

# 財務演算法

## Financial Algorithms

昀騰金融科技

技術長

董夢雲 博士

[dongmy@ms5.hinet.net](mailto:dongmy@ms5.hinet.net)

# 目 錄

## Part I：Black-Scholes(73)與 Black(76)解析解

- 一、交易室工作的介紹與開源工具的使用
- 二、Visual Studio 2022 與 VS Code
- 三、BS 模型的程式設計分析
- 四、Greeks 與隱含波動性
- 五、利率期限結構的理論與實作
- 六、利率交換程式實作
- 七、利率選擇權與交換選擇權程式實作

# Part II : Tree 、 PDE & Simulation

- 一、Binomial Tree
- 二、Trinomial Tree
- 三、MC Simulation
- 四、Finite Difference
- 五、LSMC Simulation

## Part III：其他模型

- 一、Markowitz Model(自己學習)
- 二、Black-Litterman Model(自己學習)
- 三、Interest Rate Models

# Part IV : Application

一、Structured Products

二、Basel III

三、Trading System

# 昀騰金融科技股份有限公司

技術長  
金融博士、證券分析師

董夢雲 Andy Dong



Line/WeChat: andydong3137  
E: andydong1209@gmail.com  
<https://github.com/andydong1209>  
M: (T)0988-065-751(C)1508-919-2872  
10647 台北市大安區辛亥路一段 50 號 4 樓

**現職：**國立台灣大學財務金融研究所兼任教授級專家  
台灣金融研訓院 2024 年菁英講座

**經歷：**中國信託商業銀行交易室研發科主管  
凱基證券風險管理部主管兼亞洲區風險管理主管  
中華開發金控、工業銀行風險管理處處長  
永豐金控、商業銀行風險管理處處長  
永豐商業銀行結構商品開發部副總經理

**學歷：**國立台灣大學電機工程學系學士  
國立中央大學財務管理學研究所博士

**專業：**證券暨投資分析人員合格(1996)

**專長：**Basel III 交易簿市場風險資本計算、銀行簿利率風險計算  
風險管理理論與實務，資本配置與額度規劃、資產負債管理實務  
外匯與利率結構商品評價實務，股權與債權及衍生商品評價實務  
GPU 平行運算與結構商品系統開發，CUDA、OpenCL  
CPU 平行運算與 ALM 系統開發，C#/C++/C、.Net Framework、SQL  
人工智慧(Deep Learning)交易策略開發，Python、Keras、TensorFlow

# O、序 言

## ◆ 我在臉書上會發表文章，

- <https://www.facebook.com/andy.dong.735>

## ◆ 我有在知乎上發表文章。

- 快 100 篇文章，有數篇文章談到金工的學習。
- <https://zhuanlan.zhihu.com/p/667111668>

## ◆ 我的臉書有一個簡單的財工書目，可以參考。

- [https://m.facebook.com/story.php?story\\_fbid=4678747135469331&id=100000022735693](https://m.facebook.com/story.php?story_fbid=4678747135469331&id=100000022735693)

## ◆ 我有維持一個 QuantLib 的學習社群，QuantLib-昀騰學習園地，可以去看看。

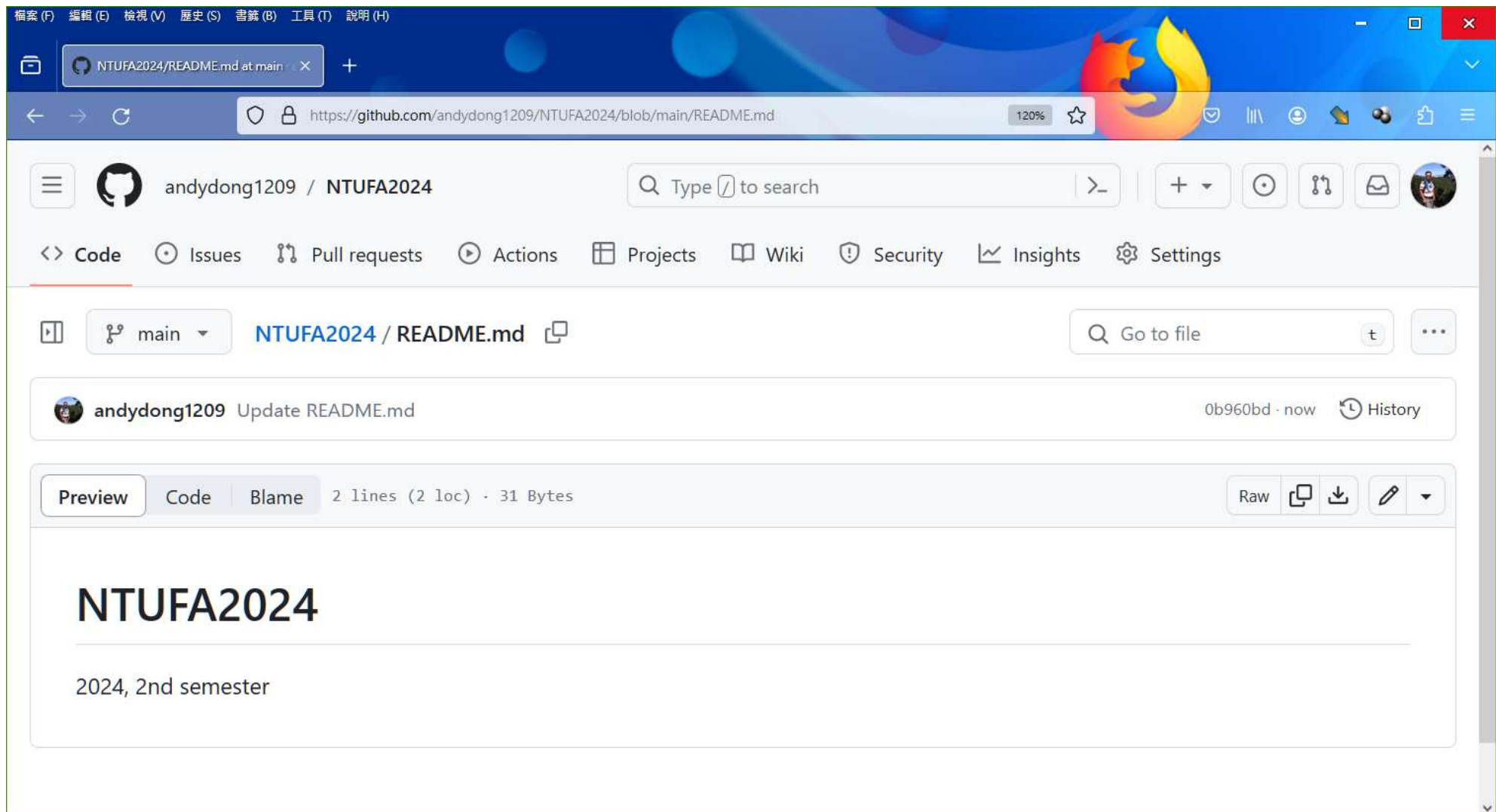
- <https://www.facebook.com/groups/421949278879071>

## ◆ 我會在金融研訓院、證券暨期貨市場發展基金會、MasterTalks，等地方上實務課程。

- 一些實務應用的課程。



◆ 課程網站(<https://github.com/andydong1209/NTUFA2024>)



The screenshot shows a web browser window with the following elements:

- Browser Tab:** NTUFA2024/README.md at main
- Address Bar:** <https://github.com/andydong1209/NTUFA2024/blob/main/README.md>
- Repository Header:** andydong1209 / NTUFA2024
- Navigation Bar:** Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, Settings
- File Path:** main / NTUFA2024 / README.md
- Commit Information:** andydong1209 Update README.md (0b960bd · now)
- File View:** Preview, Code, Blame (2 lines (2 loc) · 31 Bytes)
- File Content:**  

# NTUFA2024

  
2024, 2nd semester

# 一、財務工程的應用方向

## ◆ 金融產品的應用

- 結構商品的開發
- 評價與避險的議題
- 知乎 1~13 選擇權評價，使用 QuantLib-Python
- 知乎 14~19 雪球結構品，Black-Scholes 模型定價
- 知乎 21~23 雪球結構品，Heston 模型定價
- 知乎 38~46 零息可贖回債，CMS 每日區間計息可贖回債
- 知乎 69~70 Hybrid 模型 Heston-Hull-White 模型應用

## ◆ 金融交易的應用

- 指數選擇權高頻套利系統的開發
- 定價與套利的議題
- 知乎 34~37 SABR 選擇權高頻套利開發

## ◆ 金融風險的管理

- Basel III FRTB & IRRBB 計算模組
- 敏感性與風險值的計算
- 知乎 49~53 T-Bond FRTB 風險資本計算
- 知乎 71~79 Swap FRTB 風險資本計算與交易對手信用風險(PSR)計算
- 知乎 55~61 Heston SLV 模型的應用

## ◆ 高速運算應用相關

- 知乎 23~28 GPU CUDA 的使用
- 知乎 31~32 Monte-Carlo Simulation 的金融應用

## 二、參考資料

### 甲、入門

- ◆ \*\*Financial Numerical Recipes in C++, 2014, Bernt Odegaard 。
  - <http://www1.uis.no/ansatt/odegaard/> 。
- ◆ Numerical Methods in Finance with C++, Maciej J. Capinski, Cambridge 。
- ◆ \*The Complete Guide to Option Pricing Formulas, 2nd Ed., Espen Haug, McGraw-Hill 。
- ◆ A Course in Derivative Securities, Kerry Back, Springer 。
- ◆ Derivatives Analytics with Python, Yves Hilpisch, Wiley 。
- ◆ \*財務工程與 Excel VBA 的應用：選擇權評價理論之實作，董夢雲，證基會 。
- ◆ C++財務程式設計，戴天時，證期會 。

## 乙、中級

- ◆ \*Implementing Derivatives Models, Les Clewlow & Chris Strickland, John Wiley ◦
- ◆ C++ for Financial Mathematics, John Armstrong, Chapman & Hall/CRC ◦
- ◆ \*Quantitative Finance : An Object-Oriented Approach in C++, Erik Schlogl, Chapman & Hall/CRC ◦

## 丙、高級

- ◆ \*C# for Financial Markets, Daniel J. Duffy & Andrea Germani, Wiley ◦
- ◆ Financial Instrument Pricing Using C++, 2nd Ed., Daniel J. Duffy, Wiley ◦
- ◆ \*Monte Carlo Frameworks: Building Customizable High-performance C++ Applications, Daniel J. Duffy & Joerg Kienitz, Wiley ◦
- ◆ \*C++ Design Patterns and Derivatives Pricing, Mark Joshi, Cambridge ◦

## 丁、電腦書

- ◆ 最新 C 程式語言，施威銘研究室，旗標。有 BCB 免費編譯器 IDE。
- ◆ C++程式設計，第三版，張耀仁，碁峯。
- ◆ Visual C# 2017 程式設計經典，蔡文龍，碁峯。
- ◆ Visual C++ 2012 教學手冊，Ivor Horton 著，蔡明志譯，碁峯。
- ◆ 深度學習 C++，第三版，子由，藍海文化。
- ◆ C#本質論，第六版，Mark Michaelis & Eric Lippert，周靖譯，人民郵電。
- ◆ C++ Primer Plus，第六版，Stephen Prata，蔡明志譯，碁峯。
- ◆ C++程序設計語言，第四版，Bjarne Stroustrup，裘宗燕譯，人民郵電。
- ◆ C++編程思想，Bruce Eckel，刁成嘉譯，人民郵電。

## 三、進行方式

### ◆ 老師提供上課教材(投影片講義)

- Odengaard 講義、證基會講義、與老師著作
- 以 C++、Excel VBA 環境說明演算邏輯

### ◆ 學生使用程式語言實作，交成品

- 可以接受各類的語言，但要有 GUI。
- 包括 C/C++、C#(.Net)、Java、Pascal(Delphi)、Python、VBA。
- 期末交一個程式來展示。

### ◆ 主要理論與演算法參考書籍

- John Hull
- Les Clewlow & Chris Strickland
- Espen Gaarder Haug
- 10 Best FE Books

## 四、評分方式

- ◆ 期中作業(50%)：一次，每人自己做
- ◆ 期末程式(50%)：3 人一組
- ◆ Bonus：課程討論的參予程度



## 五、Odegaard 教書心得

### ◆ 現代財務數學化的起源，

- Black-Scholes-Merton 選擇權定價理論
- 高度實用性與可操作性，背後涉及大規模資本的移動
- Derivatives 與 Fixed Income Instruments 是財務計算的主要對象

### ◆ 此書的起源，

- Derivatives Course at Univ. of British Columbia, Canada。
- 實作增進理解。

### ◆ 為何要用 C++寫財務程式？

- 只用 Excel + VBA、Matlab、Python 不行嗎？
- 使用其他“高階”語言，最後總會“力有未逮”。
- 只有正統的程式語言，才能做到氣隨意走。

## ◆ C++是財工人員的母語，最強大的語言。

- Fortran, C#(.Net), Java 都可考慮。
- GPU 高能運算，CUDA、OpenCL 都是用 C/C++。
- C#, Java 是可接受的替代產品

## ◆ 為誰而寫，

- 財務學生==>學公式的實作
- 財務學生==>學 C++
- 資訊學生==>學財務演算法

## 六、我的工作與學習心得

### ◆ 財工人員實務使用的工具很缺乏

#### ➤ 銀行有買大型系統

- ✓ 如 Murex、Summit for Trading，Oracle、Sungard for ALM，RiskMetrics for Risk MGT。
- ✓ 但通常是為例行業務工作使用。

#### ➤ 銀行可能會有一些小型計算系統

- ✓ 如 Numerix、FinCAD、PriceIt、Bloomberg。
- ✓ 一些驗證、評價使用。

#### ➤ 價錢不便宜，有限功能

### ◆ 財工人員最常(基本)使用的工具

- Excel 是最基本的工具
- Python & R
- Matlab(?)

## ◆ 這門課的目的

### ➤ 有了這些基本知識，才能理解 QuantLib 等程式庫

- ✓ 網路上有很多免費有用的程式庫，Numerical Recipe in C/C++/C#，math.net，GSL，...。

### ➤ 目標是去使用 QuantLib，不要自己寫另一個程式庫

- ✓ QuantLib 是免費高品質的程式庫
- ✓ 使用難度高，但回報也是巨大的
- ✓ Mark Joshi 認為學會使用 QuantLib，是現代財工學習第一要項

### ➤ Tailor-made 的產品是未來財富管理的重頭戲

- ✓ 台灣、大陸的市場變化
- ✓ 用 QuantLib 為基礎，開發銀行適用的專屬系統

## ◆ 交易室(銀行)的工作情況

### ➤ 銀行、證券、保險體系的差異

## ◆ QuantLib 的多種用途

- QuantLibXL，Excel Spreadsheet 可以直接使用
  - ✓ QuantLibXL，Excel VBA 可以 Call。
- QuantLib-Python，Python 可以 import 使用，Windows & Linux 皆可使用。
- QuantLib C#，.Net 語言可以使用，Windows & Linux 皆可使用。
- QuantLib C++，Windows & Linux 皆可使用。

## ◆ GUI 的選擇

- QuantLibXL，Excel Spreadsheet 當 IO 使用
  - ✓ QuantLibXL，Excel VBA 可以用 Form。
- QuantLib-Python，可以使用 Tk/Tcl，PyQt(Qt)，wxPython(wxWidgets)。
- QuantLib C#，Windows & Linux 皆可使用 Windows Form。
- QuantLib C++，Windows & Linux 皆可使用 Nana、FLTK、Qt。

## ◆ 財務演算是一門綜合的應用課題。

➤ 涉及理論與實務兩個範疇。

➤ 理論面涵蓋：

- ✓ 財務模型：評價理論、風險理論、投資組合理論
- ✓ 隨機微積分：I'to Lemma
- ✓ 演算法：尋根、內插、矩陣運算、線性代數、模擬、非線性最適化

➤ 實務面涵蓋：

- ✓ 程式語言、物件導向
- ✓ 泛型、平行運算、設計樣式
- ✓ GUI、資料庫

## ◆ P Measure & Q Measure

- 本課程是以 Q Measure 的應用為主。
- P Measure 應用可以考以參考『AI 金融交易應用』。

## ◆ 理論的知識當然是重要的，且須不斷精進

- 現在使用的模型，LSMC, Heston, Hull-White, G2++, HJM, LMM，都不是我在博士班時期學習的
- 然而，沒有電腦實作技能，產值也是零

## ◆ 電腦科技發展快速，需不斷學習了解新技能

- OO、GP、DP、Parallel Computing，GPU 都是畢業後學習
- 最好早點熟習這些新技術，工作才能盡快上手
- 現在工作很競爭，每個人要建立自己的核心優勢
- 不是每個人都要做財工，但須要有基礎涵養

## ◆ 畢業後個人的產值函數，

$$V = f(F, M, C)$$

- Financial Theory
- Mathematics
- Computer Science(Coding)

## ◆ C++抽象化的能力很重要

- Object-Oriented 是大型系統必要的功能
- Generic Programming 是正式語言必要的功能
- Design Pattern 是另一個重要的技術
- STL、boost、平行運算、GPU 都需要
- QuantLib 等大型系統不可能沒有 OO & GP & DP 等功能



## ◆ 另一種選擇，Embarcadero RAD Studio，

- C++ Builder & Delphi(OO Pascal)。
- RAD IDE 易於使用，可開發 Android & iOS 應用。
- 目前有提供社區版，可免費使用，要申請帳號。
- 以前使用過 Borland 產品，很強大。

## ◆ Windows 與 Linux 的選擇

- 銀行的 Workstation 是 Windows 天下，Server 可能是 Linux 的地盤
- 跨平台工具的考慮
- 行動裝置的考慮
- 高運算量的考慮
- GUI & DB 的考慮
- 最後 Tools 的選擇

## ◆ 演算法學習 Road Maps

- Level 1 : BS Vanilla Option
- Level 2 : Exotic Option , Equity Tree , MC Simulation , American MC(LSMC) , Local Volatility Model
- Level 3 : Stochastic Volatility Model(Heston, SABR)
- Level 4 : Interest Rate Simulation(HW 、 G2++) , Stochastic Local Volatility Model(SLV)
- Level 5 : Multi-Factors Forward Rates(LMM)

◆ 程式、系統、軟體產品(The Mythical Man-Month , Frederick Brooks, Jr)

程式	軟體系統 (介面、資料庫、 系統整合)
軟體產品 (通用、測試完成、 文件齊全、可維護)	軟體系統產品

## ◆ 寫程式的樂趣

- 1、創造的趣味
- 2、這東西竟然對別人有用
- 3、打造精巧的機制，類似推理、解謎的過程
- 4、持續學習的樂趣
- 5、在易於操控的工具上(PC)工作的快樂
- 6、滿足內心創造事務的渴望，激發本能快樂的感受

## ◆ 寫程式的苦難

- 1、必須要表現完美
- 2、很少能由自己安排工作目標與細節
- 3、依賴別人的產出才能成事
- 4、除錯大不易，往往有非線性的變化
- 5、可能最後完成時，需求已經改變了