Business and Information Technology Faculty
Research & Creative Works

Business and Information Technology

01 Jan 2000

Computational Finance Models

Michael Gene Hilgers
Missouri University of Science and Technology, hilgers@mst.edu

Recommended Citation

Hilgers, M. G. (2000). Computational Finance Models. *IEEE Potentials* Institute of Electrical and Electronics Engineers (IEEE).
The definitive version is available at https://doi.org/10.1109/45.890082

ike many folks, I have switched from modeling mechanical systems to developing computational finance software with nary a glance back. However, I have found revealing I do this work can be implicitly self-discrediting. Many people, even educated professionals, do not understand the basic premises or possibilities of this emerging new science. For them, unless I own an island somewhere, I don't know anymore about stocks than they do.

They assume I'm writing programs to forecast stock buys and, of course, using this knowledge to personally profit. This brings us to our first fundamental lesson about financial modeling. Little effort is given to stock market *prediction*. In fact, the vast majority of stock market models are premised on the belief the current price reflects all that is known about a stock. Using yesterday's data will not help determine the price of the stock tomorrow. (Of course, in this day of high-powered computing, there are several groups working on stock market prediction. Check out the web site <http://www.predict.com> for an example of a company using nonlinear dynamics theory to do stock market prediction.)

Computational finance models do, however, attempt to model the *randomness* of a stock's price. At a fixed future time, a stock's price is modeled as a random variable with a normal distribution centered about the current price adjusted with a simple growth multiplier. The standard deviation of this normal distribution depends on the length of time into the future one peers and the volatility of the market. As the market becomes more volatile and we look further ahead, the less likely the stock will have a price near the adjusted current price.

Implementing these ideas requires a tool borrowed from physics called the Brownian motion, originally developed to describe the vibration of pollen particles in a liquid solution. In a sense, a stock's price is modeled as a point fluctuating about in "dollar space." Hence a financial modeler can no more predict what price a stock will have at a given instance in time than a physicist can predict where a particular air molecule might be.

Not THAT kind of derivative

Now that we have gone over what financial modeling does *not* do, let us consider what is done. The prototypical example of financial modeling involves a product called a *European call stock option*. It is a contract giving the holder the right to purchase a share of stock at a predetermined price, called the *strike* price, on a given day, called the *expiry*.

The basic function of financial modeling is to determine the price of such a contract. On the closing day this is easy to do. If the price of the stock on the expiry day is S_T and the strike price is K then the value of the option at the close of the contract is $\max(S_T - K, 0)$.

To see this, suppose the trading price of the stock is greater than the strike price. The holder of the option should buy the stock at the strike price. (This process is known as *exercising the option*.) The option holder makes a profit equal to the difference in the two prices. However, if the stock's trading price is less than the strike price, the holder should choose not to exercise the option. It is not smart to pay more than the stock is currently worth, so there is no return on the option.

This contract *derives* its worth from the price of the underlying stock. Hence, it is called a *financial derivative*. The goal of financial mathematics is determining the contract's price at a time *before* the expiry *independent* of the unknown price of the stock on the closing day of the contract.

Doesn't seem possible, does it? The trick is that this contract can be mimicked using other pathways through the financial marketplace, and these paths cost money to traverse. Ultimately, that cost becomes the price of the option.
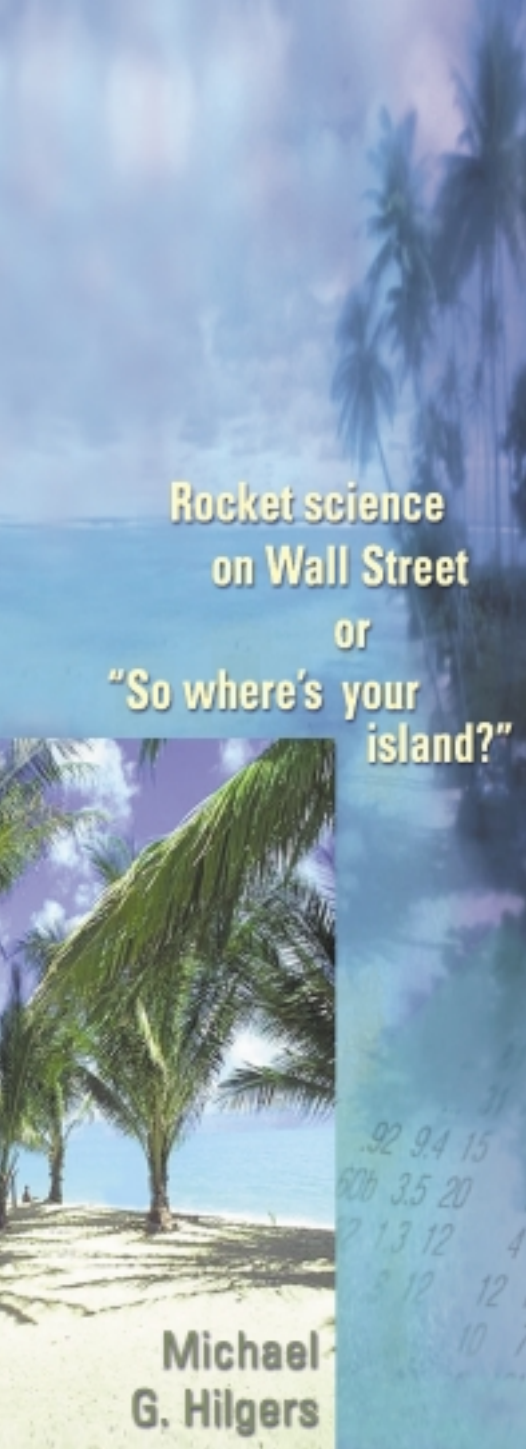
Let us work through a simple demonstration with numbers. Suppose a stock's current price is $100 and that the strike price of the option is also $100. To keep this demonstration simple, we assume stocks are traded at discrete time intervals. In fact, we suppose our option expires the next time we can trade. To make it really simple, we assume the stock can only go up to $110 or down to $95, each with its own probability. Thus, the payoff on the contract will be either $10 if the stock goes up or $0 if it goes down.

Allow me to demonstrate a different path to the same end, namely $10 if the stock goes up or $0 if the stock goes down. Suppose we buy α shares of the stock and borrow β dollars from a bank at an interest rate of 2% for the time period. The cost of doing this is $C_0 = 100\alpha + \beta$.

At expiry, we want this combination of assets to equal the payoff of the option. This means α and β must satisfy $110\alpha + 1.02\beta = 10$ if the stock goes up, and $95\alpha + 1.02\beta = 0$ if the stock goes down. This is two equations in two unknowns and has $\alpha = 2/3$ and $\beta = -62.09$ as a solution. With these numbers we determine our initial cash outlay for this alternative path is $C_0 = \$4.57$.

It is argued for a large marketplace with many investors, the value of the option must be C_0. To see this, suppose you can sell one of these options for more than $4.57 to someone. You should then buy two-thirds of a share of stock by borrowing $62.09 and adding $4.57 from your own pocket. If the

Rocket science
on Wall Street
or
"So where's your
island?"

Michael
G. Hilgers

price of the stock goes down, the holder of the option you sold will not exercise it. You sell the stock on the open market and have exactly enough to repay the loan and its interest.

If the stock's price increases, the holder will exercise the option. You sell your share of stock on the open market and have $10 left after you repay the loan plus interest. With the $100 you receive from the option holder, you have exactly enough to purchase a share at the current $110 level. You give the share to the option holder fulfilling your obligation.

Either way you have made a profit, independent of market behavior, equal to the difference between C_0 and the price you received for the option. Furthermore, there was no risk associated with this profit, which is a clue the price of the option should not be greater than $4.57. Likewise, a similar pathway, also leading to a risk-free profit, can be traversed if the option is priced less than $4.57.

Companies hire people to identify these pathways that mimic options and exploit them to make money without risk. Such people are called *arbitragers*. In a sense, their manipulations are the forces that determine the price of a financial derivative. It is standard to assume a large marketplace will not allow such *arbitrage opportunities* for very long, thereby setting the price of an option at C_0. (This is sometimes called the "no free lunch" principle.)

Models and methods

So then, determining the price of a financial derivative is the basic problem of computational finance. As imaginations have whirled, and the years passed, the menu of derivative securities has grown, but the basic ingredients remain the same. Most derivatives are contracts involving an underlying product that experiences random fluctuations in its price. The underlying product can be anything from pork bellies to home mortgage interest rates. Yet with all of this variety, there are two dominant types of mathematical models used to determine the price of financial derivatives: partial differential equations and expected value integrals.

Partial differential equation models

The argument used by Black and Scholes in 1973 to determine the price of a European call option is fundamentally similar to what we just considered. However, they allowed stock trading to happen continuously with a stock possessing a full spectrum of values, making the mathematics more difficult. They argued the current value of an option is the cost incurred by moving funds from the stock market into the bank and back, as we did. Our one step example required solving a linear system of equations. Theirs required solving a *partial differential equation* (since known as the Black-Scholes PDE).

After a clever set of transformation, this PDE becomes something familiar to the engineering community, namely the *heat equation*, which has a known solution. (I bet you never thought that something you learned in thermodynamics could be used in multi-billion dollar financial transactions. Now where are those notes...?)

Their solution of this PDE has become a modern finance mainstay. The solution is now a standard part of dozens of software packages. For it's discovery, Black and Scholes were awarded a Noble prize.

Finite difference methods

Since research tends to move in a copycat fashion, it is not surprising that a number of Black-Scholes-like partial differential equations have been produced. These models can draw upon decades of engineers' experiences of numerically solving mechanical models using techniques like the *finite difference method*.

Briefly speaking, the finite difference method involves replacing the partial derivatives in the equation with a difference quotient. A system of equations results, and its solution yields an approximation of the unknown at various locations.

There are many subtleties in this process and model complexity soon leads to long computation times. But the years of applying this technique to the equations governing fluid and solid mechanics prepared engineers well for Wall Street. Soon after the work of Black and Scholes became popular, Wall Street greedily "borrowed" experts from the defense and space industries to implement such numerical solvers. Thus, the nickname of "rocket science" for financial mathematics came to be.

Expected value integral models

An alternative model describing the European call option, discovered sometime after Black and Scholes, involves the use of an expected value integral. Recall the expected value of a function $F(X)$ of a random variable X with probability density function $p(X)$ is the integral of $F(X)$ times $p(X)$.

Applying this to the financial setting begins with the observation the value of the stock at expiry, S_T, is a random variable with an assumed probability distribution. One might conjecture that the

price we should charge for a call option is the expected value of its payoff $\max(S_T - K, 0)$.

Philosophically, this is correct, but the devil is in the details, as they say. This formulation requires the use of martingale theory and stochastic calculus. (Career note: People with Ph.D.s in either of these subjects have been extremely popular on Wall Street for about 20 years.) Knowing the price of the option once the integral is formed reduces to performing a numerical integration. Most financial derivative problems can be modeled in this style.

(Quasi)-Monte Carlo method

The Monte Carlo Method is the natural numerical method for approximating expected value integrals. It is a fairly simple algorithm that begins with generating a set of pseudo-random numbers (or vectors) according to a desired probability distribution. Averaging the value of the integrand evaluated at each of the pseudo-random points approximates the integral.

This method has the advantage over quadrature methods of being interpreted as a marketplace simulation. Each random number can be viewed as a possible outcome of the market on expiry. The combined results of each simulated market outcome are averaged for the estimate.

An extension of the Monte Carlo method becoming increasingly more popular in financial modeling is the so-called quasi-Monte Carlo method. The difference between the two approaches is that pseudo-random number generation is replaced with *low discrepancy sequences*. These are a number-theoretic creation that does not attempt to mimic randomness as much as fill space as efficiently as possible. Financial simulations performed with low discrepancy sequences have witnessed dramatic increases in performance. As a result, Wall Street is now looking for number theorists.

So many derivatives ... so little time

The volume and variety of complex financial derivatives being handled by trading firms is enormous. Thus, it is little wonder the time required to determine the price of their products has become a limiting factor for their business. In fact, the financial world's situation is much worse than that of the physical sciences community. Tidal waves of new information inundate ana-

lysts. Decisions must be made quickly. Programs need to provide answers in "real time." Computing horsepower can go only so far to remedy the situation. To demonstrate the challenges being faced in the industry, here are a couple of examples I have encountered in my consulting and research. The model used for all of these is the expected value integral with a (quasi)-Monte Carlo simulation.

Equity-indexed annuity: Many insurance companies sell an annuity product that makes a payment based on stock market growth. Normally, they average a stock index, such as the S&P 500, at month intervals over the course of the year. Then, using the value of the index at the beginning of the year as a strike price, they calculate the rate of payment on the annuity in a fashion like the payoff on a call option. In the parlance of the financial markets, this is known as an *embedded Asian option*. The product is called an *equity-indexed annuity*.

We have used both traditional and quasi-Monte Carlo methods to estimate the current value of such an annuity. This requires using a pseudo-random vector containing the 12 monthly index value. Typically, the number of random vectors required is on the order of 50,000 for the traditional approach and 20,000 for the quasi-Monte Carlo approach. Since annuities are sold daily, insurance providers are faced with generating a total of 10^7 vectors for the 365 different options they maintain. Using spreadsheets and Visual Basic, these calculations took over 24 hours for one company. This strongly limited their business practice and demonstrates the kind of challenges present in computational finance. Recently, I implemented the quasi-Monte Carlo method using C++ programs and reduced the computation time to 10 minutes. (Most every job has the right tool...)

Stochastic interest rate bonds: Bonds are a classic valuation problem. We sell them at a price today based on what we want them to be worth years from now. If the bond's interest payments are based on the randomly fluctuating interest rates of the US Treasury, this is a tough problem.

To implement the Monte Carlo method, we need an interest rate point in the random vector for each day (or week, as reasonable) in the life of the bond. This results in a random vector containing thousands of points. Also, the Monte Carlo approximation may require a mil-

lion of these vectors to converge. Hence, the valuation of a single bond may run for hours on a workstation. Recent simulations done with the quasi-Monte Carlo has shown that the number of random vectors required can be reduced to two orders of magnitude, thus, the valuation can be done in minutes. This is much closer to real time trading.

Cash flow through insurance companies: Consider this situation faced by insurance companies every year. They must demonstrate they will have the money to fulfill the obligations of their policies. This balances the return on their investments against the mortality of the customers over the next 30 years. Since most of their investments are in a variety of bonds, they need models of the coupled short-term and long-term interest rates to estimate their return on investment. To complete the simulation, actuaries must provide models of the life expectancies of their clients.

The Monte Carlo method can simulate their solvency, but here's the catch. The combined set of models for a billion-dollar company is so involved that the simulation can process only one hundred interest rate vectors in a 24-hour period. Letting the simulation run for days is impossible because of network considerations. Using only 100 sample paths simply does not produce a tight estimate. Quasi-Monte Carlo methods have helped, but this remains an example of a computational problem for which there is no practical solution . . . yet.

Read more about it

• Paul Wilmott, Sam Howison, and Jeff Dewynne, *The Mathematics of Financial Derivatives: A Student Introduction*, Cambridge, 1995.
• Y. K. Kwok, *Mathematical Models of Financial Derivatives*, Springer Finance, 1998.
• Don M. Chance, *An Introduction to Options and Futures*, The Dryden Press, '89.
• Marek Musiela and Marek Rutkowski, *Martingale Methods in Financial Modelling*, Springer, 1997.

About the author

Michael G. Hilgers is an Associate Professor of Computer Science at the University of Missouri-Rolla. He holds his Ph.D. in Applied Mathematics from Brown University. He wrote a course in financial modeling, and, recently, he has been acting as a consultant developing software with application in the insurance industry.