

# **Black-Scholes**

방정식의  
수치해석 입문

# Black-Scholes

## 방정식의 수치해석 입문

• 정다래 / 김준석 공저

# 머리말

현재도 그렇지만 미래에도 금융산업은 중요한 국가 산업이 될 것이다. 첨단 금융지식을 습득해 금융시대를 준비해야 한다. 우리나라가 국가부도 위기를 맞아 국제 통화기금(IMF)에 구제 금융을 신청한지 어언 10여년이 되었다. 사상 처음 맞는 경제 파산위기에서 벗어나기 위해 쳐질한 구조 조정의 아픔을 겪어야만 했던 한국경제는 이제 새로운 도약의 전환점에 서 있다. IMF와 똑같은 위기는 재발하지 않았지만 과도한 가계부채, 자산 거품, 국제 금융 불안과 같은 새로운 위험 요소들은 아직도 상존하고 있다.

이 책에서는 여러가지 금융상품중에 옵션의 기본적인 가격결정모델과 모델의 수치해석을 다룬다. 옵션의 현실적인 복잡성으로 정확한 해를 찾기란 매우 어렵다. 이러한 이유로 우리는 수치해석을 이용한 옵션 계산에 관심이 있다. 초보자들에게 옵션 거래의 핵심사항들을 간단명료하게 해설하고 MATLAB 코드를 제시함으로써 기본적인 수치해석 원리와 기법을 쉽고 정확하게 이해 할 수 있도록 했다. 옵션에 대해서 일반상식, 경제학, 수학, 수치해석의 네 분야를 한권의 책으로 만들었다.

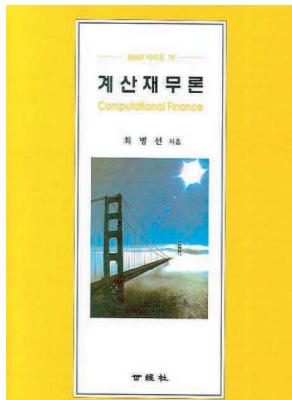
본 저서의 중점은 제 7장, 8장, 9장의 유한 차분법, Tree 가격결정모형, 그리고 몬테카를 시뮬레이션이나 내용의 완전성을 유지하기 위해 다른 주제들도 함께 담았다. 부족한 내용들은 서점에 나와있는 다른 참고 도서를 참조하면 될 것이라 사료된다. 이 책이 금융수학에 관심있는 독자들에게 조그마한 보탬이라도 될 수 있기를 바란다. 마지막으로, 제 8장 Tree 가격 결정모형과 제 9장 몬테칼로 시뮬레이션의 원고에 도움을 준 김경중님께 감사의 인사를 드린다. 무엇보다도 이 책을 꼼꼼히 살펴보면서 부족하고 서투른 부분에 많은 조언을 주신 이동선님과 양수현님에게 고마움을 전한



### Black-Scholes 방정식 입문

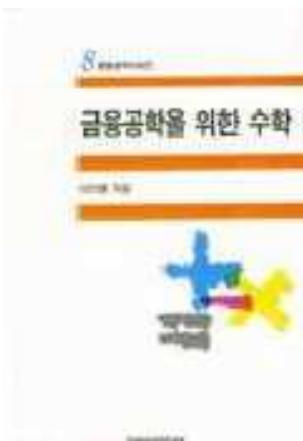
(최병선 김철웅)

금융시장에서 거래하기 위해서는 차본과 더 불어 지식이 필요하다. 이 책에서는 금융에 대한 기초실력을 다지기 위하여 편미분방정식을 이용하여 Black-Scholes식이 어떻게 유도되었는지를 밝힌다.



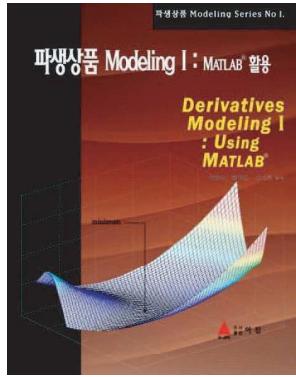
### 계산재무론 (최병선)

실천적 접근법을 통해서 금융파생상품론에 대한 이해를 돋는 계산재무론 교재. 이 책은 계산재무론의 필요성을 시작으로 금융파생상품의 가치평가, 계산재무기법들, 나무모형을 사용한 가치평가, 편미분방정식을 사용한 가치평가, 몬테카를로법을 사용한 가치평가 등을 차례대로 설명한다.



### 금융공학을 위한 수학 (이인형)

금융자산을 기초로 한 복합 상품인 파생상품의 이론가격 결정을 위해 필요한 기본적인 수학 지식을 다룬 전공서. 확률의 기본 개념, 확률과정, 금융공학을 위한 기초수학, 가격결정 패러다임 등 6개 장으로 설명했다. 파생상품 전문가와 대학원생용 책이다.



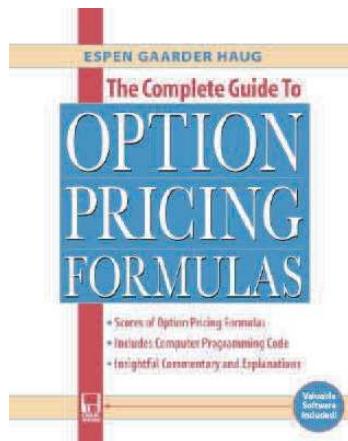
### 파생상품 Modeling I MATLAB 활용 (이경수 권명은 신진호)

파생상품에 대한 풍부한 실무 경험, 전문지식, 실무구현능력의 3박자를 갖춘다면 가장 이상적인 파생상품 전문가가 될 것이다. 이 책은 두 가지 능력배양에 초점을 맞추었다. 즉, 파생상품 평가와 리스크측정에 대한 전문지식을 체계적으로 설명하고 이를 MATLAB을 이용하여 실무에 구현하는 방법을 소개하였다.



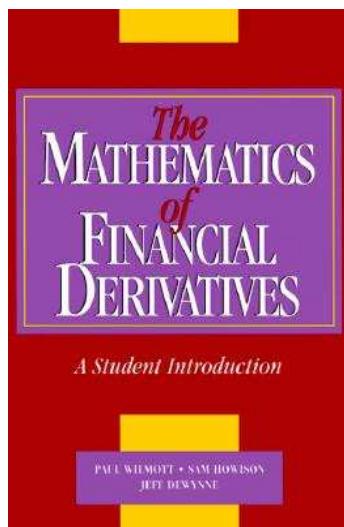
### 금융 증권을 위한 블랙솔즈의 편미분방정식 (김완세 옮김)

금융 증권을 위한 블랙 솔즈 미분 방정식 입문서. 이 책은 미분과 편미분, 테일러 급수 전개, 적분과 무한적분, 푸리에 해석과 편미분 방정식 해의 공식 등의 내용을 담았다.



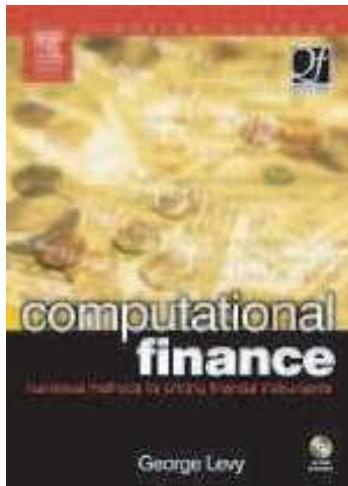
## **The Complete Guide to Option Pricing Formulas (Espen Gaardner Haug)**

The Complete Guide to Option Pricing Formulas contains thousands of formulas and explanations, including a ready-reference overview table for all option pricing formulas that gives you the fast answers you need - with more comprehensive information inside the book; chapters that feature plain vanilla options, exotic options, numerical methods in option pricing, interest rate options, volatility and correlation, and interpolation and simple interest rate calculations; examples and models that allow you to understand and benefit from author Espen Gaarder Haug's option pricing explanations and discussions without comprehensive programming knowledge; and a computer disk preprogrammed with source code and Excel spreadsheets, maximizing your understanding of the mathematical formulas by using them on your own computer. As a financial professional or student, have you ever needed an all-in-one reference for option pricing formulas and information? Look no further. The Complete Guide to Option Pricing Formulas is the first and only book to make the core set of tools and information available in one place. With its chapters covering every important area, plus written directions to help you best utilize the complimentary software package, The Complete Guide to Option Pricing Formulas is guaranteed to broaden your option trading horizons and help you wring the most value possible from each derivatives trade.



### The Mathematics of Financial Derivatives (Paul Wilmott)

Finance is one of the fastest growing areas in the modern banking and corporate world. This, together with the sophistication of modern financial products, provides a rapidly growing impetus for new mathematical models and modern mathematical methods. Indeed, the area is an expanding source for novel and relevant “real-world” mathematics. In this book, the authors describe the modeling of financial derivative products from an applied mathematician’s viewpoint, from modeling to analysis to elementary computation. The authors present a unified approach to modeling derivative products as partial differential equations, using numerical solutions where appropriate. The authors assume some mathematical background, but provide clear explanations for material beyond elementary calculus, probability, and algebra. This volume will become the standard introduction for advanced undergraduate students to this exciting new field.

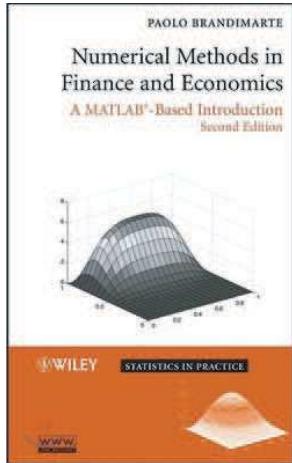


### **Computational Finance (Levy, George)**

This set contains two previously published books on computational finance: Computational Finance presents a modern computational approach to mathematical finance within the Windows environment. George Levy illustrates how numeric components can be developed by Financial Analysts that allow financial routines on the computer to be more easily performed. This book contains a bound in CD-ROM. In Computational Finance Using C and C++, Levy raises computational finance to the next level using the languages of both standard C and C++. The inclusion of both these languages enables readers to match their use of the book to their firm's internal software and code requirements. Levy also provides derivatives pricing information for equity derivatives, interest rate derivatives, foreign exchange derivatives, and credit derivatives. A unique password is bound into every book, giving the reader access to additional software on password protected website.

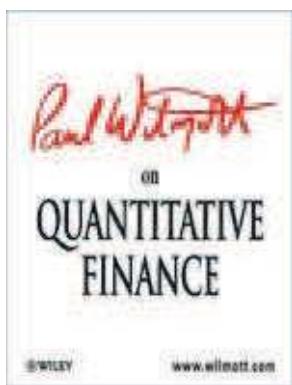
## Numerical Methods in Finance And Economics (Paolo Brandimarte)

A state-of-the-art introduction to the powerful mathematical and statistical tools used in the field of finance. The use of mathematical models and numerical techniques is a practice employed by a growing number of applied mathematicians working on applications in finance. Reflecting this development, Numerical Methods in Finance and Economics: A MATLAB-Based Introduction, Second Edition bridges the gap between financial theory and computational practice while showing readers how to utilize MATLAB the powerful numerical computing environment—for financial applications.



## Paul Wilmott on Quantitative Finance 3 Volume (Paul Wilmott)

Volume 3: Advanced Topics; Numerical Methods and Programs. In this volume the reader enters territory rarely seen in textbooks, the cutting-edge research. Numerical methods are also introduced so that the models can now all be accurately and quickly solved.



# 차 례

제 1 장	서론	23
제 2 장	파생금융상품 (Derivatives)	25
제 1 절	옵션의 개념과 주요 용어 . . . . .	25
제 2 절	옵션매입자와 옵션발행자 . . . . .	26
제 3 절	만기일 . . . . .	26
제 4 절	옵션의 상태 . . . . .	27
제 5 절	우리나라 옵션시장의 구조와 운영 . . . . .	28
제 6 절	거래제도 개요 . . . . .	28
6.1	거래소 . . . . .	28
6.2	호가단위 . . . . .	29
6.3	거래시간 . . . . .	29
6.4	거래량 . . . . .	29
제 7 절	매매제도 . . . . .	29
7.1	행사가격 . . . . .	29
7.2	호가방법 . . . . .	30
7.3	매매방식 . . . . .	30
7.4	매매거래 중단 . . . . .	30
7.4.1	사이드카(Sidecar) . . . . .	31
7.4.2	서킷브레이커(Circuit Breaker) . . . . .	31
제 8 절	결제 및 수탁제도 . . . . .	32

1.2.4	잔존기간( $t$ ) . . . . .	78
1.2.5	변동성( $\sigma$ ) . . . . .	79
<b>제 6 장</b>	<b>변동성 추정</b>	<b>81</b>
제 1 절	내재 변동성 . . . . .	81
1.1	뉴튼 래슨법(Newton-Rapson Method) . . . . .	82
<b>제 7 장</b>	<b>유한 차분법</b>	
	(Finite Difference Method)	<b>85</b>
제 1 절	개요 . . . . .	85
제 2 절	열 방정식에 대한 유한 차분법 . . . . .	87
2.1	명시적 (Explicit) 유한 차분법 . . . . .	87
2.1.1	명시적방법의 안정성 문제 - 폰 노이만 (von Neumann) 방법 . . . . .	89
2.2.1	토마스 알고리즘 (Thomas Algorithm) . . . . .	94
2.2.2	함축적 방법의 안정성 문제 - 폰 노이만 방법 . . . . .	97
2.3	크랭크 니콜슨 (Crank-Nicolson) 방법 . . . . .	98
2.3.1	크랭크 니콜슨 방법의 안정성 문제 - 폰 노이만 방법 . . . . .	101
2.4	수렴성 (convergence) 테스트 . . . . .	102
2.4.1	명시적 유한차분법 . . . . .	103
2.4.2	함축적 유한차분법 . . . . .	104
2.4.3	크랭크 니콜슨 유한차분법 . . . . .	106
제 3 절	Black-Scholes 편미분방정식에 대한 유한 차분법 . . . . .	109
3.1	명시적 방법에 의한 옵션 가격 결정 . . . . .	109
3.2	함축적 방법에 의한 옵션 가격 결정 . . . . .	111
3.3	크랭크 니콜슨 방법에 의한 옵션 가격 결정 . . . . .	113
3.4	안정성 테스트 . . . . .	116
3.4.1	명시적 유한차분법 . . . . .	116
3.4.2	함축적 유한차분법 . . . . .	118

2.5	기초자산간 상관관계를 반영한 난수생성 . . . . .	178
제 3 절	주가 경로(Stock Process) 시뮬레이션 . . . . .	180
제 4 절	옵션의 payoff 계산 . . . . .	182
제 5 절	payoff 들의 기대값 추정 . . . . .	182
제 6 절	옵션가치 도출 . . . . .	183
제 7 절	수치 분석 . . . . .	184

## 제 1 장

# 서론

본 저서는 서론을 제외하고 다음과 같이 총 8장으로 구성되어 있다.

제 2장은 파생금융상품에 대해서 다루었다.

제 3장은 MATLAB 기초이다. 본 저서의 MATLAB 코드를 이해하기 위해서 필수적으로 알아야 하는 명령어들의 설명이 있다.

제 4장에서는 브라운 운동을 소개하며 Itô Lemma를 유도한다.

제 5장은 기하옵션가격결정이론에 대해서 다루며 열방정식의 해석해로부터 Black-Scholes 방정식의 해석해를 유도한다.

제 6장은 내재 변동성을 뉴튼방법에 의해서 수치적으로 구하는 방법에 대해서 다루었다.

제 7장은 유한 차분법에 대해서 다루었다. 또한 Greek의 유한차분해를 구하여 해석해와 비교한다.

제 8장은 Tree 가격결정모형에 대해서 다루었다.

제 9장은 몬테카를 시뮬레이션에 대해서 다루었다.

## 제 2 장

# 파생금융상품 (Derivatives)

파생금융상품은 기초자산(underlying assets)의 가격변동으로 인한 손실위험을 제거하기 위해 탄생했다. 옵션계약의 대상이 되는 상품을 기초자산이라 하고 크게 곡물, 축산물, 귀금속 등과 같은 일반적인 상품(commodity)과 주식, 채권, 주가지수(KOSPI 200), 통화 등과 같은 금융상품으로 구분할 수 있고, 각각의 옵션을 상품 옵션(commodity option) 그리고 금융옵션(financial option)이라 한다.

## 제 1 절 옵션의 개념과 주요 용어

옵션(option)이란 특정 자산을 미리 정해진 가격(행사가격, strike price)으로 미래의 일정한 시기(만기, maturity)에 또는 일정한 기간 내에 매수 또는 매도할 권리가 내재된 계약을 일컫는다. 이를 유럽옵션(European option) 또는 아메리칸 옵션(American option)이라 한다. 이 때 특정자산을 매수할 권리를 콜옵션(call option), 매도할 수 있는 권리를 풋옵션(put option)이라고 한다.

그리고 새로이 2010년 9월 만기의 옵션이 상장되어 거래된다.

월	월	화	수	목	금	토
1	2	3	4	5	6	
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

월	월	화	수	목	금	토
1	2	3	4	5	6	7

월	월	화	수	목	금	토
1	2	3	4	5	6	7
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					

월	월	화	수	목	금	토
1	2	3	4	5	6	7
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30			

월	월	화	수	목	금	토
1	2	3	4	5	6	7
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30		

만기일이 가장 가까운 상품을 근월물이라 하고 그 다음이 만기인 순으로 차월물, 차차월물이라고 한다.

## 제 4 절 옵션의 상태

유리피언 콜옵션의 경우 만기일에 기초자산의 시장가격이 행사가격보다 낮은 경우 옵션은 옵션매입자는 옵션권리의 취득에 의해 옵션 매도자에게 정해진 기간에 옵션계약의 내용에 대한 이행을 청구할 수도 있으며, 반대로 자신에게 불리한 경우 옵션계약을 포기할 수 있다. 가치를 지니지 못하게 되므로 옵션은 행사되지 않고 소멸되고 만다. 이와 같이 옵션의 권리를 행사하는 것이 이득이 되지 않는 상태를 외가격(out-of-the-money; OTM)상태라 하고 이 때의 옵션을 외가격옵션이라 부른다.

옵션의 행사가격이 기초자산의 시장가격과 동일하여 옵션을 행사하거나 행사하지 않는 경우가 무차별한 경우를 등가격상태라 하며 이러한 옵션을 등가격옵션(at-the-money option; ATM option)이라 한다. 한편, 시장에서 기초자산의 가격상황이 옵션을 행사하는 것이 유리한 경우, 예를 들어 콜옵션의 경우에는 기초자산의 시장가격이 행사가격보다 높은 상태를 내가격상태라 하며 이러한 옵션을 내가격옵션(in-the-money option; ITM option)이라 한다.

## 6.2 호가단위

1계약의 크기는 지수 1포인트당 10만원이며 호가가격 변동폭은 옵션가격이 3포인트 이상일 경우는 0.05포인트, 즉 5,000원이고 3포인트 미만일 경우는 1,000원, 즉 0.01포인트이다. 옵션거래는 가격제한폭이 없지만 시장안정을 위해서 호가가격이 전일의 대상자산 가격대비 ±15%를 벗어나는 경우 거래소에서 호가접수를 거부하는 호가한도 가격제도를 두고 있다.

## 6.3 거래시간

우리나라 주가지수 옵션거래는 주식시장의 개장 시간인 9시에 거래를 시작하며 주식시장 종료시점보다 15분 늦은 3시 15분에 거래가 종료된다. 이는 확정된 주가 지수에 따라 옵션 포지션을 조정할 수 있도록 하기 위함이다. 그러나 결제일에는 주식시장보다 10분 일찍 2시 50분에 거래를 종료하도록 되어 있다. 이러한 이유는 주식시장이 종료되기 전까지의 10분간 거래가격의 왜곡가능성을 배제하기 위해서이다. 근월물의 결제일의 옵션은 2시 50분에 매매가 종료되지만 다음월물의 거래는 계속 지속되며 공휴일의 경우는 거래가 되지 않는다.

## 6.4 거래량

매수호가와 매도호가가 일치하면 거래가 체결된다. A가 10계약 매수주문을 내고 동일한 금액으로 B가 30계약 매도주문을 내게 되면, 주문이 일치된 10계약이 이루어지는데 이 때 거래가 체결된 10계약이 거래량이 된다.

# 제 7 절 매매제도

## 7.1 행사가격

행사가격은 전날의 KOSPI 200 종가에 가장 가까운 행사가격과 2.5포인트씩 높인 행사가격 4개 그리고 2.5포인트씩 낮춘 행사가격 4개를 제시함으로써 모두 9개 행사가격을 가진 옵션이 거래된다. 단, 3, 6, 9, 12월이 결제일인 옵션은 5포인트 간격으로 5개의 행사가격이 설정된다.

#### 7.4.1 사이드카(Sidecar)

사이드카는 프로그램 매매호가 관리제도의 일종으로, 주가지수선물시장에서 선물 가격이 급등락할 경우 선물시장의 충격이 현물 시장에 파급되는 것을 완화시키기 위하여 일시적으로 프로그램매매의 호가효력을 중단시키는 것이다. 선물의 거래중지와 동시에 옵션의 거래도 중지된다.

우리나라에서 사이드카의 발동요건은 KOSPI선물시장의 경우, 전날의 거래량이 가장 많았던 종목의 선물가격이 기준가격(전일종가) 대비 6% 이상 상승하거나 하락한 상태에서 코스닥스타 현물지수(선물거래대상지수)도 3% 이상 같은 방향으로 변동해 1분간 지속될 때 발동된다. 사이드카가 발동되면 주식시장의 프로그램 매매 호가가 5분간 효력이 정지된다. 5분 후에는 정상거래가 되며, 사이드카는 하루에 한 번만 쓸 수 있다. 또 주식 시장 매매거래 종료 40분 전 이후, 즉 오후 2시 20분 이후에는 발동되지 않는다. 종전에는 선물가격이 6% 이상 등락한 상태에서 1분간 지속될 경우 사이드카가 발동되었지만, 젖은 비현실적 경보음으로 오히려 시장을 왜곡 한다는 비판을 받아온 코스닥시장의 사이드카 발동요건이 2009년 7월 6일 월요일을 기점으로 대폭 강화되었다.

#### 7.4.2 서킷브레이커(Circuit Breaker)

종합주가지수가 직전 매매거래일보다 10% 이상 하락하여 1분간 지속되는 경우에는 주식시장의 모든 매매거래를 중단시키며 옵션의 거래도 중지되는 제도를 서킷브레이커라고 한다. 서킷 브레이커가 발동되면 20분 동안 시장 내 호가접수와 채권시장을 제외한 현물시장과 연계된 선물·옵션시장도 호가접수 및 매매거래를 중단한다. 서킷 브레이커가 발동된 후 20분이 지나면 매매거래를 재개되는데 이 때 시작가격은 재개시점부터 10분 동안 호가를 접수하여 단일가로 매매를 체결하여 거래가 다시 시작된다. 사이드카와 마찬가지로 하루에 1회만 발동할 수 있고, 장 종료 40분전 이후에는 발동하지 않는다.

또한 매매종료 후 위탁증거금에서 손실을 차감한 금액이 옵션의 미결제약정에 대한 유지증거금에 미달하는 경우 그 차액을 다음날 12시까지 증권회사에 추가로 납부해야 한다. 만약 기한까지 납부하지 않을 경우 증권회사는 임의로 반대매매를 하거나 예탁된 대용증권을 매각할 수 있다.

### 8.5 미결제약정 수량의 관리

미결제약정이란 옵션거래가 성립된 이후 만기일까지 반대매매, 권리행사, 또는 최종결제 등으로 청산되지 않은 약정을 말한다. 매수 미결제 약정을 보유하고 있는 것을 “매수 포지션(Long Position)을 취하고 있다.” 라고 말하고 매도 미결제 약정을 보유하고 있는 것을 “매도 포지션(Short Position)을 취하고 있다.” 라고 말한다.

### 8.6 위탁수수료

증권회사는 투자자의 위탁주문에 대한 매매거래가 성립하거나 최종결제 또는 권리행사에 의한 결제가 발생한 때 투자자로부터 증권회사가 정한 위탁수수료를 징수한다.

- 시가 ( 209.43 ) : 당일 최초로 형성된 지수
- 고가 ( 209.99 ) : 하루 중 가장 높은 지수
- 저가 ( 205.80 ) : 하루 중 가장 낮은 지수
- 행사가격 ( 210.00 ) : 옵션매입자가 만기일 또는 그 이전에 권리를 행사할 때 적용되는 가격

\* 이 장의 주요 용어

- 파생금융상품
- 유러피언콜옵션
- 프리미엄
- 옵션만기일
- KOSPI 200
- 사이드카
- 서킷브레이커
- 위탁증거금

## 제 3 장

# MATLAB 기초

MATLAB은 미국의 Math Works에서 만들어진 프로그램으로, 1984년도에 소개된 이후로 오늘날 전 세계 50만 이상이 사용하고 있다. MATLAB은 MATrix+LABoratory로서 행렬을 기본으로 최적화되어진 프로그램으로 알고리즘 개발, 데이터 수치분석이나 시각화를 위한 컴퓨터 언어이다. C언어에 비해 사용하기가 편리하다는 장점이 있으나 스크립트 기반으로 C 프로그램보다는 실행속도가 느리다는 단점을 안고 있다.

새로운 장외파생상품들의 출현으로 주로 공학계통에서 사용되던 MATLAB은 편리한 사용방법으로 최근 모델링(Modeling)을 위한 프로그래밍 언어로 인기를 얻어가고 있으며 이 책에서도 모든 코드는 MATLAB언어로 구현하였다.

다음은 Command Window를 사용할 때 기본적으로 알아두어야 할 것들이다.

- MATLAB은 대소문자를 구분한다.
- 변수의 이름을 타이핑하면 MATLAB은 그 변수의 현재 값을 스크린에 보여준다.
- MATLAB은 괄호(), 대괄호[], 중괄호{}를 사용하며 이들은 서로 교환될 수 없다.
- 위쪽 화살표(↑)와 아래쪽 화살표(↓)을 이용하면 이전에 입력한 명령문들을 순차적으로 검색할 수 있다. 아울러 첫 번째 몇 글자를 타이프하고 위쪽 화살표(↑)키를 누르면 예전의 명령어들을 다시 불러올 수 있다.
- 알아보고자 하는 주제가 있을 경우 **help**을 입력하면 명령어, 함수, 혹은 관심 있는 주제에 대한 도움말을 접할 수 있다. 밑줄로 표시된 하이퍼링크들이 제시되어 있어서 관련되는 도움 항목들 및 도움말 탐색 기로 접속할 수 있다.
- 함수나 변수 이름을 일부분만 타이프하고 텭 키를 누르면 MATLAB은 나머지 부분을 완성할 수 있는 경우들을 모두 제시하여 줌으로써 사용자가 한가지를 선택하여 이름의 나머지 부분을 완성할 수 있게 하여 준다.
- MATLAB을 끝내려면 **exit**나 **quit**를 입력한다.

MATLAB 프로그래밍 작업 시 외부의 데이터를 표출하여 입력하거나 결과 값들을 외부로 출력하는 작업이 필요할 때가 많다. 다음은 파일 입출력에 대한 기본적인 기능을 나타내는 표이다.

명령어	기능	사용방법
open	파일 열기	open('name.m')
load	파일 불러오기	load('name.m')
save	저장하기	save('filename', 'var1','var2')

## 제 1 절 MATLAB 입문

MATLAB에 들어오면 텍스트를 입력하여 이 장의 내용들을 살펴보아야 하는데 입력된 텍스트는 명령어 창에서 MATLAB 프롬프트(prompt)>> 다음에 나타난다. 입력할 내용을 보여준 다음에 생성되는 출력이 제시된다. 다음 문장으로 시작된다.

```
>> a = 1 
a = 1
```

“ $a = 1$ ”이라고 타이핑한 뒤 엔터(Enter)키를 누르면 MATLAB으로부터 출력 “ $a =$ ”와 “1 2 3”이 두 줄로 분리되어 제시된다. 이제부터는 공간을 절약하기 위해 MATLAB 출력의 빈 줄들을 생략할 것이다. **format compact**를 입력하면 MATLAB은 빈 줄들을 줄여준다.

다음의 보기는  $3 \times 1$  배열  $b$ (열 벡터)를 설정한다.

```
>> b = [1 2 3] 
b = 1 2 3
```

새 줄로 시작하도록 지시하는 세미콜론(;)을 이용하면  $1 \times 3$ 배열  $c$ (행 벡터)를 설정할 수 있다.

```
>> c = [1; 2; 3;] 
```

행렬과 벡터에 대한 연산 조작은 두 가지의 서로 다른 형태로 이루어진다. 행렬 측면에서의 연산은 선형대수학의 통상적인 규칙에 근거하며 기호  $+, -, \star, /$  및  $\wedge$ 들을 사용하여 이루어진다. 배열 측면에서의 연산은 원소끼리의 작용으로 정의되며 일반적으로 위의 기호들 앞에 점(.)을 두어 이루어진다. 따라서 b의 각 원소들의 제곱을 구하고자 한다면 다음과 같이 쓸 수 있다:

```
>> c = b.^2 
c = 1      4      9
```

위에서 얻은 새로운 벡터 c는 b와 마찬가지로  $1 \times 3$ 이므로 c와 a의 배열 곱을 얻을 수 있다.

```
>> b.*c 
ans = 1      8      27
```

어떤 형태의 행렬들은 자동적으로 구축할 수 있다. 예를 들면 원소들이 0으로 이루어지는 행렬은 zeros을 이용하여 구성할 수 있다:

```
>> d = zeros(1,3) 
d = 0      0      0
```

위에서 얻은 d는 1행 3열의 원소들이 모두 0으로 이루어지는 행렬이 된다.

다음은 행렬  $A$ 를 Transpose한 것이다. Transpose는 행렬의 행과 열을 서로 바꾸는 것을 의미하며,  $A^T$ 라고 쓴다. 이제 이를 MATLAB에서 구해보도록 하자.

$$A = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix} \implies A^T = \begin{pmatrix} 1 & 4 & 7 \\ 2 & 5 & 8 \\ 3 & 6 & 9 \end{pmatrix}$$

```
>> A = [1 2 3; 4 5 6; 7 8 9] 
A = 1      2      3
     4      5      6
```

```

6      8
10     12
>> A-B  [Enter]
ans =
-4     -4
-4     -4

```

여기서 주의할 점은 두 행렬의 크기가 동일해야 행렬의 덧셈과 뺄셈을 할 수 있다는 것이다.

$$\begin{aligned}
 A + 1 &= \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} + \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} = \begin{pmatrix} 2 & 3 \\ 4 & 5 \end{pmatrix} \\
 A - 1 &= \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} - \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 2 & 3 \end{pmatrix}
 \end{aligned}$$

위의 연산처럼 행렬에 상수를 더하거나 빼면 MATLAB에서는 다음과 같이 1을 원소로 갖는 행렬로 인식하고 계산을 한다.

```

>> A + 1
ans =
2      3
4      5
>> A - 1
ans =
0      1
2      3

```

다음은 각각의 행렬에 대한 사칙연산이다.

연산자 앞에 .이 붙게 되면 행렬의 같은 위치에 있는 각각의 원소끼리 연산을 수행하라는 의미이다. 원소끼리의 연산이므로 반드시 두 행렬의 크기는 정확하게 일치해야 한다.

이제 행렬의 일부나 전체를 불러오는 표현을 배워보자. 간단하게

```
>> A = [1 1 1 1; 2 2 2 2; 3 3 3 3; 4 4 4 4] 
A =
    1     1     1     1
    2     2     2     2
    3     3     3     3
    4     4     4     4
>> A(2, 1:4) 
ans =
    2     2     2     2
>> A(2, 1:end) 
ans =
    2     2     2     2
>> A(2,:) 
ans =
    2     2     2     2
>>
```

위의 MATLAB 코드에서 볼 수 있듯이 세 개의 표현은 모두 동일한 표현임을 알 수 있을 것이다. 모두  $A$  행렬의 2행의 1열부터 끝까지의 원소를 나열하라는 것으로, 여기서 사용된 end은 행렬의 크기의 끝을 나타내며, :은 행렬에서 전체를 나타내는 표현이다.

MATLAB의 나머지 연산인 mod에 대하여 알아보자. 예를 들어,

$$10 = (-4) * (-3) - 2$$

의 계산을 MATLAB에서 mod을 사용하면, 다음의 결과를 출력하게 된다.

```
>> mod(10, -4) 
```

```
a = 1;
b = 2;
c = a+b
```

위의 문장을 Editor에 적어 넣고, 단축키 F5를 누르자. 파일이름은 원하는 이름으로 쓰고 저장하면 원하는 결과를 얻을 수 있게 된다.

### 제 3 절 for ~ end 문

'for' 문은 'end'문과 짹을 이루어 사용된다. 'for'문과 같은 행에 있는 변수의 값을 초기값부터 증분의 크기만큼 누적시키면서 최종값에 도달된 때까지 'for'문과 'end'문 사이 문장의 명령을 수행한다. 증분이 '1'인 경우에는 '증분:'을 생략해도 무방하다.

for 변수명=초기값:(증분:)최종값

문장

end

#### [예제] for ~ end 프로그램

```
for x=0:0.5:1
    a=2^x
end
for k=5:-2:1
    b=k
end
```

[프로그램 실행결과]

```
a=3;  
if a<1  
    b=a+1  
elseif a<=2  
    c=a+2  
elseif a==3  
    d=a+3  
end
```

[프로그램 실행결과]

```
d = 6
```

## 제 5 절 while ~ end 문

‘while’문은 ‘end’문과 짹을 이루어 사용된다. ‘while’문과 같은 행에 있는 조건이 참이면 ‘while’문과 ‘end’문 사이에 있는 문장의 명령을 반복적으로 수행한다.

```
while 조건  
    문장  
end
```

[예제] while ~ end 문 프로그램

```
x = 0 1 2 3 4 5  
y = -1 -0.5 0 0.5 1
```

## 제 7 절 plot 문

실행결과를 2차원 그래프로 나타내고자 할 때 사용된다. ‘plot’문을 사용하기 위해서 2개의 변수가 필요하며 각 변수는 같은 크기의 1차원 배열(벡터)이어야 한다.

### [예제] plot 문 프로그램

```
for i=1:21  
    x(i)=0.1*(i-1);  
    y(i)=x(i)^2;  
end  
plot(x,y)  
title('Graph of y=x^2')  
xlabel('X')  
ylabel('Y')  
grid
```

[프로그램 실행결과]

## 제 4 장

# 브라운 운동 (Brownian Motion)

### 제 1 절    브라운운동(Brownian Motion)

금융문제와 같이 시간과 연동된 변동에 대한 수학적 모델은 대개 브라운 운동과 같은 확률과정을 사용하여 나타낸다. 브라운 운동의 엄밀한 수학적 기초를 닦은 미국의 수학자 Norbert Wiener의 이름을 따서 브라운 운동을 위너과정(Wiener Process)이라고도 한다. 금융시장에서 기초자산의 가격경로는 위너과정을 따른다고 가정하는 경우가 많다. 이 절에서는 위너과정에 대해 자세히 알아보기로 한다.

정규분포(normal distribution)를 따르는 확률변수  $X \sim N(\mu, \sigma^2)$ 의 확률밀도함수(probability density function)  $f(X)$ 는 다음과 같다.

$$P(X = x) = f(x) = \frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

확률밀도함수(probability density function)를 간단히 기호 pdf로 표시한다.

$$dX = \sqrt{dt}$$

로 된다.

## 제 2 절 일반화된 위너 과정

랜덤 워크나 그 극한의 위너 과정을 이용하면 주가변동의 모양을 표현할 수 있다. 시계열  $S(t)$ 의 모양은 기울기  $a$ 의 직선  $at$ 와 위너 과정에 따른 시계열  $X(t)$ 의 그래프가 합체된 것으로 표현 가능하다. 즉,

$$S(t) = at + bX(t)$$

시계열  $S(t)$ 의 변화량  $dS$ 는

$$dS = adt + bdX$$

로 된다. 이와 같은 시계열  $X(t)$ 의 움직임을 일반화된 위너 과정이라 한다. 그런데 정규분포의 성질을 사용하면 다음의 사실을 알 수 있다.

### 위너 과정

시계열  $X(t)$ 가 위너 과정을 따라 움직이고 있을 때

“ $X(t)$ 는 평균 0, 표준편차  $\sqrt{dt}$ 의 정규분포  $N(0, dt)$ 를 따름”

이므로

### 일반화된 위너 과정

시계열  $S(t)$ 가 일반화된 위너 과정에 따라 움직이고 있다면

$$S(t) = adt + bX$$

는 평균  $adt$ , 표준편차  $b\sqrt{dt}$ 의 정규분포  $N(ad, b^2dt)$ 을 따름.

이 되는 것을 알 수 있다.

$I$ 가 양수이므로,  $I = 1$ 이다. 다음 식이 성립한다.

$$\int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x-(\sigma^2 t + \mu))^2}{2\sigma^2}\right) dx = 1.$$

따라서, 다음 식이 성립한다.

$$M_X(t) = \exp\left(\mu t + \frac{\sigma^2}{2}t^2\right).$$

이 식을 미분하면, 다음 식들이 성립함을 알 수 있다.

$$\begin{aligned} \frac{dM_X(t)}{dt} &= (\mu + \sigma^2 t) \exp\left(\mu t + \frac{\sigma^2}{2}t^2\right), \\ \frac{d^2M_X(t)}{dt^2} &= \sigma^2 \exp\left(\mu t + \frac{\sigma^2}{2}t^2\right) + (\mu + \sigma^2 t)^2 \exp\left(\mu t + \frac{\sigma^2}{2}t^2\right). \end{aligned}$$

따라서, 다음 식들이 성립한다.

$$\begin{aligned} E(X) &= M_X'(0) = \mu, \\ E(X^2) &= M_X^{(2)}(0) = \sigma^2 + \mu^2, \\ Var(X) &= E(X^2) - E(X)^2 = \sigma^2 + \mu^2 - \mu^2 = \sigma^2. \end{aligned}$$

### 제 3 절 자산 가격을 위한 간단한 모델

효율적 시장가설(effective market hypothesis)은 시장에서 기본적 정보와 기술적 정보를 포함한 과거의 모든 정보가 금융자산의 가격에 반영되어 있다고 가정한다. 먼저, 자산 가격에서 절대적인 변화는 그 자체로서 유용한 정보는 아니다. 예를 들어 1포인트의 변화는 자산 가격이 200p보다 20p일 때 좀 더 의미 있다는 것이다. 따라서 변화의 이런 상대적 측정은 어떠한 절대적 수치보다 그것의 척도를 좀 더 명확하게 나타낼 수 있다.

시점이  $t$ 일 때 기초자산 가격을  $S$ 라고 가정하자. 그럼 4.1처럼  $S$ 가  $S + dS$ 로 변하는 작은 시간구간  $dt$ 를 생각해보자. 그렇다면  $dS/S$ 인 상대적 변화를 어떻게 모형화할 수 있을까? 가장 일반적인 접근법은  $dS/S$ 를  $\mu dt$ 와  $\sigma dX$ 의 두 부분으로 나누는 것이다.

$\mu dt$ 는 무위험채권에 투자한 금액의 수익처럼 예측가능하고 확정적인 것을 나타낸다.  $\mu$ 는 drift로써  $dt$ 시간동안 자산가격의 평균성장을의 측정

여기서  $S_0$ 는  $t = t_0$ 일 때 자산의 가치를 나타낸다. 그러므로  $\sigma = 0$ 이라면 정확한 미래의 자산 가격을 예측할 수 있게 된다. 자산 가격의 특징인 무작위성(randomness)을 포함하는  $dX$ 는 위너과정(Wiener process)을 따르고 다음의 성질을 갖는다;

- $dX$ 는 정규분포로부터 나온 확률 변수(random variable)이다.
- $dX$ 의 평균은 0이다.
- $dX$ 의 분산은  $dt$ 이다.

이것을 나타내는 한 가지 방법은

$$dX = \sqrt{dt}\phi$$

로서, 여기서  $\phi$ 는 표준정규분포로부터 나온 확률 변수가 된다. 표준정규분포는  $-\infty < \phi < \infty$ 에서 확률 밀도 함수

$$\frac{1}{\sqrt{2\pi}}e^{-\frac{1}{2}\phi^2} \quad (4.3)$$

를 갖는다. 표준정규분포는 다음의 성질을 갖는다.

$$\mathcal{E}[\phi] = 0, \quad \mathcal{E}[\phi^2] = 1. \quad (4.4)$$

위의 식 (4.4)을 갖는 이유는 다음과 같다.  $\phi$ 의 기댓값은 다음과 같이 정의 할 수 있다.

$$\mathcal{E}[\phi] = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \phi e^{-\frac{1}{2}\phi^2} d\phi. \quad (4.5)$$

식(4.5)의 편적분 함수  $\phi e^{-\frac{1}{2}\phi^2}$ 은 기함수가 되므로, 0을 기준으로 대칭 함수가 되므로  $\mathcal{E}[\phi] = 0$ 이 된다는 것을 알 수 있다. 이제,  $\mathcal{E}[\phi^2]$ 는 다음과 같이 정의할 수 있다.

$$\mathcal{E}[\phi^2] = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \phi^2 e^{-\frac{1}{2}\phi^2} d\phi. \quad (4.6)$$

식(4.6)을 가지고, 모멘트 생성함수(Moment generating function)를 적용시키자. 모멘트 생성함수는 다음으로 정의된다.

$$M_\phi(t) = \mathcal{E}[e^{t\phi}], \quad t \in \mathbb{R} \quad (4.7)$$

이므로,  $M''_\phi[0] = 1$ 이 된다.

$$\therefore \mathcal{E}[\phi^2] = 1. \quad (4.13)$$

따라서, 식 (4.4)의 성질을 갖게 되는 것이다.

## 제 4 절 이또과정(Itô Process)

시계열  $S(t)$ 의 변화량  $dS$ 가 다음의 식에 따라 움직이고 있다고 하자.

$$dS = a(S, t)dt + b(S, t)dS.$$

이 시계열  $S(t)$ 의 움직임을 이또과정이라 하고 이 과정을 이용하여 이 또의 보조정리(Itô lemma)을 이끌 수 있다.

### 이또의 보조정리

$S$ 가 이또과정

$$dS = a(S, t)dt + b(S, t)dX$$

를 따를 때,  $S$ 와  $t$ 의 함수  $V(S, t)$ 의 동향은

$$dV = \left( \frac{\partial V}{\partial S} a(S, t) + \frac{\partial V}{\partial t} + \frac{1}{2} \frac{\partial^2 V}{\partial S^2} b^2(S, t) \right) dt + \frac{\partial V}{\partial S} b(S, t)dX$$

를 따른다.

(증명) 함수  $V$ 의 변화량  $dV$ 는 2변수함수의 테일러 전개를 이용하면

$$\begin{aligned} dV &= V(S + dS, t + dt) - V(S, t) \\ &= \frac{\partial V}{\partial S} dS + \frac{\partial V}{\partial t} dt + \frac{1}{2} \frac{\partial^2 V}{\partial S^2} (dS)^2 + \frac{\partial^2 V}{\partial S \partial t} dS dt + \frac{1}{2} \frac{\partial^2 V}{\partial t^2} (dt)^2 + \dots \end{aligned}$$

로 나타난다.  $S$ 가 이또과정을 따르므로,  $dS$ 에

$$dS = a(S, t)dt + b(S, t)dX$$

## 제 5 장

# 옵션가격이론

### 제 1 절 옵션 가격 결정 모형의 Black-Scholes 편미분 방정식 만드는 법

Black-Scholes 편미분 방정식(partial differential equation:PDE)은 미국의 Fisher Black 교수와 Myron Scholes 교수에 의해 개발된 옵션 가격 결정 모형으로서 옵션 이론 가격을 산출할 때 일부 수정된 모델이 현재 널리 이용되고 있다. 이 모델을 이용하면 기초자산가격( $S$ ), 행사가격( $E$ ), 잔존기간( $T$ ), 무위험이자율( $r$ ), 기초자산가격의 변동성( $\sigma$ )의 값들로 콜옵션과 풋옵션의 이론가격을 직접 계산할 수 있다.

Fisher Black과 Myron Scholes는 확률미적분학을 사용하여 유러피언 옵션에 대한 공정가격을 결정하기 위한 편미분 방정식을 유도했다. 이러한 Black-Scholes 이론은 기본적으로 주식(위험자산)과 채권(무위험자산)으로 이루어진 포트폴리오의 만기 가치가 ‘주식시장이 어떤 상황을 거치더라도’ 정확히 옵션의 만기 payoff와 일치하도록 하는 트레이딩 전략을 구체적으로 제시하여 준다. 1970년 Black-Scholes 공식의 개발과 함께 1973년 개설된 시카고 옵션거래시장은 세계의 옵션시장을 크게 발전시키는 역사적 계기가 되었다.

옵션 이론가격을 구하기 위해서는 다음의 다섯 가지 변수를 반드시 알아야 한다.

이다. 여기서,  $\frac{\partial V}{\partial S}$ 는 상수로 간주한다. 이 식의  $dV$ 와  $dS$ 에 각각 이또 램마(5.2)와 이또과정 (5.1)를 대입하면

$$\begin{aligned} d\Pi &= dV - \frac{\partial V}{\partial S} dS \\ &= \left( \frac{\partial V}{\partial S} \mu S + \frac{\partial V}{\partial t} + \frac{1}{2} \frac{\partial^2 V}{\partial S^2} \sigma^2 S^2 \right) dt - \frac{\partial V}{\partial S} \sigma S dX \\ &\quad - \frac{\partial V}{\partial S} (\mu S dt + \sigma S dX) \\ &= \left( \frac{\partial V}{\partial t} + \frac{1}{2} \frac{\partial^2 V}{\partial S^2} \sigma^2 S^2 \right) dt. \end{aligned} \quad (5.4)$$

한편, 무위험 이자율을  $r$ 로 하면  $dt$  시간동안 포트폴리오의 변화량은

$$d\Pi = r \left( V(S, t) - \frac{\partial V}{\partial S} S \right) dt \quad (5.5)$$

가 성립되는 셈이다. 따라서, 두식 (5.3)과 (5.4)로 부터 Black-Scholes 편미분방정식,

$$\frac{\partial V(S, t)}{\partial t} = rV(S, t) - rS \frac{\partial V(S, t)}{\partial S} - \frac{\sigma^2 S^2}{2} \frac{\partial^2 V(S, t)}{\partial S^2} \quad (5.6)$$

을 도출하게 된다. 여기에 옵션의 만기시점에서 지불되는 지불금액함수를 경계조건으로 하면

$$V(S, T) = \begin{cases} S - E & \text{if } S \geq E \\ 0 & \text{if } S < E \end{cases} \quad (5.7)$$

식 (5.6)은 식(5.7)의 만기조건을 가지는 Black-Scholes의 편미분 방정식이다.

## 1.1 Black-Scholes 편미분방정식의 공식

이 절에서는 Black-Scholes 편미분방정식의 해를 구한다.<sup>1</sup> 우선 방정식을 열 방정식(heat equation) 형태로 전환하고 해를 구한 후 다시 치환을 통해서 원래 방정식인 Black-Scholes 편미분방정식의 해를 구한다. Black-Scholes 편미분방정식 (5.6)에 대해서 두 개의 변수  $x$ 와  $\tau$ 를 도입하자

---

<sup>1</sup> 이 절에서 더 자세한 내용은 금융증권을 위한 블랙숄즈의 편미분방정식(김완세 옮김)의 책을 참고하길 바란다.

위 계산결과를 Black-Scholes 편미분방정식 (5.6)에 대입해보면, 다음과 같이 열방정식 형태로 나타낼 수 있다.

$$u_\tau(x, \tau) = \frac{\sigma^2}{2} u_{xx}(x, \tau). \quad (5.10)$$

여기서,  $x = \log \frac{S}{E} + (r - \frac{\sigma^2}{2})(T - t)$ 에  $t = T$ 를 대입하면,  $x = \log \frac{S}{E}$  그러므로  $S = Ee^x$  이 된다. 따라서, 경계조건 (5.7)에  $t = T$ 를 대입하면

$$u(x, 0) = \begin{cases} V(Ee^x, T) = E(e^x - 1) & \text{if } x \geq 0 \\ V(Ee^x, T) = 0 & \text{if } x < 0 \end{cases} \quad (5.11)$$

로 변한다. 이제 Fourier 변환을 사용하여 식(5.10)을 풀고자 한다.

열방정식 형태인 식(5.10)를 만족하는 함수  $u(x, \tau)$ 의 변수  $x$ 에 대한 Fourier 변환은 다음과 같다.

$$\hat{u}(\lambda, \tau) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} u(x, \tau) e^{-i\lambda x} dx. \quad (5.12)$$

또한, 이계편도함수  $u_{xx}$ 의 Fourier 변환은 부분적분을 두 번 반복 적용하면 다음과 같다.

$$\frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} u_{xx}(x, \tau) e^{-i\lambda x} dx = (i\lambda)^2 \hat{u}(\lambda, \tau) = -\lambda^2 \hat{u}(\lambda, \tau). \quad (5.13)$$

식 (5.12)로부터 다음 식이 성립함을 알 수 있다.

$$\begin{aligned} \frac{\partial \hat{u}(\lambda, \tau)}{\partial \tau} &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} u_\tau(x, \tau) e^{-i\lambda x} dx = \frac{\sigma^2}{2} \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} u_{xx}(x, \tau) e^{-i\lambda x} dx \\ &= -\frac{\sigma^2 \lambda^2}{2} \hat{u}(\lambda, \tau). \end{aligned} \quad (5.14)$$

식 (5.14)로부터 다음 식이 성립한다.

$$\frac{\partial \hat{u}(\lambda, \tau)}{\partial \tau} + \frac{\sigma^2 \lambda^2}{2} \hat{u}(\lambda, \tau) = 0. \quad (5.15)$$

식 (5.15)은  $\hat{u}(\lambda, \tau)$ 의  $\tau$ 에 관한 상미분 방정식이다. 식 (5.15)은 분리가능 방정식이라고도 하며, 이는 다음과 같이 나타낼 수 있다.

$$\frac{1}{\hat{u}(\lambda, \tau)} \partial \hat{u}(\lambda, \tau) + \frac{\sigma^2 \lambda^2}{2} \partial \tau = 0.$$

제 다음 식들이 성립함을 쉽게 알 수 있다.

$$\begin{aligned} \frac{dI(\alpha)}{d\alpha} &= -2 \int_0^\infty \lambda e^{-\frac{\sigma^2 \lambda^2 \tau}{2}} \sin(\lambda\alpha) d\lambda \\ &= \left[ \sin(\lambda\alpha) \frac{2e^{-\frac{\sigma^2 \lambda^2 \tau}{2}}}{\sigma^2 \tau} \right]_0^\infty - \frac{2\alpha}{\sigma^2 \tau} \int_0^\infty e^{-\frac{\sigma^2 \lambda^2 \tau}{2}} \cos(\lambda\alpha) d\lambda. \end{aligned} \quad (5.20)$$

식(5.19)과 식(5.20)에 의해서 다음 식이 성립한다.

$$\frac{dI(\alpha)}{d\alpha} = \frac{\alpha}{\sigma^2 \tau} I(\alpha). \quad (5.21)$$

상미분 방정식(5.21)의 해는 다음과 같다.

$$I(\alpha) = I(0) \exp\left(\frac{\alpha^2}{2\sigma^2 \tau}\right). \quad (5.22)$$

또한 다음 식이 성립함을 알 수 있다.

$$I(0) = 2 \int_0^\infty \lambda e^{-\frac{\sigma^2 \lambda^2 \tau}{2}} d\lambda = \sqrt{\frac{2\pi}{\sigma^2 \tau}}. \quad (5.23)$$

식(5.22)과 식(5.23)에 의해서 다음 식이 성립한다.

$$I(\alpha) = \sqrt{\frac{2\pi}{\sigma^2 \tau}} \exp\left(\frac{\alpha^2}{2\sigma^2 \tau}\right). \quad (5.24)$$

식(5.18), 식(5.19)와 식(5.24)에 의해서 다음 식이 성립한다.

$$\begin{aligned} u(x, \tau) &= \int_{-\infty}^\infty g(y) I(x-y) dy \\ &= \int_{-\infty}^\infty g(y) \frac{1}{\sqrt{2\pi\sigma^2\tau}} \exp\left(-\frac{(x-y)^2}{2\sigma^2\tau}\right) dy. \end{aligned} \quad (5.25)$$

여기서 다음의 두 번째 변수변환을 하자.

### 2번째 변수변환

$$v = \frac{y-x}{\sigma\sqrt{\tau}} \rightarrow y = x + \sigma\sqrt{\tau}v \quad (5.26)$$

## 제 1 절 옵션 가격 결정 모형의 BLACK-SCHOLES 편미분방정식 만드는 법73

이제 마지막으로, Black-Scholes 편미분방정식의 해에 대입하면,

$$V(S, t) = e^{-r\tau} u(x, \tau) = SN\left(\frac{x}{\sigma\sqrt{\tau}} + \sigma\sqrt{\tau}\right) - Ee^{-r\tau} N\left(\frac{x}{\sigma\sqrt{\tau}}\right)$$

### Black-Scholes 편미분방정식의 공식

$$V(S, t) = SN\left(\frac{x}{\sigma\sqrt{\tau}} + \sigma\sqrt{\tau}\right) - Ee^{-r\tau} N\left(\frac{x}{\sigma\sqrt{\tau}}\right) \quad (5.29)$$

$$x = \log \frac{S}{E} + \left(r - \frac{\sigma^2}{2}\right)(T - t) \quad (5.29)$$

$$\tau = T - t \quad (5.30)$$

다음의 예제를 풀어보자.

예제 다음 유러피언(European) 콜옵션의 가격을 구하는 MATLAB 코드를 작성하시오.

현재의 주가지수  $S = 240$  포인트

권리행사가격  $E = 250$  포인트

옵션의 기간 = 2개월

주가변동  $\sigma = 38\%$

비위험 이자율  $r = 6\%$

[해답] MATLAB 코드

```
%%%%% eurocall.m %%%%%%
S=240; E=250; t=2/12; sigma=0.38; r=0.06;
d1 = (log(S/E) + (r+ 0.5*sigma^2)*t)/(sigma*sqrt(t));
d2 = d1 -(sigma*sqrt(t));
```

```
set(gca,'fontsize',20)
%%%%%%%
%%%%%
```

위의 코드에 의한 결과로 얻어진 그림 5.1의 우측의 그림은 좌측의 그림을 확대한 것이다.

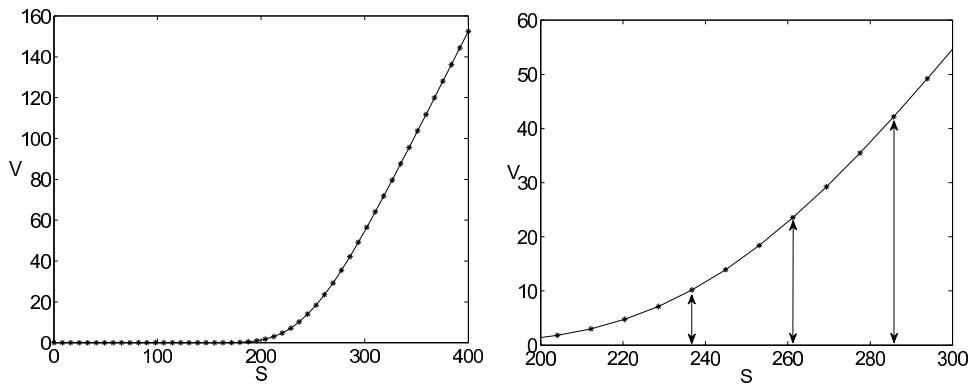


그림 5.1: 기초자산가격( $S$ )에 따른 옵션가격의 변화

### 1.2.2 행사가격( $E$ )

행사가격( $E$ )에 따라 옵션가치의 변화를 다음의 그림 5.2를 통해 확인해 보자. 먼저, `option_ex.m`는 행사가격에 따른 옵션가격을 구하는 MATLAB 코드이다.

```
%%%%%%%
%%%%% option_ex.m %%%%%%
clear all; clc; clf;
S = linspace(0,400,50); t = 2/12; sigma = 0.38; r = 0.06;
E = [200 250 300];
for i = 1:3
d1 = (log(S/E(i)) + (r+ 0.5*sigma^2)*t)/(sigma*sqrt(t));
d2 = d1 -(sigma*sqrt(t));
```

### 1.2.3 이자율( $r$ )

그림 5.3에서 볼 수 있듯이 옵션가격은 이자율( $r$ )에 대한 증가함수로서 이자율( $r$ )이 증가할 수록 옵션가격도 증가하게 된다. option\_r.m는 이자율

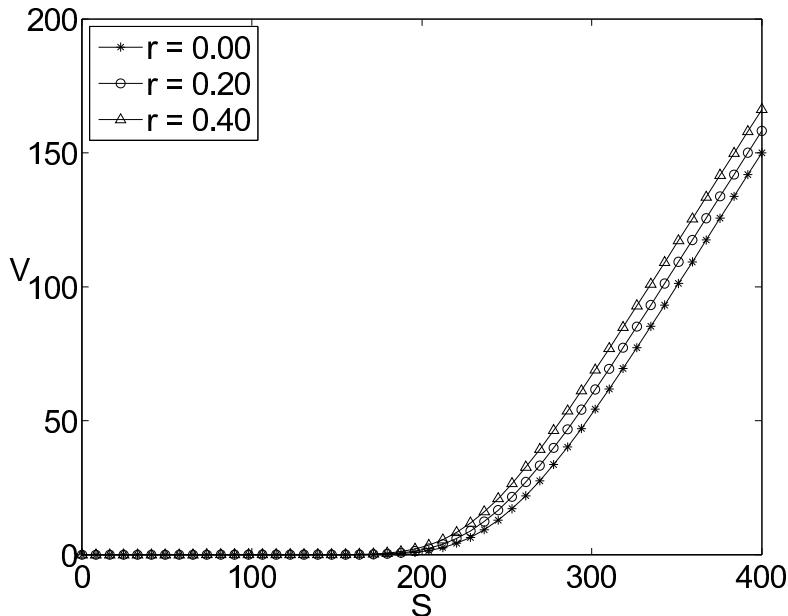


그림 5.3: 이자율( $r$ )에 따른 옵션가격의 변화

에 따른 옵션가격을 구하는 MATLAB 코드로서 그림 5.3을 얻을 수 있다.

```
%%%%% option_r.m %%%%%%
clear all; clc; clf;
S = linspace(0,400,50); E=250; t = 2/12; sigma = 0.38;
r = [0.00 0.20 0.40];
for i = 1:3
d1 = (log(S/E) + (r(i)+ 0.5*sigma^2)*t)/(sigma*sqrt(t));
d2 = d1 -(sigma*sqrt(t));
CP(i,:) = S.* normcdf(d1) - E * exp(-r(i) * t)*normcdf(d2);
end
```

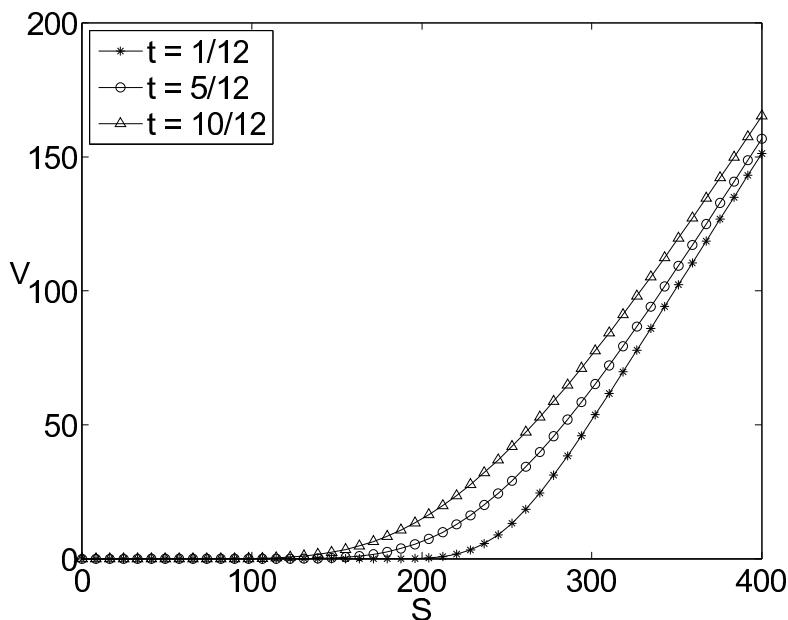


그림 5.4: 잔존기간( $t$ )에 따른 옵션가격의 변화

### 1.2.5 변동성( $\sigma$ )

변동성( $\sigma$ )은 옵션가격에 대한 증가함수이다. 이를 확인하기 위해 MATLAB 코드 `option_t.m`를 실행해보자.

```
%%%%%%%% option_sig.m %%%%%%
clear all; clc; clf;
S = linspace(0,400,50); E=250; r=0.06; t = 2/12;
sigma = [0.10 0.38 0.70];
for i = 1:3
d1 = (log(S/E) + (r+ 0.5*sigma(i)^2*t)/(sigma(i)*sqrt(t));
d2 = d1 -(sigma(i)*sqrt(t));
CP(i,:) = S.* normcdf(d1) - E * exp(-r * t)*normcdf(d2);
end
plot(S,CP(1,:), 'k*-', S, CP(2,:), 'ko-', S, CP(3,:), 'k^-')
```

## 제 6 장

# 변동성 추정

변동성(volatility)은 옵션의 만기까지의 기초자산가격 변화율의 분포의 표준편차라고 정의할 수 있다. 기초자산이 일정기간동안 어느정도 움직이는지를 나타내는 수치로 과거정보를 바탕으로 일정기간 동안의 기초자산가격 변화율의 표준편차는 역사적변동성이라고 하고, 옵션의 시장가격에 내재된 변동성을 내재변동성이라고 정의한다.

## 제 1 절 내재 변동성

Black-Scholes 옵션가격결정모형에서 산출된 이론가격이 시장가격과 같아지도록 하는 변동성을 내재변동성이라 한다. 이러한 내재변동성은 실제 시장에서 거래하는 사람들이 느끼는 체감 변동성이라고 할 수 있다. 옵션의 이론가는 기초자산가격, 행사가격, 잔존기간, 변동성, 이자율로 결정되며, 이중 변동성을 변수로 놓고 역산하여 계산한 값이 내재변동성이다. 이러한 내재변동성은 일반적으로 수치해석적 방법을 적용하여 계산한다. 이 장에서는 콜옵션의 내재변동성을 뉴튼 랩슨법을 이용하여 구해보기로 하자.

여기서  $\sigma_i$ 는 추정변동성,  $V_m$ 은 옵션의 시장가격,  $\frac{\partial V}{\partial \sigma_i} = S\sqrt{T}N'(d_1)$ <sup>1</sup>는 배가(변동성에 따른 옵션가치의 민감도, 자세한 내용은 제 7장의 유한차분법의 Greek참고)를 나타낸다. 다음은 내재변동성을 구하는 간단한 예제이다. 기초자산가격  $S = 100$ , 행사가격  $E = 100$ , 이자율  $r = 0.05$ , 만기시간은  $T = 1.0$ , 그리고 옵션의 시장가격  $V_m = 20$ 으로 가정하고 초기 값은 0.2로 설정하였고, 수치 해석적 방법으로 해를 구할 때 필요한 오차 한계치는  $1.0e - 6$ 로 하였다. `newton_implied_vol.m`는 위의 식을 이용하여 내재변동성을 구한 MATLAB 코드이다.

```
%%%%%%%% Newton_implied_vol.m %%%%%%
clear; clc;
S = 100 ; E = 100; r = 0.05; T = 1.0; Vm = 20;
vol = 0.2; tol = 1.0e-6;

d1 = (log(S/E)+(r+vol^2/2)*T)/(vol*sqrt(T));
d2 = d1-vol*sqrt(T);
price = S*normcdf(d1)-E*exp(-r*T)*normcdf(d2);
vega = S.*sqrt(T).*normpdf(d1);

while abs(price - Vm) > tol
    vol = vol - (price - Vm) / vega;
    d1 = (log(S/E)+(r+vol^2/2)*T)/(vol*sqrt(T));
    d2 = d1-vol*sqrt(T);
    price = S*normcdf(d1)-E*exp(-r*T)*normcdf(d2);
    vega = S.*sqrt(T).*normpdf(d1);
end
```

---

<sup>1</sup>위의 식에서 사용되는  $d_1$ 은 다음과 같다.

$$d_1 = \frac{\ln(S/E) + (r + \sigma^2/2)T}{\sigma\sqrt{T}}$$

## 제 7 장

# 유한 차분법

## (Finite Difference Method)

### 제 1 절 개요

유한차분법은 미분방정식 (differential equation)을 차분방정식 (difference equation)으로 이산화 시켜서 수치적인 해를 구하는 방법이다. 이 장에서 는 유한차분법을 사용하여 열방정식 (heat equation)과 블랙 솔즈 편미분 방정식의 근사해를 구할 것이다. 먼저 Taylor의 정리<sup>1</sup>를 바탕으로 하고 있는 유한차분법의 기본 원리를 살펴보자. Taylor의 정리를 이용하면 함수  $u(x+h, t)$ 는 다음과 같이  $(x, t)$  에서의  $u$  함수값과 미분값들의 무한급수로 나타낼 수 있다.

$$u(x+h, t) = u(x, t) + u_x(x, t)h + \frac{u_{xx}(x, t)}{2}h^2 + \frac{u_{xxx}(x, t)}{3!}h^3 + \dots \quad (7.1)$$

$u_x(x, t)$ 에 대해서 정리하면, 1차미분에 대한 차분식을 얻는다.

$$u_x(x, t) = \frac{u(x+h, t) - u(x, t)}{h} + \mathcal{O}(h). \quad (7.2)$$

---

<sup>1</sup>Taylor 정리: 함수  $f(x)$ 가  $x = x_0$ 에서  $n$ 번 미분가능하다고 하자.

$$p_n(x) = f(x_0) + f'(x_0)(x - x_0) + \frac{f''(x_0)}{2!}(x - x_0)^2 + \dots + \frac{f^{(n)}(x_0)}{n!}(x - x_0)^n$$

을  $x = x_0$ 에서  $f(x)$ 의  $n$ 번째 Taylor 다항식이라 한다.

## 제 2 절 열 방정식에 대한 유한 차분법

유한차분법을 이용하여 열방정식을 풀어보기로 하자. 열방정식은 식(7.8)과 같은 편미분 방정식이다.

$$u_t(x, t) = u_{xx}(x, t), \quad 0 < x < 1, \quad t > 0. \quad (7.8)$$

이 때 경계조건은  $u(0, t) = u(1, t) = 0$  ( $t > 0$ )이고 초기조건은  $u(x, 0) = \sin(\pi x)$  ( $0 \leq x \leq 1$ )을 만족한다. 해석해는  $u(x, t) = \sin(\pi x)e^{-\pi^2 t}$ 이며 그림7.2와 그림7.3처럼 그래프로 나타낼 수 있다. 이 방정식에 대한 근사해를 명시적, 함축적, 그리고 크랭크-니콜슨 유한차분법을 이용하여 구해보자.

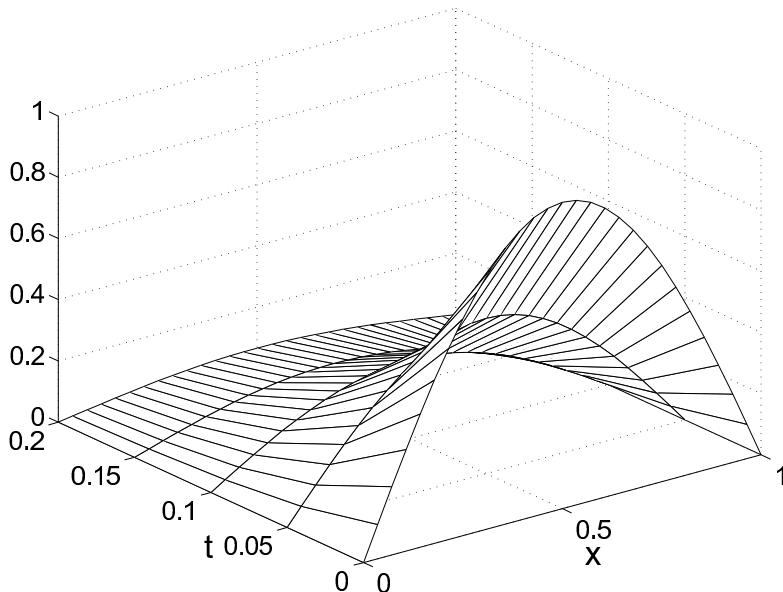


그림 7.2: 열방정식의 해석해

### 2.1 명시적 (Explicit) 유한 차분법

먼저 정수  $N_x > 0$ 을 선택하고  $h = 1/(N_x - 1)$ 이라 정의하면, 식(7.3)와 (7.7)을 이용하여, 열방정식(7.8)에 대해 다음과 같이 유한차분법을 적용할

을 명시적이라 부르는 이유이다. `heatex.m`은  $\alpha = 0.45$ ,  $N_x = 30$ 일 때, 시간에 따른 열방정식의 해를 나타낸 MATLAB 코드이다.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clf; clear; clc; alpha=0.45; Nx=30; x=linspace(0,1,Nx);
h=x(2)-x(1); k=alpha*h^2; T=0.125; Nt=round(T/k);
u(1:Nx,1:Nt+1)=0; u(:,1)=sin(pi*x); exu=u;
for n=1:Nt
    for i=2:Nx-1
        u(i,n+1) = u(i,n)+alpha*(u(i-1,n)-2*u(i,n)+u(i+1,n));
        exu(i,n+1) = sin(pi*x(i))*exp(-pi^2*(k*n));
    end
end
plot(x,u(:,1), 'k*',x,u(:,50), 'kd',x,u(:,100), 'ks',...
      x,u(:,Nt+1), 'ko');
hold
plot(x,exu(:,1), 'k',x,exu(:,50), 'k',x,exu(:,100), 'k',...
      x,exu(:,Nt+1), 'k')
legend('initial','n=50','n=100','n=192','exact solution')
xlabel('x', 'FontSize',20); ylabel('u(x,t)', 'FontSize',20)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

### 2.1.1 명시적방법의 안정성 문제 - 폰 노이만 (von Neumann) 방법

폰노이만 방법은 초기조건을 유한개의 푸리에 급수 (finite Fourier series)로 나타낸 다음, 함수의 성장을 고려하는 것이다. 푸리에 급수는 사인과 코사인의 조합으로 표현할 수도 있지만 복소지수함수로 나타내면 계산을 간단하게 할 수 있다. 시간이 지남에 따라서

$$u_k^n = e^{i\beta kh} \xi^n \quad (7.11)$$

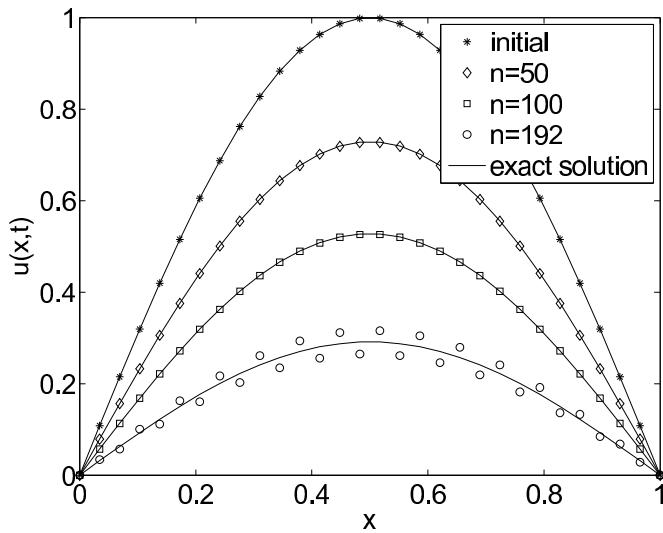
그림 7.5:  $\alpha = 0.55$ 인 불안정한 상태

그림 7.4에서  $\alpha = 0.45$ 일 때 계산된 값은 정확한 해에 가까워지지만, 그림 7.5에서 보듯이  $\alpha = 0.55$ 일 때 계산된 값은 정확한 해로 수렴하지 않는다. 즉  $\alpha$ 의 값이 커짐에 따라 구한 해가 불안정할 수 있다는 점이 명시적 유한 차분법의 단점이다.

## 2.2 함축적 (Implicit) 유한 차분법

명시적 유한차분법의 안정조건인  $0 < \alpha \leq \frac{1}{2}$ 의 제약을 피하기 위해 함축적 유한 차분법을 이용한다. 함축적 방법은 시간간격을 작게 취하지 않고도 많은 수의 격자점을 이용할 수 있다. 다만 함축적 방법은 유한차분 연립 방정식의 해를 구해야 한다. 보통 함축적 유한차분법이라고 알려진 완전 함축적 유한차분법은  $u_t$ 에 대한 후방 유한차분근사와  $u_{xx}$ 에 대한 중앙차분 근사를 이용한다. 따라서 다음과 같은 함축적 유한차분 방정식을 이끌어 낼 수 있다.

$$\frac{u_i^{n+1} - u_i^n}{k} = \frac{u_{i-1}^{n+1} - 2u_i^{n+1} + u_{i+1}^{n+1}}{h^2}.$$

이다. 그러므로  $u^n$ 과 경계조건에 의해 구해지는  $b^n$ 에 의해  $u^{n+1}$ 를 찾을 수 있다. 초기조건은  $u^1$ 에 의해 결정되므로, 각각의  $u^{n+1}$ 을 순차적으로 구할 수 있다.

치하게 되고  $d_2$ 와  $b_2$ 의 값도 다음과 같이 변하게 된다.

$$d_2 = d_2 - \frac{a_1}{d_1}c_1, \quad b_2 = b_2 - \frac{a_1}{d_1}b_1$$

하지만  $c_2$ 는 변하지 않는다.

$$\left( \begin{array}{ccc|c} d_1 & c_1 & & x_1 \\ 0 & d_2 - \frac{a_1 c_1}{d_1} & c_2 & x_2 \\ a_2 & d_3 & c_3 & x_3 \\ \ddots & \ddots & \ddots & \vdots \\ a_{i-1} & d_i & c_i & x_i \\ \ddots & \ddots & \ddots & \vdots \\ a_{N_x-2} & d_{N_x-1} & c_{N_x-1} & x_{N_x-1} \\ a_{N_x-1} & d_{N_x} & & x_{N_x} \end{array} \right) = \left( \begin{array}{c} b_1 \\ b_2 - \frac{a_1 b_1}{d_1} \\ b_3 \\ \vdots \\ b_i \\ \vdots \\ b_{N_x-1} \\ b_{N_x} \end{array} \right).$$

- 과정을 반복하여 적용하면 각각의 단계에서  $d_i$ 와  $b_i$ 들은 다음과 같
- 바뀌게 된다:

$$d_i = d_i - \frac{a_{i-1}}{d_{i-1}}c_{i-1}, \quad b_i = b_i - \frac{a_{i-1}}{d_{i-1}}b_{i-1} \quad (2 \leq i \leq N_x).$$

```

for n=1:Nt
    d=dd;
    for i=1:Nx-2
        b(i)=u(i+1,n);
    end
    for i=2:Nx-2
        xmult= a(i-1)/d(i-1);
        d(i) = d(i) - xmult*c(i-1); b(i) = b(i) -
        xmult*b(i-1);
    end
    u(Nx-1,n+1) = b(Nx-2)/d(Nx-2);
    for i = Nx-3:-1:1
        u(i+1,n+1) = (b(i) - c(i)*u(i+2,n+1))/d(i);
    end
end
plot(x,u,'ko-')
xlabel('x','FontSize',20); ylabel('u(x,t)','FontSize',20);
%%%%%%%%%%%%%%%

```

위의 코드 `heatim.m`를 실행하면 다음의 결과를 얻을 수 있다.

### 2.2.2 함축적 방법의 안정성 문제 - 폰 노이만 방법

명시적 방법의 안정성 문제와 같이 시간이 지남에 따라서

$$u_k^n = e^{i\beta kh} \xi^n \quad (7.19)$$

이 어떻게 성장하는가 보자. 식 (7.19)을 방정식 (7.15)에 대입을 하면 다음을 얻는다.

$$\begin{aligned}
 -\alpha e^{i\beta(k-1)h} \xi^{n+1} + (1 + 2\alpha) e^{i\beta kh} \xi^{n+1} - \alpha e^{i\beta(k+1)h} \xi^{n+1} &= e^{i\beta kh} \xi^n, \\
 -\alpha e^{-i\beta h} \xi + (1 + 2\alpha) \xi - \alpha e^{i\beta h} \xi &= 1, \\
 (2\alpha(1 - \cos(\beta h)) + 1) \xi &= 1.
 \end{aligned}$$

서 2계 편미분 근사값을 평균해서 구한다.

$$\begin{aligned} u_{xx}(x_i, t^{n+1/2}) &= \frac{1}{2} (u_{xx}(x_i, t^n) + u_{xx}(x_i, t^{n+1})) + O(h^2) \\ &= \frac{1}{2} \left( \frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{h^2} + \frac{u_{i+1}^{n+1} - 2u_i^{n+1} + u_{i-1}^{n+1}}{h^2} \right) \\ &\quad + O(h^2). \end{aligned} \quad (7.22)$$

두 근사식 (7.21)와 (7.22)의 절단오차는 각각  $\mathcal{O}(k^2)$ 과  $\mathcal{O}(h^2)$ 으로 근사식의 정확도가 높기 때문에 많은 계산을 하지 않아도 수치분석에서 만족스러운 해를 얻을 수 있다. 식(7.21)와 (7.22)의 우변을 같게 하면 다음과 같은 크랭크 니콜슨식을 얻을 수 있다.

$$-\alpha u_{i-1}^{n+1} + 2(1 + \alpha)u_i^{n+1} - \alpha u_{i+1}^{n+1} = \alpha u_{i-1}^n + 2(1 - \alpha)u_i^n + \alpha u_{i+1}^n, \quad (7.23)$$

여기서  $\alpha = \frac{k}{h^2}$ 이다. 식 (7.23)를 행렬 형태로 표현하면 다음과 같다.

$$\begin{aligned} & \begin{pmatrix} 2(1 + \alpha) & -\alpha & & & & \\ -\alpha & 2(1 + \alpha) & -\alpha & & & \\ & -\alpha & 2(1 + \alpha) & -\alpha & & \\ & & \ddots & \ddots & \ddots & \\ & & & & 2(1 + \alpha) & -\alpha \\ & & & & -\alpha & 2(1 + \alpha) \end{pmatrix} \begin{pmatrix} u_2^{n+1} \\ u_3^{n+1} \\ u_4^{n+1} \\ \vdots \\ u_{Nx-2}^{n+1} \\ u_{Nx-1}^{n+1} \end{pmatrix} \\ &= \begin{pmatrix} \alpha u_1^{n+1} + 2(1 - \alpha)u_2^n + \alpha u_3^n \\ \alpha u_2^n + 2(1 - \alpha)u_3^n + \alpha u_4^n \\ \alpha u_3^n + 2(1 - \alpha)u_4^n + \alpha u_5^n \\ \vdots \\ 2(1 - \alpha)u_{Nx-2}^n + \alpha u_{Nx-1}^n + \alpha u_{Nx}^{n+1} \\ \alpha u_{Nx-2}^n + 2(1 - \alpha)u_{Nx-1}^n + \alpha u_{Nx}^{n+1} \end{pmatrix} = \begin{pmatrix} b_2^n \\ b_3^n \\ b_4^n \\ \vdots \\ b_{Nx-2}^n \\ b_{Nx-1}^n \end{pmatrix}. \end{aligned} \quad (7.24)$$

위의 코드 `heatCN.m`를 실행하면 다음의 결과를 얻을 수 있다.

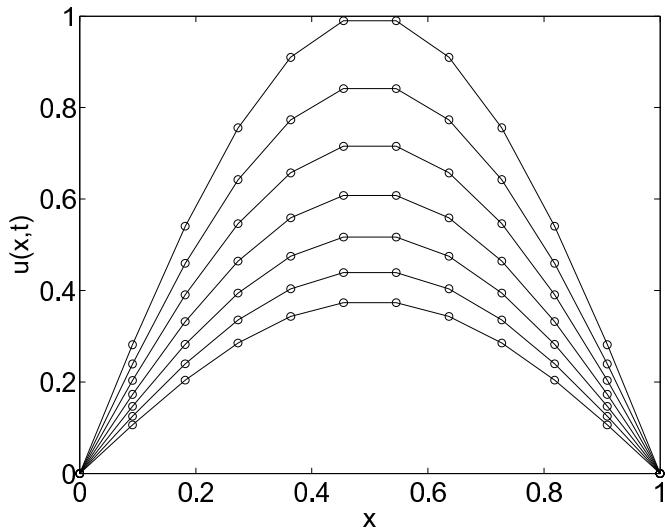


그림 7.7: 크랭크 니콜슨을 이용한 열방정식  $\alpha = 2$ 인 안정한 상태

### 2.3.1 크랭크 니콜슨 방법의 안정성 문제 - 폰 노이만 방법

시간이 지남에 따라서

$$u_k^n = e^{i\beta kh} \xi^n \quad (7.25)$$

이 어떻게 성장하는가 보자. 식 (7.25)을 방정식 (7.23)에 대입을 하면 다음을 얻는다.

$$-\alpha u_{i-1}^{n+1} + 2(1 + \alpha)u_i^{n+1} - \alpha u_{i+1}^{n+1} = \alpha u_{i-1}^n + 2(1 - \alpha)u_i^n + \alpha u_{i+1}^n, \quad (7.26)$$

여기서  $\alpha = \frac{k}{h^2}$ 이다.

$$\begin{aligned} -\alpha e^{i\beta(k-1)h} \xi^{n+1} + (1 + 2\alpha) e^{i\beta kh} \xi^{n+1} - \alpha e^{i\beta(k+1)h} \xi^{n+1} &= e^{i\beta kh} \xi^n, \\ -\alpha e^{-i\beta h} \xi + (1 + 2\alpha) \xi - \alpha e^{i\beta h} \xi &= 1, \\ (2\alpha(1 - \cos(\beta h)) + 1)\xi &= 1. \end{aligned}$$

time accurate)하고  $m$ 차 공간정확 ( $m$ th order space accurate)하다고 한다. 식(7.33)으로부터 명시적 유한차분법은 1차 시간 정확하고 2차 공간 정확함을 알 수 있다.

#### 2.4.1 명시적 유한차분법

열방정식의 명시적 유한 차분법의 수렴성을 알아보기 위해 다음의 테스트를 수행해보자.

```
%%%%%%%% heatex_convergence_test.m %%%%%%
clear; clc; T=0.1; alpha=0.1;
for iter=1:5
    N=10*2^(iter-1)+1; x=linspace(0,1,N); h=x(2)-x(1);
    k=alpha*h^2; Nt=round(T/k); u(1:N,1:Nt+1)=0;
    u(:,1)=sin(pi*x); exact=u(:,1)*exp(-pi^2*T);
    for n=1:Nt
        for i=2:N-1
            u(i,n+1)=alpha*u(i-1,n)+...
                (1-2*alpha)*u(i,n)+alpha*u(i+1,n);
        end
    end
    hh(iter)=h; tt(iter)=k;
    err(iter) = max(abs(u(:,Nt+1) - exact));
end
Order=[log(err(1)/err(2))/log(2) ...
    log(err(2)/err(3))/log(2) ...
    log(err(3)/err(4))/log(2) ...
    log(err(4)/err(5))/log(2)]';
fprintf('-----\n')
fprintf('      h          dt      max error      order\n')
fprintf('-----\n')
```

```

N=10*2^(iter-1)+1; x=linspace(0,1,N); h=x(2)-x(1);
k=alpha*h^2; Nt=round(T/k); u(1:N,1:Nt+1)=0;
u(:,1)=sin(pi*x); exact=u(:,1)*exp(-pi^2*T);
for i=1:N-2
    dd(i)= 1 + 2*alpha; c(i)= - alpha; a(i)= - alpha;
end
for n=1:Nt
    d=dd;
    for i=1:N-2
        b(i)=u(i+1,n);
    end
    for i=2:N-2
        xmult= a(i-1)/d(i-1);
        d(i) = d(i) - xmult*c(i-1);
        b(i) = b(i) - xmult*b(i-1);
    end
    u(N-1,n+1) = b(N-2)/d(N-2);
    for i = N-3:-1:1
        u(i+1,n+1) = (b(i) - c(i)*u(i+2,n+1))/d(i);
    end
end
hh(iter)=h; tt(iter)=k;
err(iter) = max(abs(u(:,Nt+1) - exact));
end
Order=[log(err(1)/err(2))/log(2) ...
        log(err(2)/err(3))/log(2) ...
        log(err(3)/err(4))/log(2) ...
        log(err(4)/err(5))/log(2)]';
fprintf('-----\n')

```

```
%%%%% heatcn_convergence_test.m %%%%%%
clear; clc; T=0.1; alpha=0.1;
for iter=1:5
    N=10*2^(iter-1)+1; x=linspace(0,1,N); h=x(2)-x(1);
    k=alpha*(h^2); Nt=round(T/k); u(1:N,1:Nt+1)=0;
    u(:,1)=sin(pi*x); exact=u(:,1)*exp(-pi^2*T);
    for i=1:N-2
        dd(i)= 2*(1+alpha); c(i)= - alpha; a(i)= - alpha;
    end
    for n=1:Nt
        d=dd;
        for i=1:N-2
            b(i)=alpha*u(i,n)+...
                2*(1-alpha)*u(i+1,n)+alpha*u(i+2,n);
        end
        for i = 2:N-2
            xmult=a(i-1)/d(i-1);
            d(i)=d(i)-xmult*c(i-1); b(i)=b(i)-xmult*b(i-1);
        end
        u(N-1,n+1) = b(N-2)/d(N-2);
        for i = N-3:-1:1
            u(i+1,n+1) = (b(i) - c(i)*u(i+2,n+1))/d(i);
        end
    end
    hh(iter)=h; tt(iter)=k;
    err(iter) = max(abs(u(:,Nt+1) - exact));
end
Order=[log(err(1)/err(2))/log(2) ...
        log(err(2)/err(3))/log(2) ...
```

### 제 3 절 Black-Scholes 편미분방정식에 대한 유한 차분법

유리피언 콜 옵션의 값을 구하기 위해서 Black-Scholes 편미분방정식을 유한차분법으로 풀어서 구한다. 편미분방정식은 Dirichlet 경계조건을 갖는 포물선형 편미분방정식이다. 특히 초기조건보다 만기시의 조건이 주어진다.  $\tau = T - t$ 를 잔존기간으로 놓음으로써, 더 자연스러운 시간의 방정식으로 바꿀 수 있다. 그러면 편미분방정식은 다음과 같이 정리된다.

$$u_\tau = \frac{1}{2}\sigma^2 x^2 u_{xx} + rxu_x - ru.$$

이 때, 정의역은  $x \geq 0$ 이고, 시간의 범위는  $0 \leq \tau \leq T$ , 초기값은  $u(x, 0) = \max(x - E, 0)$ <sup>2</sup>이고, 경계조건은  $u(0, \tau) = 0$ , 값이 큰  $x$ 에 대해서  $u(x, \tau) \approx x - Ee^{-r\tau}$ 를 갖는다.  $x$ 를  $0 \leq x \leq L$ 의 범위로 두고  $h = L/(N_x - 1)$ 과  $k = T/N_t$ 의 간격을 갖는 유한 차분 격자를 사용함으로써, 이산 해  $u_i^n \approx u((i-1)h, (n-1)k) = u(x_i, t^n)$ 을 계산할 수 있다. 모든  $1 \leq n \leq N_t$ 에서 초기 데이터에 의해 지정된 값  $u_i^1 = \max(x_i - E, 0)$  for  $1 \leq i \leq N_x$ , 그리고 경계조건에 의해 지정된 경계값  $u_1^n = 0$ 과  $u_{N_x}^n = L - Ee^{-rt^n}$ 을 갖게 된다.

#### 3.1 명시적 방법에 의한 옵션 가격 결정

시간 미분에 대해서 전방 차분, 그리고 공간 미분에 대해서 중앙 차분을 사용함으로써 다음과 같이 명시적 방법을 적용할 수 있다.

$$\frac{u_i^{n+1} - u_i^n}{k} = \frac{1}{2}\sigma^2 x_i^2 \frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{h^2} + rx_i \frac{u_{i+1}^n - u_{i-1}^n}{2h} - ru_i^n.$$

$u_i^{n+1}$ 에 대해서 정리하면 다음과 같은 식을 얻는다.

$$u_i^{n+1} = u_i^n + k \left( \frac{1}{2}\sigma^2 x_i^2 \frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{h^2} + rx_i \frac{u_{i+1}^n - u_{i-1}^n}{2h} - ru_i^n \right)$$

for  $2 \leq i \leq N_x - 1$ .

---

<sup>2</sup>초기값  $u(x, 0) = \max(x - E, 0)$ 은  $x < E$ 일 때  $u(x, 0) = 0$ 을  $x \geq E$ 일 때  $u(x, 0) = x - E$ 을 의미한다.

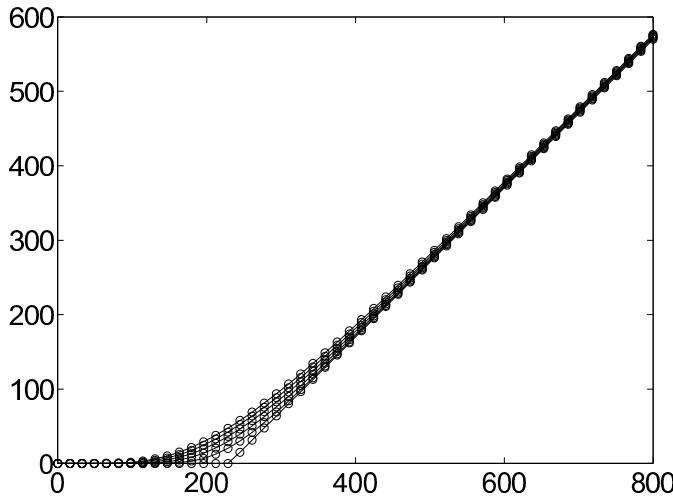


그림 7.8: 명시적 방법에 의한 블랙숄츠 방정식의 수치해

### 3.2 함축적 방법에 의한 옵션 가격 결정

시간에 대한 전방 차분을 이용한 명시적 방법에서 일어날 수 있는 불안정성 문제를 해결하기 위해서 후방차분을 이용하여 함축적 방법을 적용하면 다음의 식을 얻게 된다.

$$\frac{u_i^{n+1} - u_i^n}{k} = \frac{1}{2} \sigma^2 x_i^2 \frac{u_{i+1}^{n+1} - 2u_i^{n+1} + u_{i-1}^{n+1}}{h^2} + rx_i \frac{u_{i+1}^{n+1} - u_{i-1}^{n+1}}{2h} - ru_i^{n+1}. \quad (7.29)$$

이러한 함축적 방법은 시간의 크기에 영향을 받지 않을 뿐만 아니라 수치 계산의 정확도를 높이는 장점이 있다. 이제 위의 식 (7.29)을 정리하면

$$\alpha_i u_{i-1}^{n+1} + \beta_i u_i^{n+1} + \gamma_i u_{i+1}^{n+1} = \frac{u_i^n}{k}, \quad (7.30)$$

여기서

$$\alpha_i = \frac{rx_i}{2h} - \frac{\sigma^2 x_i^2}{2h^2}, \quad \beta_i = \frac{1}{k} + \frac{\sigma^2 x_i^2}{h^2} + r, \quad \gamma_i = -\frac{rx_i}{2h} - \frac{\sigma^2 x_i}{h^2}$$

이다.

```

b(i) = b(i) - xmult*b(i-1);
end
u(N+1,n+1) = b(N)/d(N);
for i = N-1:-1:1
    u(i+1,n+1) = (b(i) - c(i)*u(i+2,n+1))/d(i);
end
plot(x,u(:,1:20:Nt+1), 'ko-')
%%%%%%%%%%%%%

```

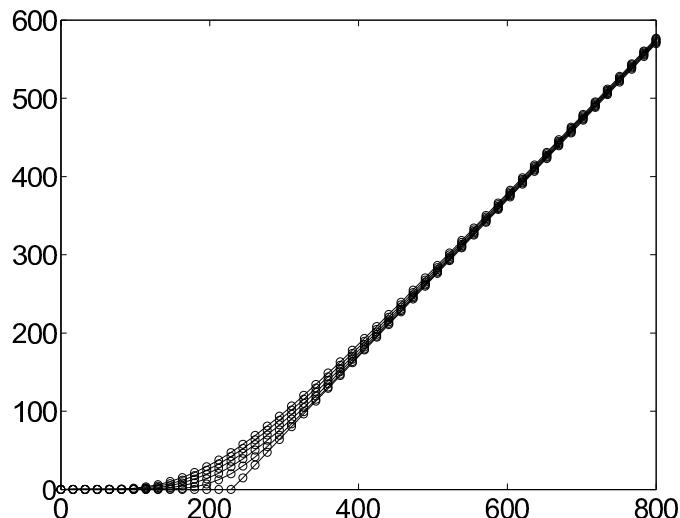


그림 7.9: 함축적 방법에 의한 옵션 가격 결정

### 3.3 크랭크 니콜슨 방법에 의한 옵션 가격 결정

크랭크 니콜스방법은 명시적방법과 함축적 방법을 조합하여 정확도를 향상시킨 방법이다. 이 아이디어를 Black-Scholes방정식에 적용하면 다음과 같은 방정식을 얻는다.

```

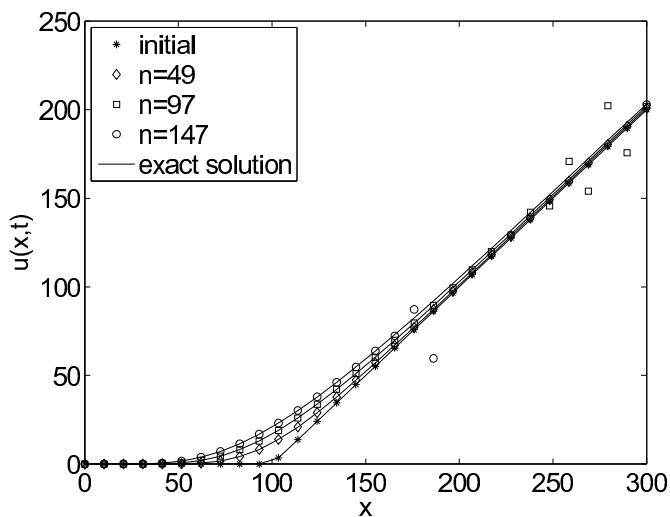
    end
end
N = Nx-2;
for i=1:N
    dd(i)=1/k+(sigma*i)^2/2+r;
    c(i)=-r*i/4 - ((sigma*i)^2)/4;
    a(i)=r*(i+1)/4-((sigma*(i+1))^2)/4;
end
for n=1:Nt
    d=dd;
    for i=1:N
        b(i) = u(i+1,n)/k + (sigma*i)^2*(u(i+2,n)...
            -2*u(i+1,n)+u(i,n))/4 ...
            + r*i*(u(i+2,n)-u(i,n))/4 - r*u(i+1,n)/2;
    end
    u(Nx,n+1)= L - E*exp(-r*k*n);
    b(N) = b(N) - c(N)*u(Nx,n+1);
    for i = 2:N
        xmult= a(i-1)/d(i-1); d(i) = d(i) - xmult*c(i-1);
        b(i) = b(i) - xmult*b(i-1);
    end
    u(N+1,n+1) = b(N)/d(N);
    for i = N-1:-1:1
        u(i+1,n+1) = (b(i) - c(i)*u(i+2,n+1))/d(i);
    end
end
plot(x,u(:,1:20:Nt+1), 'ko-')
%%%%%%%%%%%%%

```

```

u(i,1)=x(i)-E;
end
end
for n=2:Nt+1
    u(Nx,n)=L-E*exp(-r*k*(n-1));
end
exu=u;
for n=1:Nt
    for i=2:Nx-1
        u(i,n+1)=u(i,n) + k*((1/2)*(sigma^2)*((i-1)*h)^2*...
            ((u(i+1,n)-2*u(i,n)+u(i-1,n))/(h^2)) +...
            r*(i-1)*h*((u(i+1,n)-u(i-1,n))/(2*h))-r*u(i,n));
    end
    for i=1:Nx
        d1(i)=(log(x(i)/E)+(r+sigma^2/2)*k*n)...
            /(sigma*sqrt(k*n));
        d2(i)=d1(i)-sigma*sqrt(k*n);
        exu(i,n+1)=x(i)*normcdf(d1(i))...
            -E*exp(-r*k*n)*normcdf(d2(i));
    end
end
plot(x,u(:,1),'k*',x,u(:,round(Nt/3)), ...
    'kd',x,u(:,round(2*Nt/3)), 'ks',x,u(:,Nt+1), 'ko'); hold
plot(x,exu(:,1), 'k',x,exu(:,round(Nt/3)),...
    'k',x,exu(:,round(2*Nt/3)), 'k',x,exu(:,Nt+1), 'k')
legend('initial','n=78','n=156','n=234','exact solution',2)
xlabel('x','FontSize',20); ylabel('u(x,t)','FontSize',20)
axis([0 L 0 L-0.5*E])
%%%%%%%%%%%%%%%

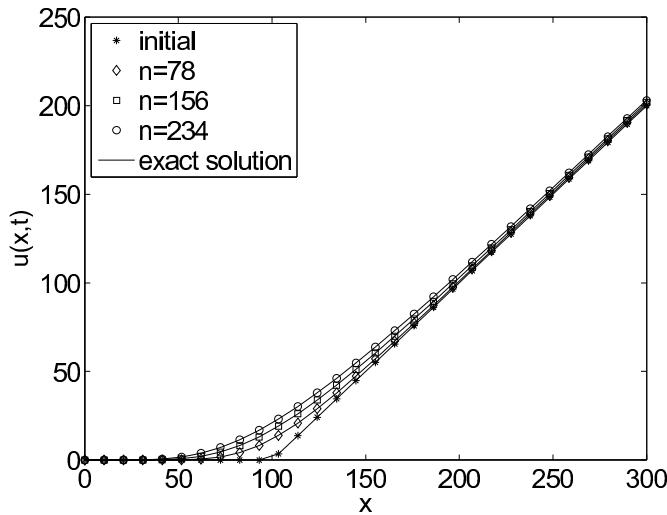
```

그림 7.12:  $\Delta t = 0.0068$ 인 불안정한 상태

```

end
exu=u;
for i=1:N
    dd(i)=1/k+(sigma*i)^2+r; c(i)=-r*i/2-((sigma*i)^2)/2;
    a(i)=r*(i+1)/2-((sigma*(i+1))^2)/2;
end
for n=1:Nt
    d=dd;
    for i=1:N-1
        b(i)=u(i+1,n)/k;
    end
    b(N)=u(N+1,n)/k - c(N)*u(Nx,n+1);
    for i = 2:N
        xmult= a(i-1)/d(i-1);
        d(i) = d(i) - xmult*c(i-1);
        b(i) = b(i) - xmult*b(i-1);
    end
end

```

그림 7.13:  $\Delta t = 0.0043$ 인 안정한 상태

### 3.5 수렴성 테스트

$u(x_i, t^n)$ 은 Black-Scholes 방정식의 정확한 해를 나타낸다. 다음은 정확한 해를 수치기법에 대입함으로써 명시적 유한차분법의 국소절단오차를 찾는 과정이다. 노드  $(x_i, t^n)$ 에서 국소절단오차는 다음과 같이 구한다.

$$\begin{aligned} T(x_i, t^n) &= \frac{u(x_i, t^{n+1}) - u(x_i, t^n)}{k} \\ &\quad - \frac{\sigma^2 x_i^2}{2} \frac{u(x_{i+1}, t^n) - 2u(x_i, t^n) + u(x_{i-1}, t^n)}{h^2} \\ &\quad - rx_i \frac{u(x_{i+1}, t^n) - u(x_{i-1}, t^n)}{2h} + ru(x_i, t^n). \end{aligned}$$

이제 노드  $(x_i, t^n)$ 에서 테일러전개를 하면 각각의 항을 다음과 같이 나타낼 수 있다.

$$\begin{aligned} T(x_i, t^n) &= u_t(x_i, t^n) + \frac{k}{2} u_{tt}(x_i, t^n) + \mathcal{O}(k^2) \\ &\quad - \frac{\sigma^2 x_i^2}{2} \left[ u_{xx}(x_i, t^n) + \frac{h^2}{12} u_{xxxx}(x_i, t^n) + \mathcal{O}(h^4) \right] \\ &\quad - rx_i \left[ u_x(x_i, t^n) + \frac{h^2}{3} u_{xxx}(x_i, t^n) + \mathcal{O}(h^4) \right] + ru(x_i, t^n). \end{aligned}$$

```
%%%%%%%
%BSex_convergence_test.m %
clear; E=100; sigma=0.5; r=0.03; L=400; T=0.1;
for iter=1:5
    N = 16*(2^iter);
    x=linspace(0,L,N); h=x(2)-x(1); k=h/50000;
    Nt=round(T/k); u(1:N,1:Nt+1)=0;
    for i=1:N
        if x(i)<= E
            u(i,1)=0;
        else
            u(i,1)=x(i)-E;
        end
    end
    for n=2:Nt+1
        u(N,n)=L-E*exp(-r*k*(n-1));
    end
    for n=1:Nt
        for i=2:N-1
            u(i,n+1)=u(i,n) + k*((1/2)*(sigma^2)*((i-1)*h)^2*...
                ((u(i+1,n)-2*u(i,n)+u(i-1,n))/(h^2)) +...
                r*(i-1)*h*((u(i+1,n)-u(i-1,n))/(2*h))-r*u(i,n));
        end
    end
    for i=1:N
        d1(i)=(log(x(i)/E)+(r+sigma^2/2)*k*n)...
            /(sigma*sqrt(k*n));
        d2(i)=d1(i)-sigma*sqrt(k*n);
        exact(i)=x(i)*normcdf(d1(i))...
            -E*exp(-r*k*n)*normcdf(d2(i));
    end
end
```

위 결과로 부터 Black-Scholes 방정식에 대한 명시적 유한차분법의 수렴도가 2차임을 알수가 있다. 그러나 안정성 문제로 인해서 함축적 방법이나 크랭크-니콜슨 방법에 비해 100배 정도 작은 시간스텝을 사용해야만 했다. 수렴도가 2차가 나온 것은 시간 스텝에 대한 오차가 무시할 정도로 작고 공간에 대해서 오차가 나온 결과이다.

### 3.5.2 함축적 유한차분법

유러피언 콜옵션방정식의 함축적 유한 차분법의 수렴성을 알아보기 위해 다음의 테스트를 수행해보자. 위험이자율이  $r = 0.03$ , 변동성이  $\sigma = 0.5$ , 현재시점이  $t = 0$ , 만기 시점이  $T = 0.1$ , 그리고 행사가격이  $E = 100$ 인 유럽형 콜옵션의 가격을 구하는 MATLAB 코드이다. 이 때, 초기 조건은  $u(x, 0) = \max(0, x - E)$ ,  $T = 0.1$ ,  $L = 400$ ,  $h = 1/N$ ,  $\Delta t = h/500$ 을 이용하였다. MATLAB 코드 BSex\_convergence\_test.m을 실행하면 다음의 결과를 얻을 수 있다.

```
%%%%%%%
% BSim_convergence_test.m %
%%%%%%%
clear; E=100; sigma=0.5; r=0.03; L=400; T=0.1;
for iter=1:5
    Nx = 16*(2^iter);
    x=linspace(0,L,Nx); h=x(2)-x(1); k=h/500;
    Nt=round(T/k); u(1:Nx,1:Nt+1)=0; N=Nx-2;
    N=Nx-2;
    u(:,1)= max(0,x-E);
    for n=2:Nt+1
        u(Nx,n)=L-E*exp(-r*k*(n-1));
    end
    for i=1:N
        dd(i)=1/k+(sigma*i)^2+r; c(i)=-r*i/2-((sigma*i)^2)/2;
        a(i)=r*(i+1)/2-((sigma*(i+1))^2)/2;
    end
end
```

```

printf('      h          dt      l2 error      order \n')
printf('-----\n')
printf('%.5f    %.6f    %.6f    \n', hh(1), tt(1), err(1))
for iter = 2:5
    printf('%.5f    %.6f    %.6f    %.6f \n', hh(iter), ...
            tt(iter), err(iter), Order(iter-1))
end
printf('-----\n')
%%%%%%%%%%%%%

```

>> BSim\_convergence\_test

h	dt	l2 error	order
<hr/>			
12.90323	0.025806	0.072626	
6.34921	0.012698	0.029825	1.283952
3.14961	0.006299	0.013253	1.170222
1.56863	0.003137	0.006204	1.095123
0.78278	0.001566	0.002995	1.050525
<hr/>			

위 결과로 부터 Black-Scholes 방정식에 대한 함축적 유한차분법의 수렴도가 1차임을 알수가 있다.

### 3.5.3 크랭크 니콜슨 유한차분법

유리피언 콜옵션방정식의 크랭크 니콜슨 유한 차분법의 수렴성을 알아보기 위해 다음의 테스트를 수행해보자. 무위험이자율이  $r = 0.03$ , 변동성이  $\sigma = 0.5$ , 현재시점이  $t = 0$ , 만기 시점이  $T = 0.1$ , 그리고 행사가격이  $E = 100$ 인 유럽형 콜옵션의 가격을 구하는 MATLAB 코드이다. 이 때, 초기 조건은  $u(x, 0) = \max(0, x - E)$ ,  $T = 0.1$ ,  $L = 400$ ,  $h = 1/N$ ,

```

for i = N-1:-1:1
    u(i+1,n+1) = (b(i) - c(i)*u(i+2,n+1))/d(i);
end
for i=1:Nx
    d1(i)=(log(x(i)/E)+(r+sigma^2/2)*k*n)...
        /(sigma*sqrt(k*n));
    d2(i)=d1(i)-sigma*sqrt(k*n);
    exact(i)=x(i)*normcdf(d1(i))...
        -E*exp(-r*k*n)*normcdf(d2(i));
end
hh(iter)=h; tt(iter)=k; F = u(:,Nt+1) - exact';
err(iter) = sqrt(sum(F.^2)/Nx);
end
Order=[log(err(1)/err(2))/log(2) ...
    log(err(2)/err(3))/log(2) ...
    log(err(3)/err(4))/log(2) ...
    log(err(4)/err(5))/log(2)]';
fprintf('-----\n')
fprintf('      h          dt          12 error      order\n')
fprintf('-----\n')
fprintf('%8.5f      %8.6f      %8.6f      %8.6f \n',hh(1), ...
    tt(1) ,err(1))
for iter = 2:5
    fprintf('%8.5f      %8.6f      %8.6f      %8.6f \n', ...
        hh(iter),tt(iter), err(iter),Order(iter-1))
end
fprintf('-----\n')
%%%%%%%%%%%%%

```

표준정규누적분포함수와 미분식은 다음과 같이 정의된다.

$$\begin{aligned} N(d) &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^d e^{-\frac{x^2}{2}} dx \\ N'(d) &= \frac{1}{\sqrt{2\pi}} e^{-\frac{d^2}{2}} \end{aligned}$$

### Greeks

Delta ( $\Delta$ )	$\frac{\partial V}{\partial S} = N(d_1)$
Gamma ( $\Gamma$ )	$\frac{\partial^2 V}{\partial S^2} = N'(d_1)/(S\sigma\sqrt{T})$
Theta ( $\Theta$ )	$\frac{\partial V}{\partial t} = -S\sigma N'(d_1)/(2\sqrt{T}) - rEe^{-rT}N(d_2)$
Rho ( $\rho$ )	$\frac{\partial V}{\partial r} = ET e^{-rT}N(d_2)$
Vega	$\frac{\partial V}{\partial \sigma} = S\sqrt{T}N'(d_1)$

Greeks는 말 그대로 그리스 문자들을 말한다. 각각의 그리스 문자는 각각의 지표의 변동에 대한 옵션가치의 변동비율, 즉 민감도를 의미한다. 이들 그리스문자들은 헤징과 관련하여 옵션의 투자 전략에 매우 중요하게 사용된다. Greeks은 옵션의 가격에 대한 해석해가 있는 경우, 각각의 Greeks의 정의에 따라 구한다. 그러나 대다수의 이색옵션과 같이 옵션가격의 해석해를 구할 수 없는 경우에는 수치 기법에 의존하여 Greeks의 근사적인 값을 구해야 한다.

#### 4.1.1 델타( $\Delta$ )

델타( $\Delta$ )는 주가 변화에 따른 옵션가치의 변화율을 의미한다. 옵션의 가치를  $V$ 라 하고 주가를  $S$ 라 하면 델타는 다음과 같다.

$$\Delta = \frac{\partial V}{\partial S}$$

블랙숄즈모형에서 콜옵션의 델타는 콜옵션의 가치인 식(7.34)를 주가  $S$ 에 대해 편미분해서 얻으므로,

$$\Delta = \frac{\partial V}{\partial S} = N(d_1)$$

```
E=230; sigma=0.5; r=0.03; T=1; Nx=50; Nt=100;
L=800; k=T/Nt; x=linspace(0,L,Nx); h =x(2)-x(1);
for i=1:Nx
    v(i) = max(x(i)-E,0);
end
N = Nx-2;
for i=1:N
    dd(i)=1/k+(sigma*i)^2+r;
    c(i)=-r*i/2 - ((sigma*i)^2)/2;
end
for i=1:N-1
    a(i)=r*(i+1)/2-((sigma*(i+1))^2)/2;
end
for n=1:Nt
    d=dd;
    for i=1:N-1
        b(i)=v(i+1)/k;
    end
    v(Nx)= L - E*exp(-r*k*n);
    b(N) = v(N+1)/k - c(N)*v(Nx);
    for i = 2:N
        xmult= a(i-1)/d(i-1);
        d(i) = d(i) - xmult*c(i-1);
        b(i) = b(i) - xmult*b(i-1);
    end
    v(N+1) = b(N)/d(N);
    for i = N-1:-1:1
        v(i+1) = (b(i) - c(i)*v(i+2))/d(i);
    end
```

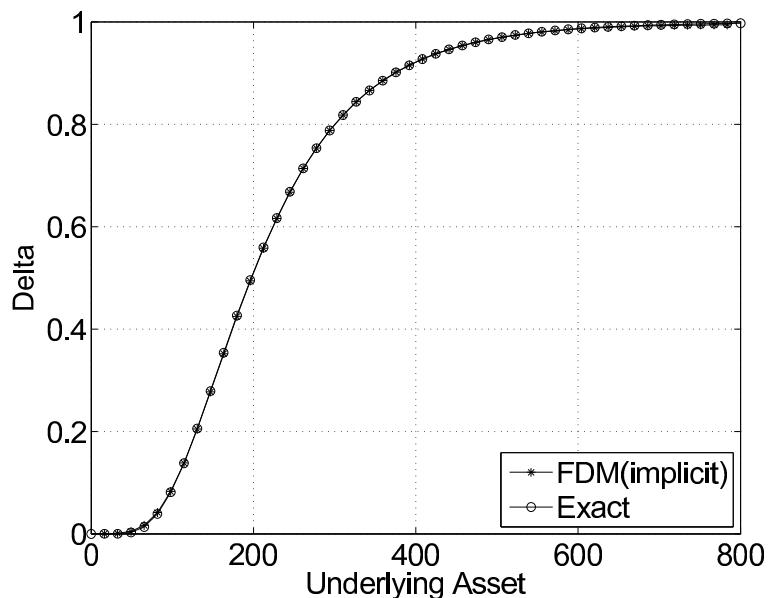


그림 7.15: 유한차분법과 해석해을 이용하여 구한 델타

$$\Gamma = \frac{\partial^2 V}{\partial S^2}$$

감마에 대한 근사식은 다음과 같다.

$$\Gamma \approx \frac{V_{i+1}^n - 2V_i^n + V_{i-1}^n}{h^2}$$

BSim\_gamma.m는 유한차분법에 의한 수치해와 블랙-숄즈 식에 의한 감마를 구하는 MATLAB 코드이다.

```
%%%%%%%% BSim_gamma.m %%%%%%
clear all; clf; clc;
E=230; sigma=0.5; r=0.03; T=1.0; Nx=50; Nt=100; L=800;
k=T/Nt;
```

```
% Gamma of numerical solution
for i = 2:Nx-1
    Gamma(i) = (v(i+1)-2*v(i)+v(i-1))/(h^2);
end
plot(x(2:Nx-1),Gamma(2:Nx-1), 'k*-'); hold on

% Gamma of exact solution
for i=1:Nx
    d1=(log(x(i)/E)+(r+sigma^2/2)*T)/(sigma*sqrt(T));
    Gamma2(i) = (exp(-0.5*(d1^2)))/(x(i)*sigma*sqrt(2*pi*T));
end
plot(x, Gamma2, 'ko-'); grid on

xlabel('Underlying Asset', 'fontsize', 20)
ylabel('Gamma', 'fontsize', 20)
legend('FDM(implicit)', 'Exact', 1)
set(gca, 'fontsize', 20)
%%%%%%%%%%%%%
```

그림 7.16은 MATLAB 코드를 실행한 결과이다. 함축적 유한차분법에 의한 감마와 해석해에 의한 감마가 매우 비슷함을 알 수 있으며 행사가격에 가까워지면 갑자기 커지는 현상을 확인할 수 있다.

#### 4.1.3 세타( $\Theta$ )

세타( $\Theta$ )는 시간( $t$ )의 변화에 따른 옵션가격의 변화율을 의미한다. 옵션의 잔존기간( $T - t$ )에 따라 옵션의 가치가 변화하는 정도를 나타내는 척도이다. 세타는 다음과 같이 옵션의 가치  $V$ 를 시간  $t$ 에 대해 편미분해서 얻는다.

```
N = Nx-2;
for i=1:N
    dd(i)=1/k+(sigma*i)^2+r;
    c(i)=-r*i/2 - ((sigma*i)^2)/2;
end
for i=1:N-1
    a(i)=r*(i+1)/2-((sigma*(i+1))^2)/2;
end
for n=1:Nt
    d=dd;
    for i=1:N-1
        b(i)=v(i+1)/k;
    end
    v(Nx)= L - E*exp(-r*k*n);
    b(N) = v(N+1)/k - c(N)*v(Nx);
    for i = 2:N
        xmult= a(i-1)/d(i-1);
        d(i) = d(i) - xmult*c(i-1);
        b(i) = b(i) - xmult*b(i-1);
    end
    v(N+1) = b(N)/d(N);
    for i = N-1:-1:1
        v(i+1) = (b(i) - c(i)*v(i+2))/d(i);
    end
    if (n == Nt-1)
        ov = v;
    end
end
% Theta of numerical solution
```

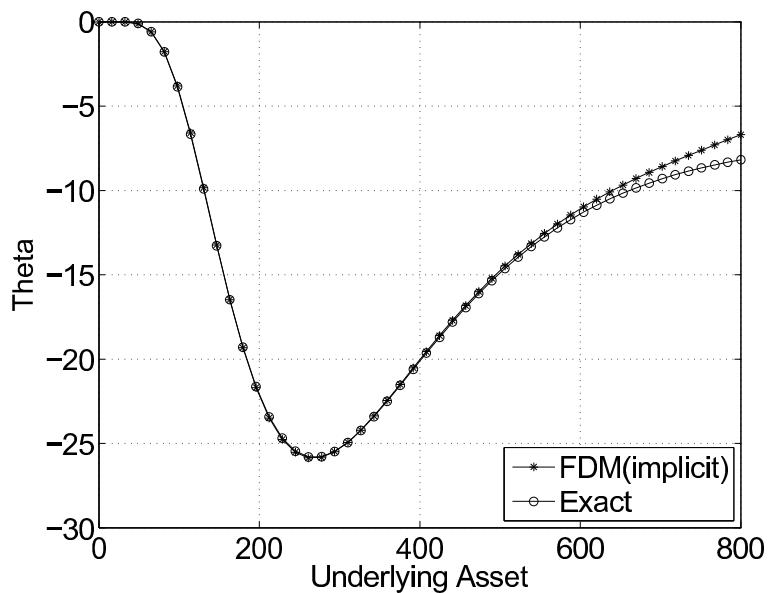


그림 7.17: 유한차분법을 이용하여 구한 세타

```
%%%%% BSim_rho.m %%%%%%
clear all; clf; clc;
E=230; sigma=0.5; T=1; Nx=50; Nt=100; L=800; k=T/Nt;
x=linspace(0,L,Nx); h =x(2)-x(1); r = [0.03 0.04];
for j = 1:2
for i=1:Nx
    v(i) = max(x(i)-E,0);
end
N = Nx-2;
for i=1:N
    dd(i)=1/k+(sigma*i)^2+r(j);
    c(i)=-r(j)*i/2 - ((sigma*i)^2)/2;
end
```

```
% Rho of exact solution
r = (r(1)+r(2))/2;
for i = 1:Nx
    d1=(log(x(i)/E)+(r+sigma^2/2)*T)/(sigma*sqrt(T));
    d2=d1-sigma*sqrt(T);
    Rho2(i) = E*T*exp(-r*T)*normcdf(d2);
end
plot(x,Rho2,'ko-'); grid on

xlabel('Underlying Asset','fontsize',20)
ylabel('Rho','fontsize',20)
legend('FDM(implicit)','Exact',4)
set(gca,'fontsize',20)
%%%%%%%%%%%%%%%

```

그림 7.18은 MATLAB 코드를 실행한 결과이다. 유한차분법에 의한 로우와 해석해에 의한 로우는 비슷함을 알 수 있으며 행사가격 근처에서 급변함을 확인할 수 있다.

#### 4.1.5 베가(Vega)

베가는 변동성  $\sigma$ 에 대한 옵션가격의 변화율을 의미한다. 베가는 그리스문자가 아니라 천문학에서 별의 이름이다. 베가는 옵션의 가치  $V$ 를 변동성  $\sigma$ 에 대해 편미분해서 얻는다.

$$vega = \frac{\partial V}{\partial \sigma} \approx \frac{V_i^n(\sigma + \Delta\sigma) - V_i^n(\sigma)}{\Delta\sigma}$$

`BSim_vega.m`는 유한차분법에 의한 수치해와 블랙-숄즈 식에 의한 베가를 구하는 MATLAB 코드이다.

```
end
for i=1:N-1
    a(i)=r*(i+1)/2-((sigma(j)*(i+1))^2)/2;
end
for n=1:Nt
    d=dd;
    for i=1:N-1
        b(i)=v(i+1)/k;
    end
    v(Nx)= L - E*exp(-r*k*n);
    b(N) = v(N+1)/k - c(N)*v(Nx);
    for i = 2:N
        xmult= a(i-1)/d(i-1);
        d(i) = d(i) - xmult*c(i-1);
        b(i) = b(i) - xmult*b(i-1);
    end
    v(N+1) = b(N)/d(N);
    for i = N-1:-1:1
        v(i+1) = (b(i) - c(i)*v(i+2))/d(i);
    end
end
if j == 1
    v1 = v;
else
    v2 = v;
end
end
% Vega of numerical solution
Vega = (v2-v1)/(sigma(2)-sigma(1));
```

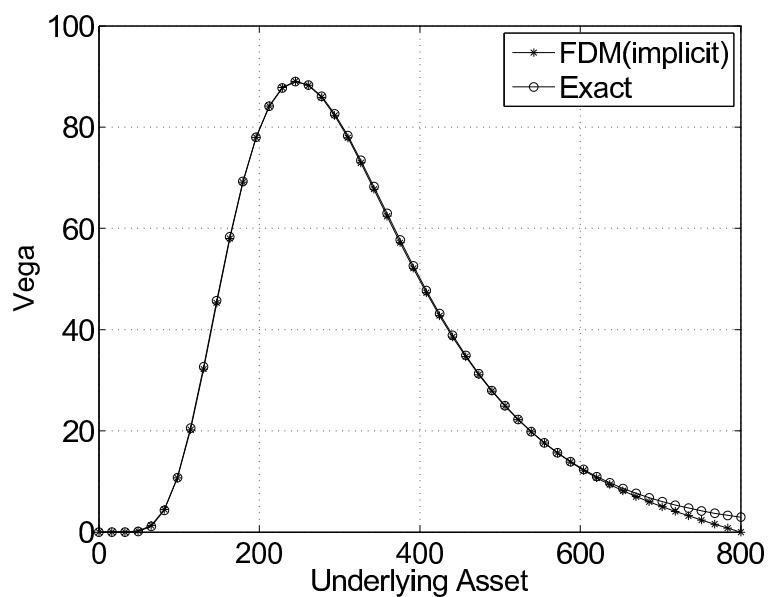


그림 7.19: 합축적 유한차분법을 이용하여 구한 베가

## 제 8 장

# Tree가격결정모형 (Tree Pricing Model)

블랙-숄즈의 옵션가격결정모형은 기초자산의 가격변화가 연속적인 값을 갖는다고 가정하여 옵션의 가격을 도출한다. 이 모형을 풀기 위해서는 확률 해석학과 편미분방정식 등을 비롯한 상당한 수학적 지식이 요구된다. 이에 의해 이산모형은 기초자산의 가격변화가 이산적인 값을 갖는다고 가정하고 옵션의 가격을 도출한다. 이 중에서 Tree모형은 기초자산의 가격이 상승하거나 하락하는 두 가지 값만을 갖는 상황에서 옵션의 가치를 구한다. Tree 모형은 블랙-숄즈의 모형에 비해 수학적으로 더 쉽게 접근할 수 있다. 그 중에서 이항모형은 1979년에 Cox, Ross 그리고 Rubinstein이 옵션의 가격을 결정하기 위해 개발한 모형이다. 이 모형은 위험중립가치평가원리를 적용하여 옵션의 가격을 산정한다.

### 제 1 절 이항옵션가격결정모형의 가정

이항옵션가격결정모형(이후 이항모형)은 다음과 같은 가정들로부터 도출된다.

**(1) 완전자본시장가정:** 자본시장에서 차익거래의 기회가 존재하지 않으며,

격  $S$ 가 기말에  $S_u$ 가 되거나  $S_d$ 가 되는지의 여부에 따라 기초 옵션의 가치  $f$ 와 기말의 옵션의 가치  $f_u$ 와  $f_d$ 는 그림 8.2와 같이 나타낼 수 있다.

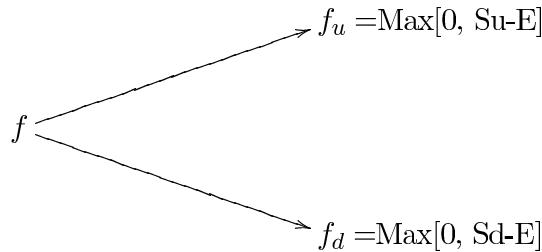


그림 8.2: 1기간 이항모형-옵션

블랙-숄즈모형에서 이용한 Covered call option strategy 즉, 콜옵션매도와 주식매수를 적절히 조합하면 무위험포트폴리오를 구성할 수 있다는 논리를 그대로 이용하여 콜옵션을 1개 매도하고 주식을  $\Delta$ 개 매입하는 무위험 포트폴리오를 구성해보자. 투자자는 콜옵션매도로부터 받은 옵션프리미엄과 은행대출로 받은 금액으로 주식을 구입한다. 그림 8.3는 무위험 포트폴리오의 가치를 나타낸 것이다. 무위험 포트폴리오는 기초자산의 가격이

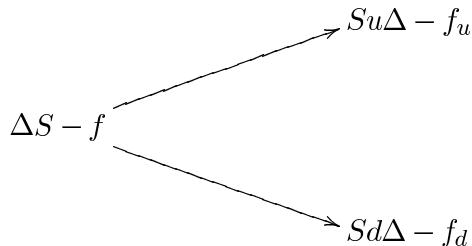


그림 8.3: 1기간 이항모형의 포트폴리오의 가치

오르거나, 내리거나 동일한 수익을 보장해주어야 하므로 아래의 식이 성립 한다.

$$S_u \Delta - f_u = S_d \Delta - f_d$$

$\Delta$ 를 구해보면

$$\Delta = \frac{f_u - f_d}{S_u - S_d} \left( \approx \frac{\partial f}{\partial S} \right) \quad (8.1)$$

이다. 기말주가  $S_T$ 의 기대값을 확률측도  $Q$ 을 이용하여 도출해 보자.

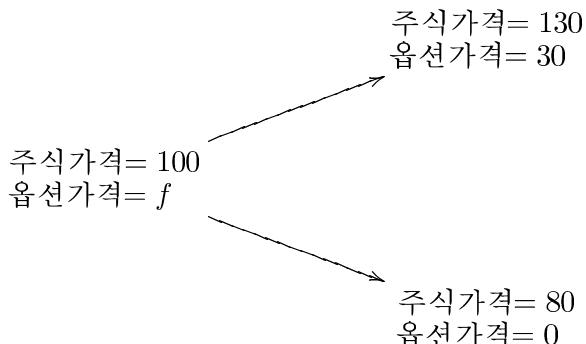
$$\begin{aligned} E(S_T) &= QS_u + (1 - Q)S_d = QS(u - d) + Sd \\ &= \frac{e^{rT} - d}{u - d} S(u - d) + Sd = Se^{rT} \end{aligned}$$

위의 식은 확률측도  $Q$ 를 사용할 경우 주가는 무위험이자율만큼의 비율로 증가해간다는 것을 보여준다. 즉 확률측도  $Q$ 를 이용하여 주가를 계산하는 것이 바로 위험중립을 가정하는 것임을 보여준다. 위에서 계산한 옵션의 가치  $f$ 도 옵션의 가치가 위험중립가정하에서 미래의 기대수익을 무위험이자율로 할인한 현재가치이다. 따라서  $Q$ 를 위험중립확률(Risk Neutral Probability)이라고 부른다.

### 예제

현재 주가가 100이고, 1기간 후의 주가가 130이나 80이 된다고 가정해보자. 1기간 후에 배당이 없는 주식을 100에 매입할 수 있는 유럽형 콜옵션의 가격을 구해보자.

먼저, 문제를 다음과 같은 그림으로 표현해보자.



이러한 주식을 기초자산으로 하는 콜옵션의 이론가격  $f$ 를 구하기 위하여 콜옵션 1계약을 매도하고  $\Delta$ 만큼 주식을 매입하여 무위험 포트폴리오를 만들어보자.

```
%%%%%%%%%%%%%% binomial_1time.m %%%%%%
T = 1; N = 1; dt = T/N; S = 100; E = 100; r = 0.10;
Su = 130; Sd = 80; St = [Su Sd]; p = 0.5;
for i = 1:N+1
    call(i) = max(St(i)-E,0);
end
i = N;
Delta = (call(i) - call(i+1))/(St(i)-St(i+1));
V = St(1)*Delta - call(1);
currentV = V*exp(-r*T);
Callprice = S*Delta-currentV
%%%%%%%%%%%%%
```

옵션의 가격은 기초자산에 비례해서 계산하기 때문에 옵션의 가치는 기초자산의 가격이 상승 또는 하락할 확률과 무관하게 결정되어 진다. 이 항모형에 의한 옵션가격결정공식을 보면, 옵션의 가치는 현재 기초자산의 가격  $S$ , 행사가격  $E$ , 무위험이자율  $r$ , 기초자산가격의 상승폭  $u$ , 기초자산 가격의 하락폭  $d$ 의 값만 주어지면 구할 수 있음을 알 수 있다.

다음 예제에서 위험중립적 가치평가로 옵션의 가치를 구해보자. 위험 중립적 가치평가는 것은 만기의 옵션가치의 기대값  $p f_u + (1-p) f_d$ 을 위험중립적 시장에서 기대현금흐름을 무위험이자율로 할인함을 의미하는 것이다.

### 예제

앞의 예제를  $S = 100, E = 100, u = 1.3, d = 0.8, r = 10\%$  일때, 위험중립적 가치평가방식에 따라 다시 풀어보기로 하자.

먼저 주가의 상승확률  $p$ 를 구해보자.

$$p = \frac{e^{0.1} - 0.8}{1.3 - 0.8} = 0.6839$$

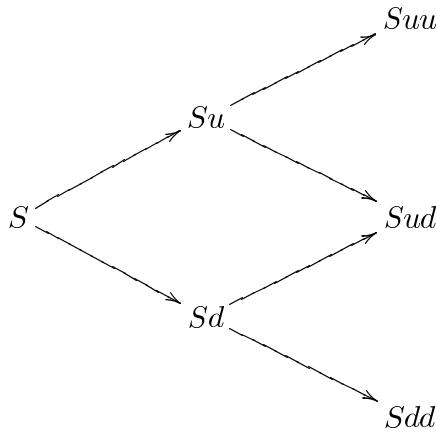


그림 8.4: 2기간 이항모형-주식

하면 1기간이 경과하면 주가는  $Su$ 와  $Sd$ 에 각각  $u$ 를 곱한 만큼 상승하거나  $d$ 를 곱한 만큼 하락하게 된다. 2기에서 주가는  $Suu$ ,  $Sud$  또는  $Sdd$ 의 세 가지 가격 중 하나가 된다. 무위험이자율은  $r$ 로 1기간은  $\Delta t$ 로 한다. 그림 8.4은 2기간 모형에서 주가의 움직임을 나타낸다. 옵션의 가격은 주가의 변화에 대응하여 같은 논리로 변화한다. 현재의 옵션가격  $f$ 는 1기간이 지난 후에  $f_u$ 가 되거나,  $f_d$ 가 된다. 여기서 다시 1기간이 경과하면 옵션의 가치는  $f_{uu}$ ,  $f_{ud}$  또는  $f_{dd}$ 가 된다.(그림 8.5) 2기간이항모형에서 1기에서 옵션이 가질 수 있는 가치  $f_u$ 와  $f_d$ 는 각각 위험중립 가정 하에서 2기에서의 옵션의 기대이익을 무위험이자율로 할인한 가치이다.

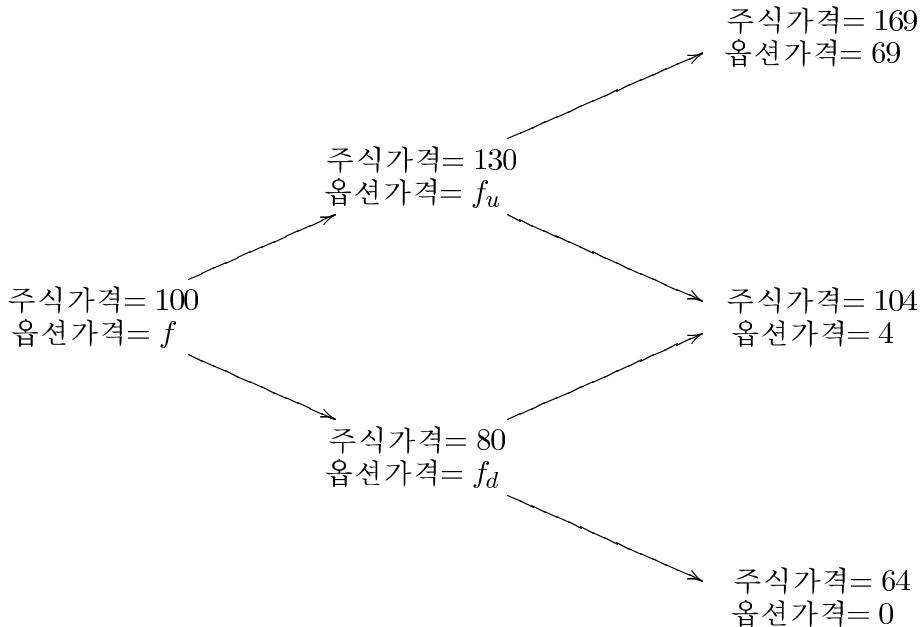
$$f_u = e^{-r\Delta t} (Q f_{uu} + (1 - Q) f_{ud}) \quad (8.3)$$

$$f_d = e^{-r\Delta t} (Q f_{ud} + (1 - Q) f_{dd}) \quad (8.4)$$

마찬가지로 초기의 옵션의 가치  $f$ 는 위험중립가정하에서 1기에서의 옵션의 기대이익을 무위험이자율로 할인한 현재가치이다.

$$f = e^{-r\Delta t} (Q f_u + (1 - p) f_d)$$

이다. 식(8.3)과 식(8.4)을 위의 식에 대입하여 현재의 옵션의 가치  $f$ 를 구



승화률인  $p$ 의 값을 구해보자.

$$p = \frac{e^{rt} - d}{u - d} = \frac{e^{0.1} - 0.8}{1.3 - 0.8} = 0.6103$$

따라서, 2기간 말의 기대가치의 현재가치를 구하면 현재시점에서 옵션의 가치를 구할 수 있다. (식 (8.5) 참고)

$$\begin{aligned} f &= e^{-0.1 \times 2} [(0.6103)^2 \times 69 + 2 \times 0.6103 \times 0.3897 \times 4 + (0.3897)^2 \times 0] \\ &= 22.6021 \end{aligned}$$

이처럼 2기간 말의 옵션가치로부터 직접 현재시점의 옵션의 가치를 구하는 대신 각 기간별로 차례대로 옵션의 가치를 구하는 방법도 있다.

먼저, 1기간 말의 옵션의 가치를 다음과 같이 구하자.

$$\begin{aligned} f_u &= e^{-0.1} (0.6103 \times 69 + 0.3897 \times 4) = 39.5163 \\ f_d &= e^{-0.1} (0.6103 \times 4 + 0.3897 \times 0) = 2.2090 \end{aligned}$$

흐름을 나타낸다. 아래 그림에서는 간단히 6기간의 이항모형을 표현하였다.

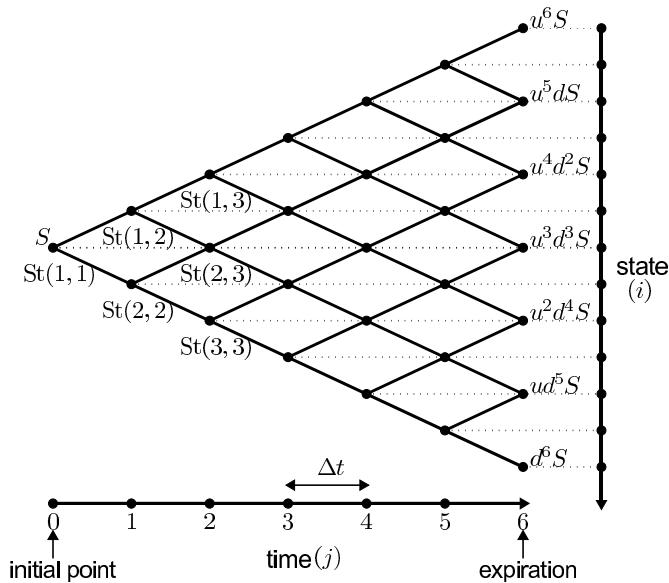


그림 8.6: 이항옵션

### 3.2 모수의 결정

주가의 이항과정을 구성하기 위해서는 모수인  $p$ ,  $u$ ,  $d$ 를 결정해야 한다. Cox와 Ross, Rubinstein의 모형에 따라  $p$ ,  $u$ ,  $d$ 를 계산해보자.<sup>3</sup> 앞에서 살펴본 대로 위험중립 가정 하에서 주식의 기대수익률은 무위험 이자율  $r$ 이다. 따라서  $\Delta t$ 기간 만의 주가의 기대값은  $Se^{r\Delta t}$ 이므로 다음의 식이 성립한다.

$$\begin{aligned} Se^{r\Delta t} &= pSu + (1-p)Sd \\ e^{r\Delta t} &= pu + (1-p)d \end{aligned}$$

주가가 랜덤워크를 따른다는 가정 하에 아주 짧은 시간  $\Delta t$ 동안 주가변화율

<sup>3</sup>Cox, Ross, Rubinstein의 모형에 따라 확률  $P$ 를 계산할 경우 시간 구간의 수  $N$ 을 증가시킬 경우  $P$ 가 음의 값을 가질 수 있다. 이러한 단점을 보완하기 위해 Jarrow와 Rudd는  $P$ 를  $1/2$ 로 고정시켜 이항모형을 구성했다.

만기에서부터 그 전기의 옵션의 기대값을 계산해가며(Backward Induction) 초기까지 이 과정을 반복하여 옵션의 가치를 계산한다. 2기간 이항모형에서의 옵션가격결정식인 식(8.5)를 이항분포의 전개식을 이용하여 다시 표현하면 다음과 같다.

$$f = e^{-2r\Delta t} \sum_{n=0}^2 \binom{2}{n} Q^n (1-Q)^{2-n} \text{Max}[0, u^n d^{2-n} S - E]$$

위 식에서  $\binom{2}{n} Q^n (1-Q)^{2-n}$ 은 2기간 동안의 가격 변동 중 주가가  $n$ 번 상승하고  $(2-n)$ 번 하락할 확률을 의미하며  $\text{Max}[0, u^n d^{2-n} S - E]$ 은 그 때의 옵션의 가치를 의미한다. 따라서 만기( $T$ )까지  $n$ 기간 남아 있는 옵션의 균형 가격은 다음과 같이 평가할 수 있다.

$$f = e^{-rT} \sum_{n=0}^T \binom{T}{n} Q^n (1-Q)^{T-n} \text{Max}[0, u^n d^{T-n} S - E]$$

이 장에서 다룬 이항옵션모형에서는 옵션의 가격이 주가의 상승확률이나 하락확률과는 무관하게 결정된다. 오직 차익거래의 기회가 존재하지 않는다는 가정만이 필요하며 주가의 상승 및 하락확률에 대한 정보는 필요하지 않다. 따라서 투자자들이 주가상승배수나 주가하락배수, 그리고 무위험 이자율에 대해 동질적인 기대를 갖는다면 주가의 상승 및 하락확률에 대해서로 다른 기대를 하더라도 옵션의 균형가격은 동일하게 평가한다. 또한 옵션의 가치를 평가하기 위해 위험중립가치평가 원칙을 적용함으로써 투자자의 위험선호도와 무관하게 옵션의 가격이 결정된다. 이는 시장이 균형 상태에 있을 때에는 무위험 포트폴리오의 수익률과 무위험 이자율이 같기 때문이다.

## 제 4 절 이항모형의 수치분석

무배당주식을 기초자산으로 하는 유럽형 콜옵션의 가치를 구하는 이항모형을 수치분석을 통해 도출해보자. 시간공간을  $i\Delta$ , 상태공간(주가)을  $j$ 로 하는 2차원 격자를 구성하고, 옵션의 만기  $T$ 를 각 구간의 길이가  $\Delta t$ 인  $N$ 개의 구간으로 나눈다.  $i\Delta$ 시점의  $j$ 번째 node를  $(i, j)$ 로 나타내면  $i\Delta$ 시점에

```

for i = N : -1 : 1      % i = time index
    for j = 1:i           % j = state index
        Call(j) = exp(-r*dt)*(p*Call(j+1)+(1-p)*Call(j));
    end
end
Callprice = Call(1)
%%%%%%%%%%%%%%%

```

결과는 다음과 같다.

Callprice = 13.9408

즉 예제의 조건을 만족하는 유럽형 콜옵션은 13.9408의 가치를 지니고 있음을 확인할 수 있다.

이항모형은 각 단계마다 기초자산의 가격이 일정한 값만큼 상승하거나 하락한다는 가정에서 출발하였다. 이러한 가정은 매우 비현실적인 것처럼 보인다. 그러나 각 단계의 시간의 크기를 작게 하여 충분히 많은 단계로 나눈 경우에는 기초자산의 가격 변동이 현실적인 분포를 갖게 된다. 실무적으로 대부분 30~50단계의 이항모형을 적용하는 것이 보통이나 이 정도가 되면 이상적인 이항분포가 거의 연속적인 분포모형으로 수렴하게 된다.

여기에서 중요한 것은 이항모형이 블랙-숄즈모형의 대체 역할 뿐 아니라 훨씬 다양한 형태의 옵션의 가치를 평가하는 데 활용된다는 점이다. 이항모형은 블랙-숄즈모형에 비해 제약적인 가정이 훨씬 적기 때문에 복잡한 구조를 갖는 옵션의 가격을 계산하는 데 활용될 수 있다. 예를 들면, 조기 행사가 가능한 미국형 옵션이나 불규칙한 현금흐름을 갖는 자산에 대한 옵션의 가치를 평가할 경우 블랙-숄즈 모형을 이에 맞게 변형하는 것보다 이항모형을 이용하는 것이 훨씬 효과적인 방법이 된다.

## 제 9 장

# 몬테칼로 시뮬레이션 (Monte Carlo Simulation)

이 장에서는 통계적 표본추출법(Statistical Sampling)을 이용하는 몬테 칼로 기법에 대해서 알아볼 것이다. 몬테 칼로를 이용하여 주가지수 프로세스를 생성하기 위해서는 주가지수를 따르는 확률분포를 구해야 한다. 주가지수가 따르는 연속확률분포를 이산형의 난수로 대체하고 시뮬레이션을 통해 난수를 추출하여 그 난수들이 갖는 분포를 찾고 원래의 주가지수가 따르는 확률분포의 함수에 대한 근사값을 구하는 것이 몬테칼로방법이다.

### 제 1 절 몬테 칼로 시뮬레이션의 과정

몬테칼로 방법을 이용하여 옵션가격을 결정하는 과정은 다음과 같이 요약 할 수가 있다.

- $M$ 의 소인수는  $a - 1$ 의 인수이다.
- 만약  $M$ 이 정수 4으로 나누어지면  $a - 1$ 도 4으로 나누어진다.

위의 조건에 의해 모수가 결정되면  $R_{n+1}$ 를 구하고 이를  $M$ 으로 나누어  $U_{n+1}$ 를 얻는다.

$$U_{n+1} = \frac{R_{n+1}}{M}, \quad n = 0, 1, 2, \dots$$

```
%%%%%%%% uniform_rand1.m %%%%%%
a=7; c=1; M=18; n=M+3; k1=zeros(n,2); R=zeros(n+1,1);
U=R; R(1)=1; U(1)=R(1)/M;
fprintf('Iteration Random integer Random number \n');
for i=2:n+1
    R(i)=mod(a*R(i-1)+c,M);
    U(i)=R(i)/M;
    fprintf('      %d      %d
i-1, R(i-1), U(i-1));
end
%%%%%%%%%
```

위의 코드 `uniform_rand1.m`를 실행한 결과는 다음과 같다.

```
>> uniform_rand1
Iteration Random integer Random number
 1      1      0.055556
 2      8      0.444444
 3      3      0.166667
 4      4      0.222222
 5     11      0.611111
 6      6      0.333333
 7      7      0.388889
 8     14      0.777778
```

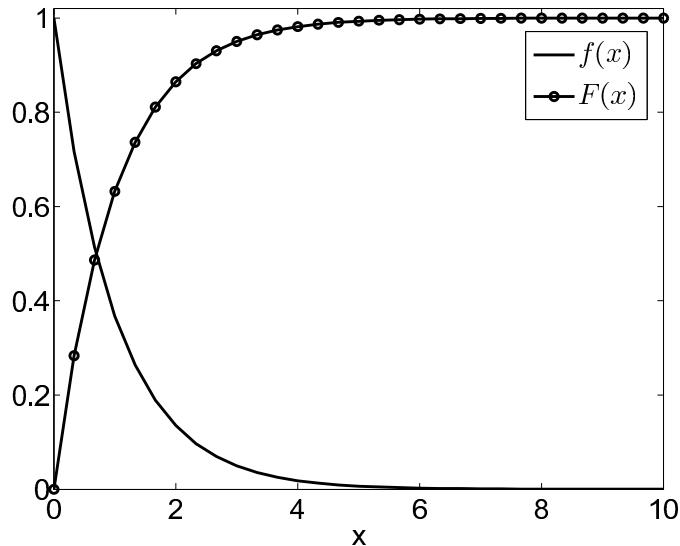


그림 9.1: 확률밀도 함수  $f(x) = e^{-x}\chi_{[0,\infty)}(x)$ 의 확률분포함수와 누적분포함수

따라서  $F(x)$ 의 역함수는  $F^{-1}(u) = 2\pi u$ 로 주어진다.

확률벡터  $(x, y)$ 가 2변량 표준정규분포를 따르면, 결합확률밀도함수 (Joint Probability Density Function)와 결합확률분포함수 (Joint Probability Distribution Function)는 각각 다음과 같다.

$$\begin{aligned} f(x, y) &= \frac{1}{2\pi} e^{-\frac{x^2+y^2}{2}} \\ F(x, y) &= \int_{-\infty}^y \int_{-\infty}^x f(u, v) du dv \end{aligned}$$

극좌표(polar coordinate), 일양분포의 역함수법, 그리고 지수분포의 역함수법을 조합해서, 정사각형  $[0, 1] \times [0, 1]$ 에서 일양분포하는 점열로부터 2변량 표준정규분포를 따르는 점열을 구할 수 있다. 다음과 같은 극좌표들로 정규확률벡터  $(x, y)$ 를 변환시키자 (그림 9.5 참고).

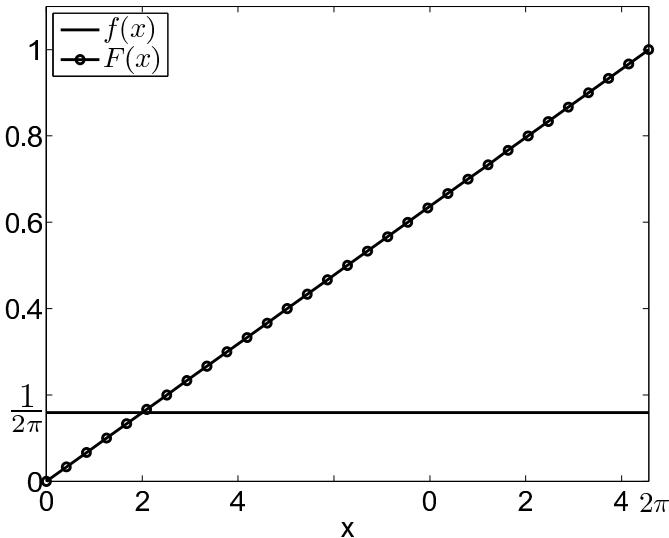


그림 9.3: 확률밀도 함수  $f(x) = \frac{1}{2\pi}\chi_{[0,2\pi]}(x)$ 의 확률분포함수와 누적분포함수

$$J := \left| \begin{array}{c} \partial(x,y) \\ \partial(r,\theta) \end{array} \right| = \left| \begin{array}{cc} \frac{\partial x}{\partial r} & \frac{\partial y}{\partial r} \\ \frac{\partial x}{\partial \theta} & \frac{\partial y}{\partial \theta} \end{array} \right| = \left| \begin{array}{cc} \cos \theta & \sin \theta \\ -r \sin \theta & r \cos \theta \end{array} \right| = r$$

$$\begin{aligned} F(x, y) &= \int \int_{\Omega_{xy}} f(u, v) dudv = \int \int_{\Omega_{xy}} \frac{1}{2\pi} e^{-\frac{u^2+v^2}{2}} dudv \\ &= \int \int_{\Omega_{r\theta}} \frac{1}{2\pi} e^{-\frac{r^2}{2}} r dr d\theta = \int \int_{\Omega_{r\theta}} f(r, \theta) dr d\theta \end{aligned}$$

따라서, 2변량 표준정규분포의 확률밀도함수  $f(r, \theta)$ 는 다음과 같다.

$$f(r, \theta) = \frac{1}{2\pi} r e^{-\frac{1}{2}r^2} \chi_{[0, \infty) \times [0, 2\pi]}(r, \theta) \quad (9.6)$$

$$= r e^{-\frac{1}{2}r^2} \chi_{[0, \infty)}(r) \frac{1}{2\pi} \chi_{[0, 2\pi]}(\theta) \quad (9.7)$$

또한, 변수변환  $s = r^2/2$  의 해서, 위의 식을 다음과 같이 쓸 수 있다.

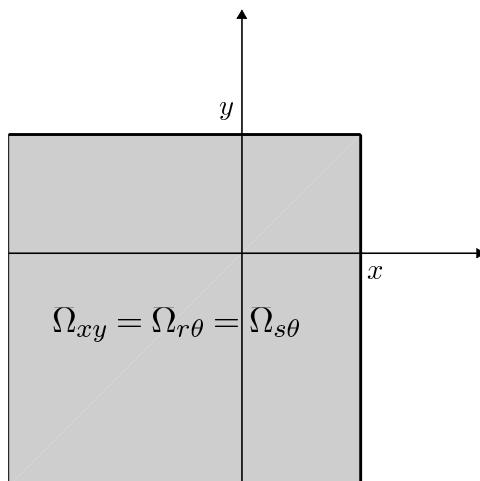


그림 9.5: 결합확률 분포함수를 계산하는 영역

극좌표로 나타낸 표준정규확률벡터이다. 지금까지의 내용을 정리하면 다음과 같다. 서로 독립이며 구간  $[0, 1]$ 에서 균등분포하는 확률변수  $u$ 와  $v$ 에 대해서, 다음과 같이 확률변수  $x$ 와  $y$ 를 정의하자.

$$y = \sqrt{-2 \log(1-u)} \sin(2\pi v),$$

이 확률변수  $x$ 와  $y$ 는 서로 독립이며 또한 표준정규분포를 따른다. 위의 변환방법을 Box-Muller 변환이라 하고, 이를 이용하면, 2변량 균등분포를 따르는 확률벡터  $(u, v)$ 로부터 2변량 정규확률벡터  $(x, y)$ 를 구할 수 있다. `box_muller.m`은 Box-Muller의 방법으로 표준정규분포를 따르는 난수를 생성하는 MATLAB 코드이다.

```
%%%%%%%% box_muller.m %%%%%%
clear; clc; clf; N=10000; U1=rand(N,1); U2=rand(N,1);
Z1=sqrt(-2*log(U1)).*cos(2*pi*U2);
Z2=sqrt(-2*log(U1)).*sin(2*pi*U2);
```

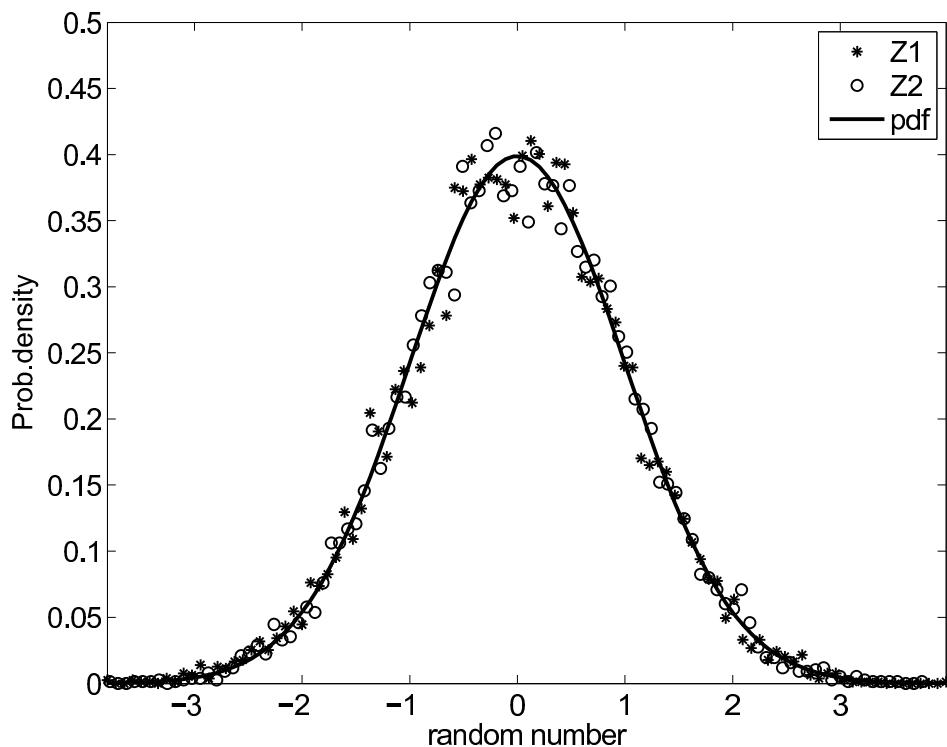


그림 9.6: Box-Muller에 의해 생성된 난수의 확률밀도분포와 확률밀도함수

행렬  $A$ 를  $L$ 과  $L^T$ 의 행렬 곱으로 나타내보자.

$$A = \begin{pmatrix} a & b \\ b & c \end{pmatrix} = LL^T = \begin{pmatrix} \alpha & 0 \\ \beta & \gamma \end{pmatrix} \begin{pmatrix} \alpha & \beta \\ 0 & \gamma \end{pmatrix} = \begin{pmatrix} \alpha^2 & \alpha\beta \\ \alpha\beta & \beta^2 + \gamma^2 \end{pmatrix}$$

각각의 원소들을 비교하면

$$\alpha^2 = a, \quad \alpha\beta = b, \quad \beta^2 + \gamma^2 = c$$

을 얻게 되고, 이를 다시 정리하면 다음의 결과를 얻을 수 있다.

$$\alpha = \sqrt{a}, \quad \beta = \frac{b}{\sqrt{a}}, \quad \gamma = \sqrt{c - \frac{b^2}{a}}$$

$$\begin{aligned}
 \begin{pmatrix} \phi_1^* \\ \phi_2^* \end{pmatrix} &= L \begin{pmatrix} \phi_1 \\ \phi_2 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ \rho & \sqrt{1-\rho^2} \end{pmatrix} \begin{pmatrix} \phi_1 \\ \phi_2 \end{pmatrix} \\
 &= \begin{pmatrix} \phi_1 \\ \phi_1\rho + \phi_2\sqrt{1-\rho^2} \end{pmatrix} \tag{9.9}
 \end{aligned}$$

그렇다면 왜 식 (9.9)과 같은 방식으로 계산을 하면 두 난수가 상관계수  $\rho$ 를 갖는 난수로 변하게 되는 것일까? 이는 간단하게 확인해볼 수 있다.

$\phi_1, \phi_2 \sim N(0,1)$ 의 성질을 가지고 있으며 위의 행렬에 따라 다음이 성립함을 확인할 수 있다.

$$\begin{aligned}
 \phi_1^* &= \phi_1, \\
 \phi_2^* &= \phi_1\rho + \phi_2\sqrt{1-\rho^2}
 \end{aligned}$$

이제 두 난수  $\phi_1^*, \phi_2^*$ 에 대하여 다음을 계산해 보자.

$$\begin{aligned}
 E[\phi_1^*] &= E[\phi_1] = 0, \\
 E[\phi_2^*] &= E[\phi_1\rho + \phi_2\sqrt{1-\rho^2}] = \rho E[\phi_1] + \sqrt{1-\rho^2}E[\phi_2] = 0, \\
 Var[\phi_1^*] &= Var[\phi_1] = 1, \\
 Var[\phi_2^*] &= Var[\phi_1\rho + \phi_2\sqrt{1-\rho^2}] \\
 &= E[(\phi_1\rho + \phi_2\sqrt{1-\rho^2})^2] - E[\phi_1\rho + \phi_2\sqrt{1-\rho^2}]^2 \\
 &= \rho^2 + 1 - \rho^2 = 1.
 \end{aligned}$$

두 확률변수  $X$ 와  $Y$ 의 공분산(Covariance)는 다음과 같이 정의된다.

$$\begin{aligned}
 Cov[X, Y] &= E[(X - E[X])(Y - E[Y])] \\
 &= E[XY - E[Y]X - E[X]Y + E[X]E[Y]] \\
 &= E[XY] - E[Y]E[X] - E[X]E[Y] + E[X]E[Y] \\
 &= E[XY] - E[X]E[Y]
 \end{aligned}$$

른다고 하자. 그러면 지수의 변화는 다음의 식으로 나타낼 수 있다.

$$\frac{dS}{S} = \mu dt + \sigma dX$$

위 험중립원칙 하에서  $\mu$ 를 무위험 이자율  $r$ 로 대체하자.

$$\frac{dS}{S} = rdt + \sigma dX$$

주가에 자연로그를 취하고  $\ln S$ 의 확률과정에 이토의 렘마를 적용하면

$$d\ln S = \left( r - \frac{\sigma^2}{2} \right) dt + \sigma \sqrt{dt} \phi$$

을 얻고 이를 이산모형으로 변환하면 다음을 얻는다.

$$\begin{aligned} \ln \left( \frac{S_{t+\Delta t}}{S_t} \right) &= \left( r - \frac{1}{2}\sigma^2 \right) \Delta t + \sigma \phi \sqrt{\Delta t}, \\ S_{t+\Delta t} &= S_t \exp \left[ \left( r - \frac{1}{2}\sigma^2 \right) \Delta t + \sigma \phi \sqrt{\Delta t} \right], \quad \phi \sim N(0, 1) \end{aligned}$$

모수들  $S_t, r, \sigma, \Delta t$ 가 주어지고 난수  $\phi$ 을 생성하면  $S_{t+\Delta t}$ 를 구할 수 있다. `stock_process.m`는 지수과정을 구하는 MATLAB 코드이다.

```
%%%%% stock_process.m %%%%%%
clear; S(1)=100; r=0.03; vol=0.3; T=1; N=100; dt=T/N;
t=linspace(0,T,N+1); w=randn(1,N);
for i=2:N+1,
    S(i)=S(i-1)*exp((r-1/2*vol^2)*dt+vol*w(i-1)*sqrt(dt));
end
plot(t,S,'*-'); xlabel('Time'); ylabel('Stock Price');
%%%%%
```

그림 9.7 은 위의 `stock_process.m` MATLAB 코드를 실행한 결과를 나타낸 것이다.

그림 9.8 은 지수 프로세스 100개를 실행한 결과를 나타낸 것이다.

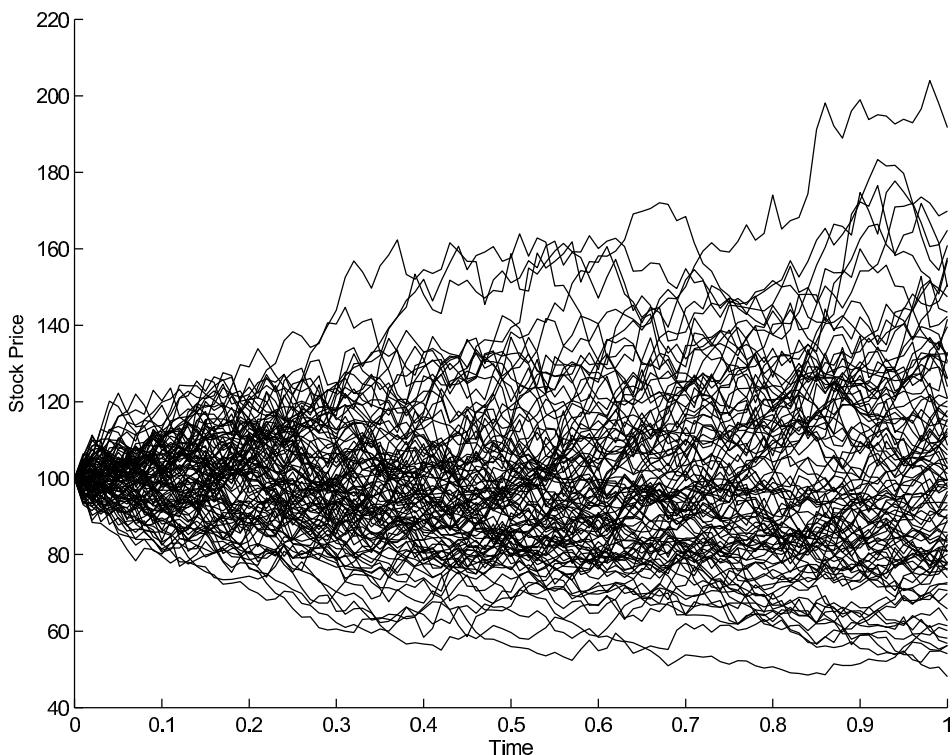


그림 9.8: 100개의 지수 과정 생성

$N$ 으로 나누면 옵션의 기대값을 구할 수 있다. 유럽형 콜옵션의 만기시의 기대값은 다음과 같다.

$$E_Q(f_T) = \frac{1}{N} \sum_{i=1}^N \text{Max}(S_{T,i} - E, 0)$$

## 제 6 절 옵션가치 도출

기대값을 무위험이자율로 할인하여 옵션가치를 도출한다. 유럽형 콜옵션의 가치는 다음과 같다.

Price = 9.3821

난수를 뽑아 계산하므로 매번 시행마다 Price는 다른 결과를 얻게 되지만, 비슷한 결과를 얻게 된다는 사실을 알 수 있다. 위에서 몬테칼로 시뮬레이션 실행시 MATLAB 코드에 나와 있는  $n = 1$ 로 두고 하는 이유는 무엇일까? 정규분포( $N(0, 1)$ )를 따르는 난수  $\phi_i$  ( $1 \leq i \leq n$ )가 있다고 가정했을 때,

$$\sum_{i=1}^n \sqrt{\Delta t} \phi_i = \phi_1 \sqrt{\Delta t} + \cdots + \phi_n \sqrt{\Delta t} = \phi \sqrt{n \Delta t} \quad (9.10)$$

이 성립하기 때문이다. 즉, 전체 시간을  $n$ 개로 나누어 여러 번에 걸쳐 계산한 다음 합산한 것이나 전체 시간을 한 번에 계산하는 것이 확률적으로 동일한 값을 갖는다는 것이다.

$$\begin{aligned} E\left[\sum_{i=1}^n \sqrt{\Delta t} \phi_i\right] &= \sum_{i=1}^n \sqrt{\Delta t} E[\phi_i] = 0 \\ Var\left[\sum_{i=1}^n \sqrt{\Delta t} \phi_i\right] &= \sum_{i=1}^n \Delta t Var[\phi_i] = \sum_{i=1}^n \Delta t = n \Delta t \end{aligned}$$

따라서, 다음이 성립함을 확인할 수 있다.

$$\sum_{i=1}^n \sqrt{\Delta t} \phi_i = \sqrt{\Delta t} \phi,$$

여기서  $\phi \sim N(0, n\Delta t)$ 이다. 그러므로 몬테칼로 시뮬레이션을 시행할 때 대부분의 경우에 있어서  $n = 1$ 로 두는 것이다.

다음은 두개의 기초자산이 있는 경우에의 MC 방법에 대해서 알아보자. MC\_call2d.m 은 MATLAB 코드이다. 이를 실행하면 다음과 같은 결과를 얻는다.

Price = 14.2164

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% MC_call2d.m %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear; S1=100; S2=100; E1=100; E2=100; r=0.05; T=0.5;
```

## 참고 문헌

- [1] Brandimarte P., *Numerical Methods in Finance and Economics*, Wiley, 2/E, 2006
- [2] Clewlow L., Strickland C., *Implementing Derivatives Models*, John Wiley & Sons, 1998
- [3] Glasserman P., *Monte Carlo Methods in Financial Engineering*, Springer, 2003
- [4] Higham D.J., *An Introduction to Financial Option Valuation*, Cambridge Univ Press, 2004
- [5] Hull J.C., *Options, Futures, and Other Derivatives*, Prentice Hall, 7/E, 2008
- [6] Seydel R.U., *Tools for Computational Finance*, 3/E, Springer , 2006
- [7] Taleb N., *Dynamic Hedging ; Managing Vanilla and Exotic options*, John Wiley & Sons, 1996
- [8] Wilmott P., *Paul Wilmott on Quantitative Finance*, 2/E, Wiley , 2006
- [9] Wilmott P., Howison S., Dewynne J., *The Mathematics of Financial Derivatives*, Cambridge Univ Press, 1995

크랭크 니콜슨 (Crank-Nicolson) 방법, 98

토마스 알고리즘(Thomas algorithm) 방법, 94

파생금융상품, 25

폰 노이만 (von Neumann) 방법, 89

풋옵션(put option), 25

함축적 (Implicit) 유한 차분법, 91

확률미분방정식(stochastic differential equation), 60

후방차분(backward difference), 86