on the grid at the cost of the initial stock price not being on the grid any more, the grid shift method is not suitable to barriers option because we have to put both barriers on the grid nodes.

Still, based on an equally spaced grid, we offer an innovative and general method to meet the requirements. We set the barriers as boundaries firstly and then divide the distance equally. So, both the barriers are definitely on the grid nodes, at the cost that the initial stock price may not be on the grid nodes, nor at the center of the grid.

At the same time, some papers suggest we should simply put the critical values just between the nodes, so we also test this.

4. Improved Crank-Nicolson Method

The Crank-Nicolson scheme has faster convergence (quadratic in time and space, compared to linear in time, quadratic in space for EFD and IFD) while maintaining the stability of the implicit method. But, CN time-stepping can have problems if the time-step is larger than the explicit time-step size since C-N is not a positive coefficient method. Normally, this problem is not severe. However, the following situations can cause difficulties: digital payoffs, barriers. In these cases, we can observe slow convergence (not at the second order rate) and obvious oscillations. Regarding our option, we have to improve CN.

a. Rannacher time-stepping

Rannacher (1984) suggests a payoff smoothed (if required) and after each rough initial state, we take fully implicit finite time-steps (two in implementation), and use C-N thereafter. Note that these methods are not guaranteed to preclude oscillations, but we are guaranteed to get second order convergence. Second order convergence does not imply no oscillations. In practice, these methods work remarkably well.

The rationale is that high frequency error components will be dampened by the implicit steps, leading to smooth convergence. The expected rate of convergence remains quadratic since only a finite number of implicit steps are taken. Furthermore, this type of time-stepping can help eliminate oscillations in the solution derivative values. Effective hedging of the underlying contract is then made easier.

b. Discontinuous Payoff

Discretely monitored barriers introduce discontinuities at observation dates, while the payoff itself is discontinuous for digital options.

D. M. Pooley (2002) suggests that discontinuities in the payoff function (or its derivatives) can cause inaccuracies for numerical schemes when pricing financial contracts. D. M. Pooley (2002) discussed three techniques: averaging the initial data, shifting the grid, and a projection method and thinks these techniques are not sufficient to restore expected behaviour. D. M. Pooley (2002) concludes that when combined with a special time-stepping method, high accuracy is achieved.

As a discrete monitored binary option, the option should suffer all the possible problems. But as the strike price is just the same as the barrier (the boundary), so according to the constructed grid, there is no evidence of any discontinuous payoff. Therefore the discontinuous payoff is not applicable here.

However, generally, discrete payoffs remain a significant problem for option pricing.

Pricing Results

When discretizing in time, many methods are available. Explicit schemes are typically simple to implement, but suffer from stability issues. Implicit methods are unconditionally stable, but only exhibit linear convergence. If possible, it is advantageous to use Crank-Nicolson timestepping to achieve quadratic convergence. However, Crank-Nicolson timestepping is prone to spurious oscillations if twice the maximum stable explicit timestep is exceeded. Further, if discontinuous initial conditions are present, the expected quadratic convergence may not be realized.

1. **Indirect method gives us an amazing good result, which convergent quickly.**

Direct and indirect pricing gives us different values, which converge to each other generally. Regarding the option, indirect pricing quickly gives us a near-accurate stable answer, due to the very high knock-in probability. Due to the high probability, the double knock-in barriers option is nearly a standard binary option.

In this way, if we use the indirect method, it means we use a standard binary option as a benchmark to calculate. At the same time, as we use direct method, this means we use zero as the benchmark. It is obvious that the option value is near to that of a standard binary option, so the convergence is very fast.

2. **Generally, OnNodes method gives us a better result**

The reason is obvious that all the critical events are very sensitive to the timing. When we put the critical value on the nodes, it is easy to capture the event in time.

A special reason is that all the critical values are on the boundaries, which means we do not need any trade-offs to put this critical value on node or that value on node.

3. **Improved CN method does give us a smooth, but worse result in discrete monitoring**

In continuous monitoring it seems Improved CN method is the same as the CN - there are no oscillations with either method. In discrete monitoring, Improved CN method does give us a better result. There is no oscillation except for oscillation in the between-nodes method of Improved CN.

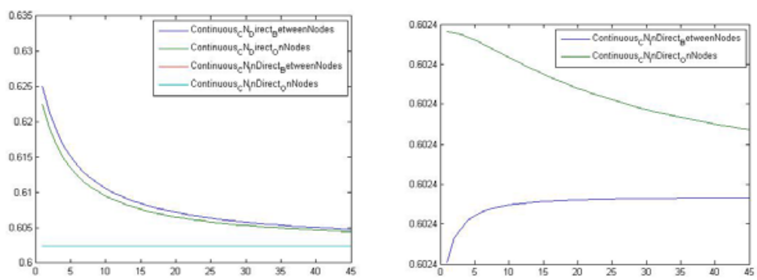
4. **CN method gives a better convergence speed and improved CN method has a monotonic convergence.**

Compared to other methods, CN and Improved CN have faster convergence speed. At the same time, IFD is always stable and has no oscillation. Improved CN also has no oscillation.

5. **Extrapolation improves convergence for IFD and improved CN, but which does not make any real sense.**

It is obvious that IFD and improved CN have montonic convergence, so it is possible to use extrapolation to speed the convergence. At the same time, it is obvious also that even with traditional extrapolation, the results couldn’t be as good as CN method.

Figure 1: Continuous Monitored Option Pricing



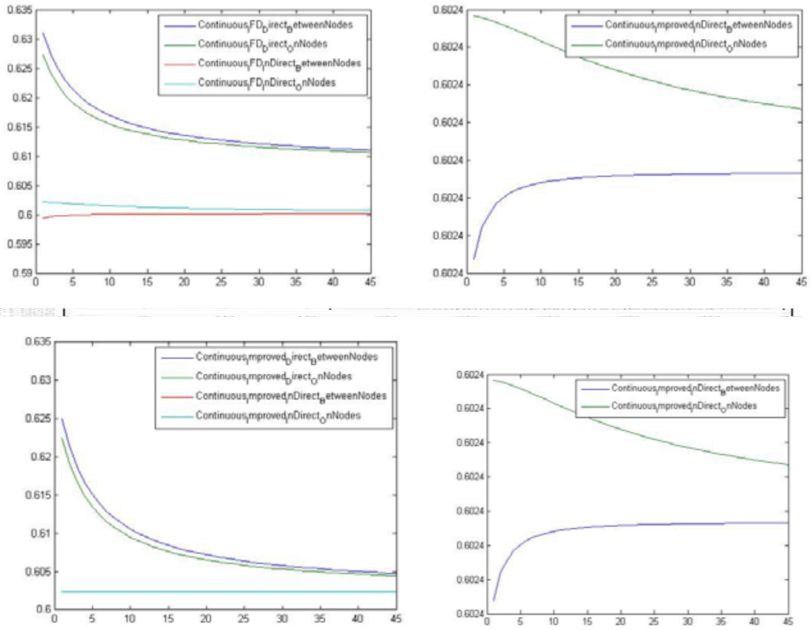
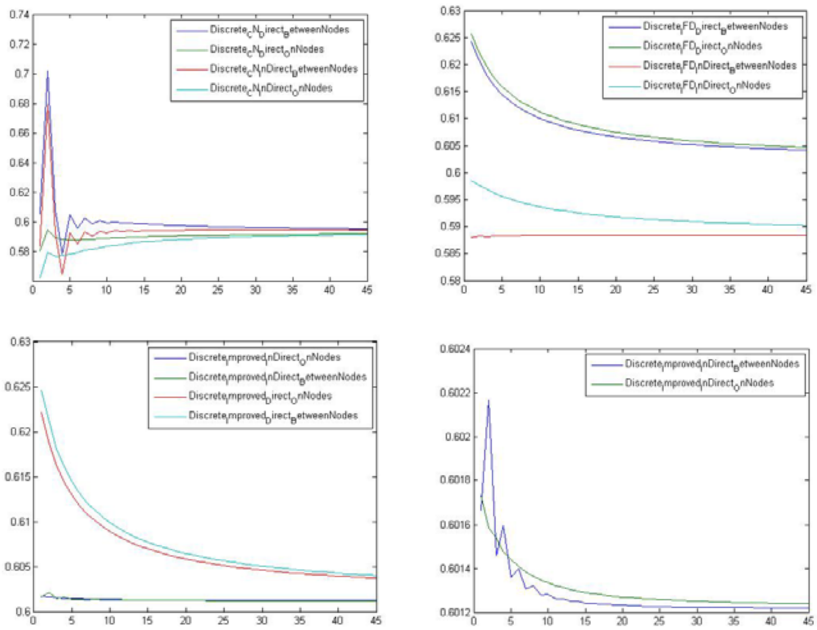


Figure 2: Discrete Monitored Option Pricing



Multinominal Solution

Alford, J. and N.Webber (2001) investigate multinonimal method and that the heptanomial lattice is the fastest and most accurate of the lattices of higher order, and recommend its use as standard in many one factor lattice implementations. In order to ensure that convergence achieves its theoretical rates, Alford, J. and N.Webber (2001) suggests smoothing to ensure that the payoff function is sufficiently differentiable, and truncation to increase the convergence rate as a function of computational effort. George M. Jabbour et al. (2001?) gives the needed parameters to implement multinominal.

We didn't follow all the suggestions of Alford, J. and N.Webber (2001) to smooth the payoff. The reason is that we are pricing a daily monitoring binary option. The monitoring is the most important factor and then barriers (strike). Due to the monitoring and due to the narrow space between two barriers, the convergence is limited.

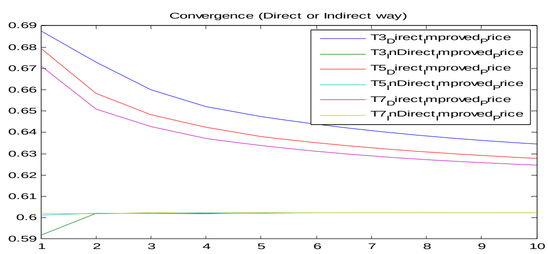
We implemented standard multinominal with 3, 5 and 7 branches. They are recombined. In order to improve the accuracy and convergence, when we price knock-in option, we calculate the nodes on and just out the barriers by BS formula.

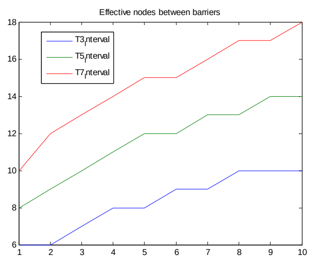
Due to the narrow space between the barriers, in fact, only part of the nodes are involved into the calculation. Most of nodes out of the barriers are useless. So, we also implemented improved code to use a reduced tree to calculate.

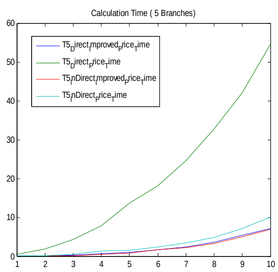
At the same time, we implement the code to calculate knock-in directly and that to calculate knock-out option first and then use knock-in and knock-out parity to get knock-in value.

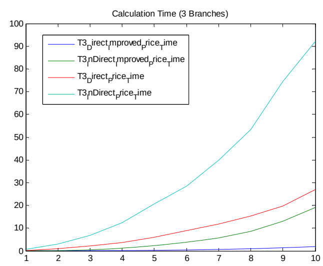
So, we have 3 (3, 5 and 7 branches) x 2 (Standard and improved) x 2 (Directly and Indirectly) = 12 pricing ways.

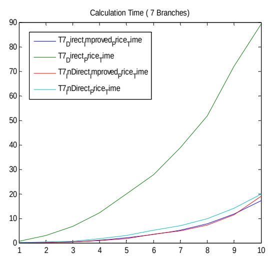
Pricing Results











1. The effective nodes between barriers are very limited

At most, only 18 nodes are involved for one time step even for 7 branches tree with 10*40=400 time steps. The number of nodes is shown in the following table.

Time Steps(*40)	1	2	3	4	5	6	7	8	9	10
Tree with 3 branches	6	6	7	8	8	9	9	10	10	10
Tree with 5 branches	8	9	10	11	12	12	13	13	14	14
Tree with 7 branches	10	12	13	14	15	15	16	17	17	18

2. The accuracy is bad and the convergence is slow, comparing Finite Difference solution.

Partly this is because the born low convergence of multinominal method, but the most important factor is the effective nodes are involved into calculation is too limited. This is resulted from the feature of the option.

3. **The reduce tree technique improved speed a lot.**

It is because only a few nodes are involved into calculation, so the reduced tree gives us times speed, comparing with a standard multinominal method.

4. **Extrapolation improves convergence for multinominal method, but which does not make any real sense.**

It is obvious that multinominal method have montonic convergence, so it is possible to use extrapolation to speed the convergence. At the same time, it is obvious also that even with traditional extraplolation, the results couldn't be as good as other methods.

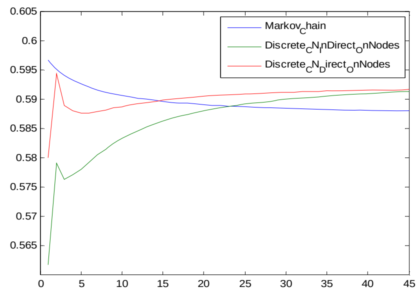
The Markov Chain Solution

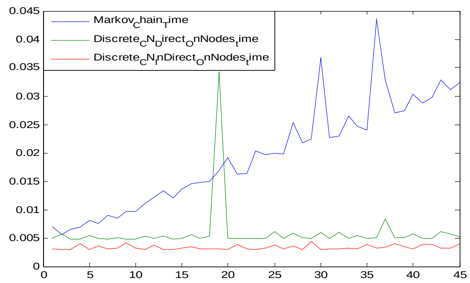
Duan, Jin-Chuan et al (2003) suggests using the Markov chain method to price discretely monitored barrier options. It provides a natural framework for this pricing process because the discrete time step of the Markov chain can be easily matched with the monitoring frequency of the barrier. The underlying asset price can also be partitioned so as to place the barrier suitably, independent of the time steps. The method can efficiently handle difficult cases where the barrier is close to the initial asset price. This method is suitable to the option.

Comparing to the traditional lattice scheme, the independence of price and time dimensions in the Markov chain approach allows us to adjust the time step of the Markov chain to exactly fit the barrier monitoring frequency without sacrificing the fineness of the asset price approximation. In comparison to the finite-difference (finite-element) approach, the method avoids the computational burden associated with the unnecessary refinement of time due to the numerical approximation of the partial differential equation.

Implementation

We implemented the Markov chain approach for knock out option only, and we use in-out parity to calculate the value of knock in. The results show it is a better method than traditional lattice method.





Pricing Results

1. **Markov chain is more accurate and fast than traditional multinominal method.**

The critical reason is that Markov chain can divide the space independent of the time steps. So, between the barriers, there can be more nodes involved into calculation. Another reason is that it is easy to put the critical values (such as barriers, and strike price) on the nodes, for the spatial dimension could be divided non-equally. The third reason could be that the spatial steps are nearly the branches in the multinominal, so the accuracy is higher.

2. **Markov chain seems nearly accurate as FD method, but Markov chain is slower than FD method.**

According to the code, compared to the CN method, Markov chain has a lot of norm distribution function calls and one matrix power operation. Maybe it is the reason that Markov chain is slower than CN method. Any way, the absolute gap is very small.

3. **Markov chain could be used in GARCH and other complicated environments, which could not be done by FD, which depends on PDE.**

Markov chain is very flexible, so it is strong to deal with non normal distribution and American options, which could not be done by FD.

4. **Extrapolation improves convergence for Markov chain method, but which makes a little real sense.**

It is obvious that Markov chain method have montonic convergence, so it is possible to use extrapolation to speed the convergence. At the same time, it is obvious also that even with traditional extrapolation, the results couldn't be improved too much, for the convergence speed is very high for Markov chain method.

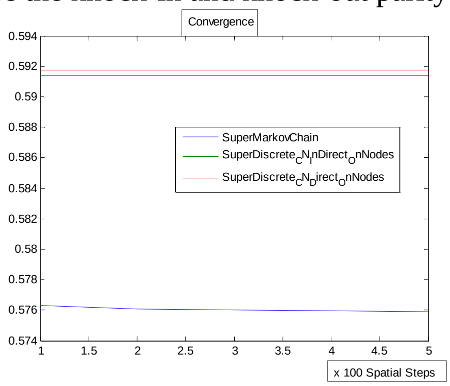
Monte Carlo Method

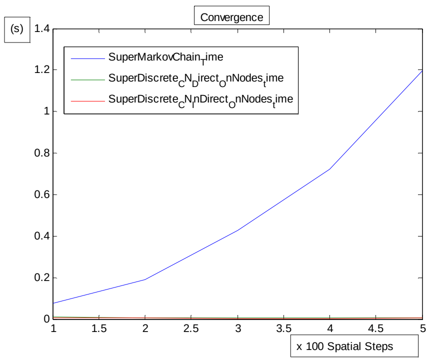
Monte Carlo Method is very general. It could be used nearly all kinds of options, but it is not suitable to options with early exercise, such as American options.

Monte Carlo method is independent of features of the option, except for some knock-out features. So, generally, convergence speed of MCS is not under control, which may be related to options, but a little related to MCS itself. Although there are some ways to improve the convergence speed, but MCS is still time consumed very greatly, especially for continuous monitoring options.

Conclusion

In this coursework, we are required to price a daily monitoring double barriers binary put option. We tried to use analytic (continuous monitoring only), finite difference, multinominal method, Markov Chain method to price the option and its corresponding continuous monitoring option. We also compared the direct pricing way and the way to use the knock-in and knock-out parity to pricing, an indirect way.





According to our work, we can draw the following basic conclusions:

1. Regarding to exotic options, multinominal method is not suitable, especially to the narrow double barriers options with discrete monitoring.

The main reason is that due to the discrete monitoring, the constructed tree is limited to be fine. Another reason is that the double barriers will reduce the constructed tree greatly, so the accuracy is weak. The third reason is that is that it is hard to deal with critical values, such as barriers and strike.

2. Finite Difference method and Markov Chain method are flexible and strong. They can deal with exotic options easily, especially for discrete monitoring.

   FD has two state variable, space and time, and unlike multinominal method, these two states are not linked together. With time interval is fixed, it is easy to divide the space discretionarily with some small techniques.

   Markov Chain method is more flexible, so it is strong to deal with non normal distribution and American options, which could not be done by FD.

3. Finite Difference has a fast convergence speed and the convergence could be monotonic with some improvements.
4. Sometimes, knock-in and knock-out parity can give us a fast and accurate indirect way to price options.

References

Ahn, D-H. (1999). Pricing Discrete Barrier Options with an Adaptive Mesh Model. Working Paper.

Alford, J. and N.Webber (2001).Very High Order LatticeMethods for One Factor Models. Working Paper.

D. M. Pooley, K. R. Vetzaly, and P. A. Forsyth (2002). Convergence Remedies For Non-Smooth Payoffs in Option Pricing.

Duan, Jin-Chuan et al. Pricing Discretely Monitored Barrier Options by a Markov Chain. Journal of Derivatives; Summer2003, Vol. 10 Issue 4, p9-31.

Duan, J.C. and J.G. Simonato, 1999, American Option Pricing under GARCH by a

George M. Jabbour et al. (2001?) Multinomial Lattices and Derivatives Pricing. Handouts.

Haug, Espen Gaarder (2006) The Complete Guide to Option Pricing Formulas. McGraw-Hill

Hui, C.H. (1996) One-Touch Double Barrier Binary Option Values.Applied Financial Economics, Vol. 6, pp. 343-346

Markov Chain Approximation, working paper,Hong-Kong University of Science and Technology.

Peter Forsyth and Ken Vetzal (2006).Numerical PDE Methods for Pricing Path Dependent Options. Handouts.

Rannacher, R. (1984). Finite element solution of diffusion problems with irregular data. Numerische Mathematik 43, 309-327.