CONTENTS

Welcome to the Help system and courseware for Mantic Software's Binomial Market Model.

If you don't know how to use the Microsoft Windows hypertext Help system, use the mouse to click the word **Help** in the main menu at the top of this window. Choose **How to Use Help** from the list that drops down.

Please register your copy of this program. Registration is free, and Mantic will send registered users information about new products as they become available.

To activate context-sensitive help when running the simulation, press the [**F1**] key on your keyboard.

Choose a topic from the list below, or click the **Search** button with the mouse. You can also use the buttons labeled >> and << to proceed through screens of related material in an orderly sequence.

Mantic Software Corporation

Mantic Software Corporation produces software and books that teach the principles of finance and investment. Click here: Options Laboratory to find out about *Options Laboratory*, the complete, interactive environment for learning to use stock and index options effectively!

The *Binomial Market Model* program you're running now is "freeware" that you may copy and redistribute*, provided that all the files on the distribution diskette are reproduced, complete and intact.*

Please register your copy of this program. Registration is free. Mantic has an active R&D program, and we will inform registered users as new products become available. To register, send us your name and postal address via:

Internet mail: mantic@csn.org
Telephone: 303-224-1615

Mail: Mantic Software Corporation
1523 Country Club Road
Ft. Collins CO 80524

We also welcome your comments, questions, and critique. The distribution disk and Windows program group for *Binomial Market Model* both contain a short file called COMMENTS.TXT that you can edit to send us your thoughts by E-mail or on paper. To edit and print COMMENTS.TXT, double-click the file name or Windows icon with the left mouse button.

Sales information only: 800-730-2919

Please **DO NOT** call these numbers to inquire about Binomial Market Model. Use Internet mail or Mantic's company telephone number, listed above for that purpose.

Options Laboratory

Options Laboratory is an interactive, exploratory environment for learning how to invest effectively with options. This program offers state-of-the-art modeling and risk-analysis tools in an elegant, easy-to-use graphical environment designed for exploration and experimentation. It comes with ***A Course in Options***, a textbook customized to the *Options Laboratory* software.

Click here: ordering information to find out how to order *Options Laboratory*.

The course covers everything from basic theory to cost reduction strategies and sophisticated hedging techniques, with an emphasis on analytical tools and risk analysis. Everything is explained in "plain English", using *Options Laboratory's* interactive graphics to give you real insight. You'll learn more from a week with *Options Laboratory* than from two months studying financial textbooks. Mantic's unique Animation feature even shows graphically how your investment position will evolve as time passes and the options approach expiration.

Options Laboratory is also a tool for the real world. It can analyze any position with up to four options and the underlying stock or index. Use it to analyze and track actual investments with standardized (exchange listed) option contracts. The program has a Basic mode for quick setup and experimentation, so you can test possible strategies with just a few mouse clicks. In Advanced mode, use its analytical tools to decide whether to continue holding a position, close it out, or adjust it.

The software correctly evaluates European- and American-style options, including complete early exercise analysis -- it even identifies the early exercise threshhold price for options, so you can plan beforehand for the possibility of early exercise. All results are reported in dollars-and-cents terms. For example, rather than just telling you the value of each option's "delta" parameter, *Options Laboratory* interprets the parameter to tell you exactly how the whole position and each component will react when the underlying stock or index changes price.

You'll be amazed by the program's probability-weighted analysis, which estimates the likely outcome using sophisticated statistical techniques. It's a super way to compare alternative strategies. And *Options Laboratory* even allows you to enter your own estimate of market drift (trend or direction), which is then applied in the statistical analysis.

You can see some sample *Options Laboratory* screens by choosing the **OptionsLab** menu item in the Binomial Market Model main window. Click here: ordering information to find out how to order *Options Laboratory*.

Options Laboratory ordering information

Options Laboratory costs $129.95, plus $5.00 shipping and handling. We accept Visa, MasterCard, the American Express Card, and dollar-denominated checks. To order, call:

1-800-730-2919 toll-free in the U.S.
1-303-224-1615 outside the U.S.

International orders are welcome ($10.00 shipping and handling outside the continental United States).

University site licenses are available at a substantial discount; please contact Mantic at 1-303-224-1615 for details.

Options Laboratory: Main window

While *Binomial Market Model* demonstrates the mathematical basis of option valuation, to explore the practical investment applications of options you need *Options Laboratory,* which isn't part of the software you're running right now.

Binomial Market Model includes three windows to illustrate *Options Laboratory*. You can see them by choosing the **OptionLab** menu item in the main window. The three windows are **Position setup**, **Projection**, and **Share equivalent risk**.

[Please note, these windows are illustrations. They don't "do anything". Click here: Options Laboratory for an overview.]

Options Laboratory allows you to quickly set up any position involving the underlying security or index, and up to four options. You enter the risk-free interest rate, current price, and volatility of the underlying in the top panel of the window, by dragging "scroll bars" with the mouse. If shares of the underlying good are part of the position, you enter the number bought or sold (long or short) by typing or clicking on a list of common contract numbers.

In the upper right corner of the window you can choose European or American style option contracts. (The program can perform complete early-exercise analysis, taking dividends into account if any are scheduled.)

Then enter up to four option specifications. *Options Laboratory* "knows about" standard option contracts, so you can simply choose expirations and strike prices with the mouse; or you can enter non-standard expirations and strikes. Set up each option position with the number of contracts involved. The example shows a calendar spread with four contracts on each leg.

Now you can click **Project** to see the position's future value at any time until the first options expire.

Options Laboratory: Projection Window

[The window described below illustrates the *Options Laboratory* software. You can access it by choosing the **OptionLab** menu item in the main window. It's an advertisement! It doesn't do anything! Click here: Options Laboratory for an overview.]

... Options Laboratory's Projection window allows you to see the future performance and other characteristics of your option position. You can see the combined (Net) characteristics of the entire position, and also look at each individual option.

The simplest plot gives the value of the position and its components as a function of stock price, over a plus-or-minus three standard deviation price range. You can click any point on the graph with your mouse to read the exact value conveniently.

When you click the **Animate** check box, *Options Laboratory* will display a series of graphs over time. This lets you see how the position will evolve. For instance, calendar spreads show a progressively widening price range over which they will turn a profit as time passes. If the stock moves against you early in the life of the calendar spread, you may have a temporary loss with a good probability of becoming profitable later.

The program can also graph the probability-weighted outcome of the position. This plot, not illustrated, multiplies the payoff at possible future stock prices by the log-normal probability of those stock prices being achieved. Thus, very large price movements are given less weight than smaller movements over the projected timeframe.

When American options are evaluated (with or without dividends), the program also flags the *early exercise threshhold price* (not illustrated). This is the first stock or index price at which it becomes economically rational to exercise the option before expiration.

All projections can be biased by adjusting the volatility used to make projections, so you can explore situations where the volatility is expected to change. Projections can be made using the underlying security's volatility, or the implied volatility of each option. And since volatility is a trendless statistical measurement, the program also allows you to enter your own estimate of *market drift* as a continuously compounded annual percentage rate.

Options Laboratory: Share-equivalent risk

[The window described below illustrates the *Options Laboratory* software. You can access it by choosing the **OptionLab** menu item in the main window. It's an advertisement! It doesn't do anything! Click here: Options Laboratory for an overview.]

... Another *Options Laboratory* analysis gives the position's *share-equivalent risk*, a form of delta analysis. This shows exactly how much risk the position incurs for small, instantaneous changes in the price of the underlying good.

Options Laboratory interprets the options' delta parameters in the actual position you've set up, weighting the deltas by the number of long and short contracts. This gives a direct, quantitative measure of the impact of short-term price movements. It's important to realize that under some circumstances, fairly small stock price movements can transform a bullish position into a bearish one, or *vice-versa*.

As you can see from the illustration, the analysis can plot results for the combined position (all options plus underlying security), and each component option. The share equivalent risk analysis can be animated over time, like all other *Options Laboratory* analyses.

Program description

Binomial Market Model uses software as a teaching vehicle. The program explores some sophisticated financial ideas in an experimental software laboratory setting. We created this simulation mainly for fun, and to learn something ourselves, but it was one of those tales that grows in the telling. Friends found it interesting, so Mantic Software is making it available as freeware.

Unlike Mantic's other products, this one is entirely "on-line" - it doesn't come with a printed textbook or manual. We'd like to hear your reaction to this on-line learning experience. After you try the program and read the associated article, please let us know how you liked them. Click the company name Mantic Software Corporation to find out how to contact us.

Binomial Market Model simulates the market price of one stock in a market as it changes over time. The simulation is described and analyzed in *A Model of Runs and Cycles in Price Data*, which is linked as hypertext into this Help file. You can use the **File | Print** menu item of the Help system to print the article.

This simulation is based on the well-known binomial model of price movements, a statistical method used extensively in mathematical methods for computing the fair value of call and put options. It shows how (and why) the binomial model inevitably generates trends, price runs and cycles that appear remarkably similar to actual market price movements. Using this program with the article mentioned in the previous paragraph, you'll gain a better understanding of the real meaning of stock price volatility.

The program plots the generated price series along with short- and long-term moving averages. It also accumulates probability distributions of the generated prices and compares them to financial theory. It displays the binomial price movement map with associated probabilities, and computes the likelihood of price runs of various lengths.

Program windows

Action in this program happens in the Main and Plot windows.

Main window

The Main window is divided into the Simulation panel and the Model parameters panel. The Simulation panel is used to control the speed and time step size of the simulation. The Model parameters panel is used to set the key statistical parameters.

To generate a price series, all you have to do is make sure the **Price series** check box is marked in the Simulation panel, then click the button labeled **Proceed**. The program will begin generating prices. Let it run a while, then click the button labeled **Plot** (it is the Proceed button, relabeled) to see the graph.

To accumulate a probability distribution, mark the **Distribution** check box in the Simulation panel, then click **Proceed**.

Plot window

The Plot window is used to display all graphical output. The two basic graphs are the price series and probability distribution.

The **Other graphs** menu item in the Plot window allows you to display four additional graphs:

- Binomial price map
- Runs vs other paths
- Probabilities of long runs
- Binomial option valuation

You can use the mouse to directly read values from any graph drawn in the Plot window.

Write data menu item

You can save price series data to a disk file. To do this, first display the price seriesgraph in the Plot window. Next, use the mouse buttons to select the first and last points you want saved by clicking the desired graph points. Then select the **Write data** menu item, which will ask you to provide a file name.

The data is written as an ASCII text file (the sort of file you can edit with the Microsoft Windows Notepad accessory, or any other text editor). Each price sample is on a separate line. If the Simulation control panel was set for one price step per sample, each sample point contains the simulated time and the price. If more than one price step is generated per sample, the data lines contain: simulated time, final price, low price, and high price during the time period.

The time stamp is written as a floating point (decimal) number indicating the simulated day and fraction of a day. For instance, 120 minutes into day 15 would be recorded as 15.500 since there are 240 simulated minutes per day. See Minutes per sample period.

Runs vs other paths graph

To produce this graph, the program considers all 2^n possible paths through a 1000 move binomial price map. Remember, a *run* is a path that stays entirely in the blue (or entirely in the red) area of the price map. This blue line of this graph shows the proportion (the fraction) of paths which are runs, out of every possible path through the map. The green line shows what's left - the fraction which are paths that wander. The sum of the two lines is 1.0.

You can use the mouse to directly read values from any graph drawn in the Plot window.

See also Price runs.

Option valuation graph

The program can illustrate how to calculate the value of a call or put option with the binomial method of Cox, Ross and Rubenstein. Activate this demonstration using the **Other graphs | Option value** menu choice in the Plot window.

A pop-up window will appear, where you select American or European exercise style and the striking price of the option. The demonstration always assumes there will be 15 price moves until the option expires. The duration of each move is controlled as usual, by the binomial parameters in the Main window.

The program first shows the value of the option at expiration, listing the option value at each terminal price of the price map. Then, each time you click the **Proceed** button, the model backs up one price step and propagates the price backward. The mathematics of this back-propagation is explained briefly in the article

Probabilities of long runs graph

To produce this graph, the program considers all 2^n possible paths through a 1000 move binomial price map, and calculates the probability of reaching each terminal node at the right edge of the graph. It adds up the probabilities along all the paths that lead to each such node.

The blue line shows the probability of *positive* runs (that stay above or equal to the initial price) with length at least *n* for values of *n* on the X-axis. Why "at least" *n* ? Because any run that reaches the right edge of the price map at a price higher or lower than the initial price, requires a least one more move to fall back to the initial price.

The red line shows the corresponding probabilities for *negative* runs. The lines don't coincide unless the probability of an "up" move exactly equals the probability of a "down" move. That only happens under special circumstances. You can adjust the values of risk-free interest rateand volatilityin the Model parameters panel to see when this happens.

The probability of a run with length exactly *n* is just the probability of a run *n* or longer, minus the probability of a run *(n+1)* or longer. Using this fact, the program figures the *expected run length* for positive and negative runs. Although the expected run length is only about 23 in most circumstances, note that the probability of a run of length 1000 or longer is not much less than the chance of a run 500 or longer. Quite long runs are surprisingly likely in the binomial process.

You can use the mouse to directly read values from any graph drawn in the Plot window.

For more on this topic, refer to How probable are long price runs?

Mouse / hairlines

You can read values from graphs drawn in the Plot window. Select the point whose coordinates you want by placing the mouse "sprite" over it. Then click either mouse button. The X- and Y-values will be displayed in the upper left area of the Plot window.

Binomial model

The binomial model of stock price movements was developed by Cox, Ross and Rubenstein as part of a method for computing the "fair value" of stock options. The idea is that the stock price moves in small increments from its current price. For each price movement, we flip a biased coin to decide whether the next price will be higher (with probability prob[Up]) or lower (probability prob[Down]).

If the price is to rise, it increases by a specified small percentage - in other words, it's multiplied by a number slightly greater than 1.00. If it's to fall, it decreases by a specified small percentage - it is multiplied by a number slightly less than 1.00.

The up/down probabilities and multipliers are fixed by the stock's price volatility and the risk-free interest rate. You can see and adjust these values in the Model parameters panel of the Main window. Refer to Binomial equations.

This sequential price change process is an example of what mathematicians call *continuous diffusion*. Such processes are characterized by incremental change with "no surprises". In this case, no surprises means "no unexpected returns": the expected return for owning the stock is determined entirely by its price volatility and the interest rate. Binomial price movements can be visualized with a price movement map, which you can display using the Other graphs option in the Plot window.

Refer to the article for a complete discussion.

Binomial price movement map

The price movement map is drawn when you select the **Other graphs** menu item in the Plot window. The map represents all the possible sequences of up-and-down price movements over 15 steps - in other words, all the possible paths the stock price can take in 15 steps. There's nothing special about 15, it's just a span chosen for illustration.

The initial price is shown at the left. Each move carries the price up or down, and to the right. There are 2^n possible paths through a map of length *n* moves. As described in the explanation of the binomial equations, an up move followed by a down move returns the price to its original value. This gives the map its characteristic "fishnet" appearance.

Green *probability bars* at the right edge of the graph depict the probability that the stock will end up at each end-price after 15 moves. You can see that these bars have the characteristic skewed-bell shape of a log-normal probability distribution. In fact, the distribution is almost log-normal, but not quite. It is skewed somewhat by the effect of interest rate in the binomial equations.

These bars are not scaled so you can read their values meaningfully with the mouse; but the area under the whole green curve adds up to 1.0 by definition.

For more on this topic, refer to Figuring the probability of each path.

Any path staying entirely within the blue, upper region of the map (or entirely within the red, lower region) is called a *run*. Paths that cross the red/blue boundary one or more times are called *wanderers*.

Simulation panel

This panel lets you control the time behavior of the simulation.

The Real time (milliseconds/tick) scroll bar governs how fast the simulation progresses in human terms. The simulation proceeds in discrete steps called *ticks.* At each tick, one or more simulated events (price changes) may occur. If the scroll bar is set to 50 ms/tick, 20 events will happen each second of human time (as measured by your wristwatch). Ordinarily, the next binomial price is generated at each tick event. You can slow things down with this scroll bar.

A trading day consists of 240 minutes (4 hours). There is no time gap between trading days, and trading takes place 365 days per year.

The model generates *price samples* as the simulation proceeds. The Minutes per sample scroll bar controls how much simulated time passes between price samples. If **Minutes per sample** is set to 240, the simulation will record a price sample once per day (the closing price). If it is set to 10, there will be a sample every 10 minutes, making 24 per day.

The simulation can generate from one to 500 prices in the time between samples. This is controlled by the Steps per sample scroll bar. When more than one price is generated between samples, the program records the low, high and closing prices.

See also: Warp speed
Price series mode
Distribution mode
Plot / proceed button

Model parameters panel

The two key parameters of the simulation are the stock's price volatility and the current risk-free interest rate. These may be adjusted by two scroll bars in the Model parameters panel. When you change these scroll bars, or the Minutes per sample and Steps per sample scroll bars in the Simulation control panel, the program recalculates the probabilities of up and down movements and their magnitudes from the Cox-Ross-Rubenstein binomial equations.

If the simulation is running in Price seriesmode, it simply continues with changed parameters. In Distribution mode, the accumulated distribution is reset whenever the model parameters are changed.

See also: Moving averages (STMA and LTMA)

Warp speed check box

The Warp speed check box is another device for controlling simulated time. When it is checked, the simulation clock advances from one event to the next without wasting any human time in between. When this box is not checked, every tick causes simulated time to advance by one simulated minute, even if no events happen at that tick. This slows things down considerably.

Generally, you will want to leave this box checked.

Real time scroll bar

The Real time (milliseconds/tick) scroll bar governs how fast the simulation progresses in human terms. The simulation proceeds in discrete steps called *ticks.* At each tick, one or more simulated events (price changes) may occur. If the scroll bar is set to 50 ms/tick, 20 events will happen each second of human time (as measured by your wristwatch). Ordinarily, the next binomial price is generated at each tick event. You can slow things down with this scroll bar.

Minutes per sample period scroll bar (or Days per sample period)

A trading day consists of 240 minutes (4 hours). There is no time gap between trading days, and trading takes place 365 days per year.

The model generates *price samples* as the simulation proceeds. The **Minutes per sample** scroll bar controls how much simulated time passes between price samples. If **Minutes per sample** is set to 240, the simulation will record a price sample once per day (the closing price). If it is set to 10, there will be a sample every 10 minutes, making 24 per day.

In Distribution mode the time scale of this scroll bar changes to days per sample, because we typically are interested in the probability distribution of prices over long time spans.

Steps per sample period scroll bar

The simulation can generate from one to 500 prices in the time between samples. This is controlled by the **Steps per sample** scroll bar. When more than one price is generated between samples, the program records the low, high and closing prices.

Volatility scroll bar

The stock's price volatility is the most important statistical parameter of the simulation. It is defined as the standard deviation of the logarithm of the daily rate of return on the stock. A mathematical review is provided in the article.

Volatility is always measured over time. When plotting a price series, *Binomial Market Model* also calculates short-term and long-term volatilities as described in the article, using the moving-average time periods specified in the Model parameters panel. When Steps per sample is set to 1, the short- and long-term volatilities come out very close to the volatility scroll bar setting. If more price steps per sample are generated, the two volatilities often differ substantially.

Risk-free interest rate scroll bar

The risk-free interest rate figures into the Cox-Ross-Rubenstein equations because they take into account possible arbitrage among a call or put options, shares of stock, and risk-free bonds. The risk-free interest rate is typically considered to be the yield on short-term Treasury securities, or the yield on money-market mutual funds.

See An example of the binomial formulas and Binomial prices are not quite log-normal.

Moving averages (STMA and LTMA)

The program can compute short-term and long-term moving averages. The time period of each average is specified as the *number of samples* over which the average is taken, not the number of days! This is appropriate because of the fractal nature of the binomial model.

To specify STMA or LTMA, you type the number of samples into either box and press the [Enter] key.

If the number of Minutes per sample is 240 (one simulated 4-hour trading day), then the short-term and long-term moving averages are in days.

Reset button

The **Reset** button returns the program to its initial state, except that the random number generator is not re-initialized. Thus the generated price series will be different.

Proceed / Plot button

This button is accessible in both windows. When it is labeled **Proceed**, clicking it causes the simulated time to proceed. When it is labeled **Plot**, clicking it with the mouse pauses the simulated clock and plots the price series or probability distribution based on current model parameters.

Price Series and Distribution check boxes

When the **Price series** check box is marked and you click the Proceed button, the program will generate a series of binomial prices. You display the price graph by clicking the Plot button.

When the **Distribution** check box is marked and you click the Proceed button, the program will start accumulating the probability distribution of generated prices. To see a graph of the cumulative probability distribution function, click the Plot button.

The "smoothness" of the distribution depends on the number of price steps per sample. If there is only one step per sample, only two prices are possible: the current price times the "up" multiplier, and the current price times the "down" multiplier. The distribution function graph will reflect this with a two-level "staircase" shape.

If you select more steps, the distribution function becomes smoother. At 500 steps per sample, the cumulative distribution becomes quite smooth after 1000 samples or so.

Notice that the accumulated distribution differs systematically from the blue, log-normal cumulative distribution function that is also drawn in the graph. The experimental distribution coincides with the log-normal one only when the probability of up and down moves are equal. You can adjust the Volatility and Interest rate scroll bars to demonstrate this. Watch the calculated values in the Model parameter panel as you adjust these bars with the mouse.

How to print topics from this file

You can print any topic from this file whenever it is displayed on the screen. With your mouse, choose the **File** menu item in this window. Select **Print topic** from the sub-menu that drops down.

A Model of Runs and Cycles in Price Data

You can proceed through the pages of this article in sequence, by clicking the buttons labeled >> and << above this window. You can also print pages if you prefer to read a paper copy.

Due to a defect in the Microsoft Windows Help subsystem, the text of formulas and equations displayed in this article are typeset into positions slightly above the lines where they should appear. This is not a problem with your computer system, nor with the Binomial Market Model. At the time this software was released, there was no known way to avoid the problem.

Article contents

Introduction

This article and the associated software are designed to give you some practical insight into the *binomial model* of stock price movements. The binomial model is presented in every finance text that covers the theory of option valuation. It's the most widely used method for computing the "fair value" of American-style options (contracts permitting the holder to exercise them before expiration).

Its theoretical foundation is the famous "random walk" hypothesis, but textbooks don't present it from that perspective. I thought it would be instructive and entertaining to see and test the real implications of this model, so I wrote some software to help visualize the workings and statistical properties of a binomial market.

It's a simple model, driven by a uniformly distributed random number generator, with straightforward statistical properties and no market logic or intelligent trading agents. Yet it generates displayed remarkably market-like price behavior, and not just in broad statistics. Long- and short-term price runs and cyclical movements appeared plainly in the graphs, which are much more realistic-looking than I'd expected. As Figure [1]illustrates, even the long- and short-term moving averages have a surprisingly realistic appearance, at least to the "naked eye". Why do price series derived from random numbers exhibit so much apparent structure?

The answer is that the binomial model *inevitably* produces a kind of non-periodic, cyclic price motion. Prices in a binomial series are more than random points with a certain statistical distribution, even though a random number generator helps produce them. Like real market prices, each binomial price is related to other, recent prices, and carries an implicit forecast about the rate of change and size of future price movements.
Binomial cycles result from a particular *process* of price movements, and the binomial process is a reasonable description of some aspects of real markets. To the extent that it reflects reality, it must contribute to the cyclical patterns we see in real stock prices. It may even be possible to distinguish this inherent, non-periodic cyclicality from meaningful changes in a stocks price behavior.

One conclusion I personally take away from this mathematical experiment is that statistics by themselves are weak tools for characterizing market behavior. To *really* understand markets, we will have to model and understand the underlying processes which produce the price movements we observe.

What is "volatility"?

Binomial price movements depend on two parameters: the *risk-free interest rate* (the rate earned on loans with essentially no risk of default, eg. the money-market mutual fund rate), and the stock's *price volatility*.

Volatility, denoted by the Greek letter σ , is familiar to option traders as a measure of how much a stock's price is likely to change over time. It actually measures how much the daily rate of return deviates, on average, from its average value. Statisticians describe this as the *dispersion* of the daily returns about their mean. If the daily rate of return were the same each day, the volatility of the price series would be zero.

This is clearer if we look at some formulas. Mathematically, σ is the standard deviation of the *logarithm* of the daily rate of return on a stock. From a sequence of daily prices $P_1, P_2, P_3 \ldots P_n$ we calculate each daily rate of return as today's price divided by yesterday's price, $R_{i+1} = P_{i+1} / P_i$. This is just the relative or percent change in stock price from each day to the next. Then we take the natural logarithms of these numbers, $X_i = \log_e(R_i)$, and compute the standard deviationof the series X_i in the usual way:

$$\sigma = \sqrt{\frac{\sum (X_i - \overline{X})^2}{n - 1}}$$

where $\overline{X}$ is the average of all the X_i

This daily σ can stretched mathematically to span any time period - an hour, a month, or a year - but it's conventionally stated as an annualized value. According to statistical theory, if a normally distributed (Gaussian) random variable has standard deviation σ over time *t*, then its standard deviation over time *T* is $\sigma \sqrt{T / t}$.

Binomial price movements

To create binomial price movements, we essentially reverse the process used to compute s . Whenever we want to know the "next" price, we flip a biased coin to decide whether the next price movement will be up or down. The new price is calculated by multiplying the current price by an "up" percentage or a "down" percentage:

$P_{i+1} = u \times P_i$ if the price moves up

$P_{i+1} = d \times P_i$ if the price moves down

You can see that *u* and *d* are just conditional, periodic rates of return; they're special cases of the daily return P_{i+1} / P_i used to calculate
s . The trick is choosing a bias for the coin that produces the correct proportion of up and down movements, and values of *u* and *d* so that the standard deviation of the generated series
$P_1, P_2, \ldots P_n$ actually converges to
s if we generate a long enough price series.

Cox, Ross and Rubenstein showed how to choose these values so that in the long run, the distribution of binomial stock price movements is almost log-normal. Then the values of $\log(P_{i+1} / P_i)$ naturally distribute into a classical bell-shaped curve, like the grades of students in a large class, and have the desired standard deviation. The C-R-R formulas are:

$u = e^{s \sqrt{t}}$	the "up" multiplier
$d = 1 / u$	the "down" multiplier
$p_u = \dfrac{(r - d)}{(u - d)}$	probability of an "up" movement
$p_d = 1 - p_u$	probability of a "down" movement

Physical and economic meaning of binomial parameters

Why not exactly log-normal? The model assumes traders can freely perform arbitrage among mispriced puts and calls, stock, and risk-free bonds; so the risk-free interest rate *r* over the time between consecutive binomial prices figures into the equations. Thus *r* and sigma combine to skew the distribution slightly away from true log-normal; the binomial series converges exactly upon log-normal only in the special case when $p_u = p_d$.

We should take a moment to think about the physical and economic meaning of the binomial parameters.

The up multiplier *u* predicts that if the stock price makes an up movement, it will rise by an incremental amount related to time and volatility. The expression $u = e^{\sigma\sqrt{t}}$ actually has its roots in physics of "Brownian motion", not statistics or finance. Brownian motion is the motion of gas molecules as they travel randomly through space, having their direction of motion altered by chance as they bang into each other. It's the original random walk! Einstein studied this motion and showed that on average, the distance traveled in a random walk along a single dimension is characterized by
$e^{\sigma\sqrt{t}}$. In physics, temperature is a key factor determining the value of sigma.

Students of price trends and persistence say that when the actual values of *u* and *d*, and the probabilities match this physical model, the price movements are a random walk. If the distance traveled over time is less than $e^{\sigma\sqrt{t}}$, the price series is said to be anti-persistent. If the distance is greater, the series shows more persistence (a greater tendency to sustain trends of direction) than would be consistent with a random walk.

The risk-free rate of return *r* specifies how far the price must move *consistently* to provide the same rate as a risk-free bond. In other words, if the stock moved up by a proportion $(1+r)$ on every single move, its return at the end of the process would be exactly the risk-free discount rate. Repeated upward price motions compound the risk-free rate at each price movement, so that after *n* price movements the compounded result will be
$(1+r)^n$.

If the return for owning a stock, given that it goes up, doesnt exceed the risk-free rate, a rational investor would always prefer to own a risk-free bond that pays better, so the model requires that $u > r$. Similarly, if the return for owning the stock, given that it goes down, exceeds the risk-free rate, nobody hold the bonds. Thus for the model to make economic sense, we require that
$u > r > d$.

How realistic is this model?

A number of academic studies have measured whether actual stock price movements have log-normal distributions over long time spans. In general, these studies conclude with a qualified *yes.* The humped shape and long tails of a log-normal distribution, illustrated by the green bars at the right side of Figure [3], form a pretty good statistical picture of stock price movements for a large universe of stocks. The tails (the price extremes outside the main hump) of actual stock price distributions tend to be "fatter" than theory predicts, and the hump not quite as large, so real prices show some systematic deviation from log-normality.

In other words, the probability of extreme up or down movements is greater than we'd expect from theory. The usual interpretation of this observation is that price volatility is not "stationary" - it occasionally changes (jumps to a substantially different value) when a company's prospects are suddenly perceived to change. The basic binomial model doesn't capture these volatility jumps, and while there are several plausible ways to extend it, nobody has shown conclusively that jump processes actually produce better results. In fact, some option pricing studies have come to the opposite conclusion.

On the other hand, it's also reasonable to be skeptical about such studies. Statistical characterizations have their value, but there are other skewed, long-tailed distributions that may be plausible candidates to describe market price movements. Fat humps and long tails tell us plainly that actual price movements are either statistically persistent, or antipersistent, compared with the random walk hypothesis.

From a trader's point of view, the binomial model is a reasonable one under "calm" circumstances. When buyers and sellers arrive at market in a more-or-less random order, typically when there's no news about the stock, we'd expect roughly random sequences of up and down ticks. We expect clusters of buyers or sellers to drive the price up or down only when something noteworthy happens. Even then, market specialists are expected to use their inventory and capital to maintain smooth, incremental stock price changes.

It's fractal

The binomial model is *fractal* - it has similar behavior and statistical properties at any time or price scale. Its fractal with respect to price because, at each step, the current price is multiplied by *u* or *d,* changing it in proportion to its magnitude. Its fractal with respect to time because interest earnings and the size of up and down moves are scaled by the factor $e^{r\sqrt{t}}$. The model also preserves its internal consistency even if the stock price drifts very high or low after a long series of movements.

By implication, any conclusions we draw from the model should hold true whether it's applied over short, medium or long time spans. In the real world, that just isn't true. For instance Malkiel's popular book, *A Random Walk Down Wall Street*, presents data showing that the expected return for holding stock more than five years is much better than the return for short holding periods.

An example of the binomial formulas

We saw earlier that the standard deviation statistic contains no information about price trends. To explain why price trends appear in binomial series, we must examine the specific sequences of up-and-down movements the binomial process can generate, and estimate their probabilities. Thats the only way to really understand whats going on in Figure [1]. Since the model is fractal, our analysis of price movement sequences will apply at all time and price scales.

s in the C-R-R equations is an annualized standard deviation; *t* is the time between consecutive price samples, stated as a fraction of a year; and *r* is the interest rate over time *t*. If the risk-free rate is 4% and our simulated stock has annual volatility of 0.15, then for daily price movements:

$t = 1/365 = 0.002740$	one day as a fraction of a year
$r = (1+0.04)^t = 1.000107$	daily return on risk-free bonds
$u = e^{s\sqrt{t}} = 1.007882$	magnitude of movement if price rises
$d = 1/u = 0.992179$	magnitude of movement if price falls
$p_u = 0.504880$	probability that the price will rise in this time
$p_d = 0.495120$	probability that the price will fall

Suppose the stock's price is 100 dollars at time t_0. The next "moment" in the series, time t_1, is one day later, when the price will be either $P_1 = 100 \times u = 100.7882$ (with probability 50.4880%) or $P_1 = 100 \times d = 99.2179$ (with probability 49.5120%). If the price goes up at step t_1 then falls at t_2, price $P_2 = P_0 \times u \times d = P_0$ since $d = 1/u$.

Over time, this process carries the stock price through an arbitrary series of up and down movements like *UUDDUDUDUDDD....* Figure [3] illustrates this by mapping out every possible sequence of up and down movements for 15 steps from an initial price of 100, when $s = 0.15$, the risk-free rate is 4%, and the price steps are 30 days apart. Since there are two possible moves (up or down) at each time step, there are 2^n possible paths of *n* steps through the map; but the reciprocal relationship of *u* and *d* means there are only *n* possible prices the stock can have after *n* price movements.

Figuring the probability of each path

The green bars at the right end of the price map show the probability of each price outcome at the final step. Knowing that the probability at each step is either p_u or p_d, we can explore all the possible paths leading to each terminal node of the map and accumulate the probabilities along each path as we proceed.

A better way to find these path probabilities is to observe that if a node at the end of an *n*move path has *j* more "up" moves than "down" moves, the number of possible paths to that node must be $C(n,j)$, the number of combinations of *n* things taken *j* at a time. This is true because the order of the moves is irrelevant: *UUUDDUD* gets to the same place as *DDDUUUU*, or any other combination of four "ups" and three "downs". Each such path to the end has *j* up moves with probability p_u, and $(n-j)$ down moves with probability p_d, so the total probability of reaching that particular end node is the probability of that price outcome, times the number of ways to get there: $C(n,j)p_u^j p_d^{n-j}$.

The probability bars in Figure [3] have the characteristic shape of a lognormalprobability distribution. This shape reflects the fact that the price can never reach zero, no matter how many down moves take place; each downward move shrinks the previous price by a fixed fraction *d*, always less than one. The upside potential is unlimited, since *u* is always greater than one.

You can see why the probability of ending up at prices not too far from the initial price is higher than the chance of reaching the extremes. There are many combinations of up/down motions leading to the middle of the price range, but the two price extremes can only be reached by relatively improbable sequences of all-up or all-down movements.

Binomial prices are not quite log-normal

It's sometimes asserted that in the long run, the binomial process converges to a log-normal distribution. That isn't quite true. Figure [4] illustrates how the interest rate distorts the probability distribution of long binomial prices runs.

The data in the Figure were produced by repeatedly generating sequences of 500 binomial price moves over one simulated year, starting each sequence at an initial price of 100. The 500th (year-end) price of each sequence is taken as a sample data point, and the graph shows the cumulative probability distribution function (CDF) of more than 1000 data points. In other words, it shows the probability that the binomial stock price after 500 moves will be less than or equal to each price *P* on the x-axis. You can see that this binomial distribution deviates systematically from the log-normalCDF drawn in blue. If I pick values of s and interest rate to make $p_d > p_u$, the binomial CDF will spill out to the left of the log-normal line. The curves match only in the special case where $p_u = p_d$. At 4% interest, this happens when s is about 0.28.

Price runs

So, where do the prominent price cycles in Figure [1] come from?

Consider any price path that always stays within the blue region of Figure [3]. In such a "price run", the simulated stock price never falls below the initial price. It's a simplified form of price cycle: the price heads up and stays up for a while, until it eventually falls back below the initial price. Similarly, a "down" run stays within the red area until it finally moves back above the initial price.

A run that reaches the right edge of the blue area somewhere above the initial price after *n* price steps, requires additional price steps before it can possibly fall back to the initial price. So in Figure [3], *all runs of length 15 or longer* pass through the right edge of the price map. Obviously the map can be extended to cover any number of price steps.

How likely are such price runs?

I don't think there's a simple, closed formula involving combinations or permutations that gives the answer, because the constraint that runs can't cross the middle of the price map excludes certain combinations of up-and-down movements. For example, *UDDUU* is an excluded (wandering) path while *UUDDU* is a valid run in the blue region. Still, we can count the runs. Look first at the path consisting exclusively of upward moves. This is the upper edge of the blue region in Figure [3], and there is only one way to get to any node along this path: all the moves leading to it are up moves.

Now if you look at any "interior" node adjacent to this upper edge, but still within the blue area, there will be two ways to get to it (down or up from the preceding nodes), except that the bottom blue nodes are reachable from just a single blue predecessor. Obviously the total number of paths to any interior node is the sum of the number paths to its one or two direct predecessors. With this observation, you can work your way through the price map from left to right. When you reach a node at the right end, you'll have accumulated the total number of paths leading to it.

This counting procedure is easy to perform, but the bookkeeping is best done by a computer.

How probable are long price runs?

First, let's see what proportion of all the possible *n*-move paths are runs, and what proportion wander above and below the initial price. Figure [5] plots the answer for typical conditions ($s = 0.15$, interest rate 4%, daily price movements). Each point on the blue curve indicates the proportion of all the possible paths of length *n* price steps *or longer*, that are runs above or below the initial price. The green curve counts all the other "wandering" paths.

Using the mouse to help read graph values accurately, you can see that more than half of all paths of length 9 or less are runs. Even when we consider paths of at least 200 moves, more than 10% are runs.

If a price outcome after *n* steps is *j* "up moves" above the initial price, the probability of any run leading to that price is, as before, $p_u^{\,j} p_d^{\,n-j}$. And we know how to count the runs leading to each price, so we can calculate the cumulative probability of runs to each price at the right edge of the price map, and add them up to get the total probability of runs with length greater or equal to value of *n*.

Figure [6] plots the result of this computation for our familiar case when $s = 0.15$ and interest is 4%. It shows that the chance of a run exceeding 200 moves is more than 10%, and likelihood of runs exceeding 1000 moves is nearly 5%. Given these probabilities for runs of various lengths, we can also estimate the probability-weighted, expected length of runs above and below the initial price - about 44 steps in the present example. But notice how the curves flatten as *n* increases. The probability of a run at least 500 steps long is not much less than the probability of a run at least 200 steps long. Intuitively, the reason is that runs which do migrate far from the original price require many moves before they can possibly return to that original price. Expect some very long runs.

10,000 steps? 20,000? Each new price move doubles the number of possible paths in the map, and every path that was a "wanderer" after *n* steps forks into two wandering paths in the next price step. However, that next step also turns some paths, which had been successful runs, into wanderers. Therefore the proportion of wanderers always increases, and the probability of an awesomely long run is vanishingly small.

All good runs come to an end; but in real markets, it ain't necessarily so. As I mentioned earlier, studies of actual market history indicate that if you hold "an average stock" long enough, the return on your investment will become deterministically positive, in obvious contradiction to the binomial model.

The Cox-Ross-Rubenstein option value

Having come this far, we ought in passing to discuss the binomial value of an American option. Figure [7] illustrates option values attached to a price map. Initially, the map spans the time from "now" to the terminal price when the option (in this example, a $100 call) expires. Off each price node at the right edge of the map, the program displays the value the option would have if the stock has the corresponding price at expiration. If the stock closes at $100 or below, the option expires worthless. Above $100, its value rises dollar-for-dollar with the stock.

Now let's back up one step in the price map (Figure[8]). As we move left to the predecessor node of each terminal price, we can calculate the expected value of the option at the preceding price, since we know probabilities p_u and p_d. The expected value is just the probability-weighted sum of the two terminal values. To be absolutely precise, we discount this value by the risk-free interest rate for the time between the nodes.

This procedure is simply repeated as we propagate back through each step of the price map, until the current date is reached. The result is the "fair value" of the option. As a further refinemen, we can test to see whether the option should be exercised at each step. It should be exercised if its intrinsic value is less than the value that would be received by exercising it. For instance, if the option is a put, cash is received for shares when it's exercised. If the interest that can be earned on the cash is more than the value of the option, it would be economically rational to exercise it.

The binomial forecast

Essentially, the binomial model acts out a forecast of future price movements implied by the value of σ , the stock price volatility. It's true that any particular path produced by a simulation is selected by a random number generator; but as the analysis demonstrates and the eye confirms, *every* binomial path of any significant length exhibits runs and cycles.

So we have a statistically driven price movement model, with no trading or economic information content, that inherently produces price cycles much like those in real markets. What can it tell us about real markets? Here's an entertaining speculation: perhaps binomial cycles could be used as norms for comparison with real price series. For instance, we might try to "subtract" the inherent cyclicality implicit in a particular value of σ from actual price series, by statistically comparing the frequency/power spectrum of binomial cycles to the spectra of real markets. What's the motivation for this idea?

The binomial process approximates what mathematicians call a *continuous diffusion* process. Roughly speaking, this means small, incremental stock price movements in the absence of surprises. These processes assume we can accurately characterize the expected return for owning the stock, which presumptively has no "unexpected returns" when the price gaps up or down.

Of course, it's those unexpected returns that traders want to spot; and the sooner, the better. We'd like to notice when a stock starts experiencing "discontinuous diffusion".

Under the continuous diffusion assumptions, a stock's price volatility implies certain statistical expectations about its future price movements within a specified time period: in other words, probable limits. We usually describe this in static terms. For instance, we might calculate a 99% chance (three standard deviations) that the stock price will be between 85 and 119 after six months. It could exceed those limits, but the chance of great deviations is much less than the chance of small ones.

Inherent cyclicality in price movements

A more fundamental interpretation of volatility refers to the underlying price change process. The relative price change P_{i+1} / P_i is an estimate of the slope or first derivative of the price line over the time period
$(t_{i+1} - t_i)$; thus
s actually characterizes the *rate of change* of the stock price. The price change over a specified (long) time is the sum of all the little changes that take place during that interval, which binomial prices approximate with discrete up and down price movements. The price limit boundaries described in the previous paragraph are approximated by the maximum number of up or down movements that can take place in the specified time.

A useful trading tool?

If you're the kind of person who likes to think about "technical" trading tools, the foregoing analysis of run probabilities suggests some ideas you might like to explore.

First, is the distribution of run lengths for actual stock price movements really fractally similar on different time scales? This could be tested by accumulating the actual number of "up" and "down" runs at different time scales: say, comparing the frequency of runs in hourly price samples with runs in the weekly and monthly numbers. If the relationship between the short-term and long-term distributions changes suddenly, that may be a sign of something interesting happening to the stock.

Second, is the distribution of "down" runs similar to that of "up" runs? We would expect so in a market at equilibrium. If the equilibrium shifts or breaks down, there should be a period when the up and down run distributions become sharply different. There are a number of tests that can be used to determine when probability distributions are significantly different from each other or a "standard" benchmark distribution.

I'd like to tell you I've done this with real data - but I haven't yet. I'd be delighted to hear from anyone who tries these tests, or has other or better ideas.

Conclusions

These speculations are entertaining, but a more basic lesson is this: if you believe that most of the time, the next stock price is closely related to recent prices - which is the core assumption of the binomial process - then you must expect cycles and long runs (trends) to appear in price data even when psychology, expectations, momentum and other technical factors are neutral. Price trends alone, without measuring other tangible factors, can easily mislead unless you account for the inherent cyclicality described here.

A deeper observation has to do with the importance of *process* over statistics. Despite its simplicity, the binomial model is very carefully and cleverly contrived. Its process structure - a rigid assumption of fixed-magnitude up-or-down movements - directly constrains the simulated stock's time-rate-of-change.

Real markets, on the other hand, often exhibit powerful trends that move outside these statistical constraints. When this happens we can say "the volatility has changed", but that really means the model broke down. Although we adjust the parameters, sooner or later the model will fail again. Some market trends are driven by intangibles like trading psychology, but others are based on persistent value relationships among investment alternatives (like the relationship between stock and bond prices). Models that don't account for the non-linear, feedback properties of these relationships obscure reality, even if the resulting probability distributions have have statistically plausible shapes. The most important thing to remember about empirical probability distributions is the enormous amount of information discarded to derive them.

Further reading

John Hull's book, *Options, Futures and Other Derivative Securities,* is a standard text that covers the binomial model.

I prefer the presentation in Rajna Gibson's *Option Valuation*, which I find more complete and readable than Hull's. She also includes a critique of the binomial model *vs.* real security prices, and a useful review of the work on price movement processes where the volatility occasionally "jumps".

You might also enjoy Edgar Peters's book, *Fractal Market Analysis: Applying Chaos Theory to Investment and Economics.* This book covers all the "standard" statistically characterized process models, then focuses on a particular technique called "rescaled range analysis" or "R/S analysis". The R/S method is useful for processes that, like the binomial one, generate runs and cycles that are intrinsically non-periodic. I don't subscribe to everything Peters asserts, but it's an interesting take on the subject.

About the author

Roger Ison is co-founder and president of Mantic Software Corporation, Ft. Collins, Colorado. Mantic makes interactive software to teach the theory and practice of investment. Before founding Mantic, Ison was an R&D manager and business strategist at Hewlett-Packard. He studied political science and economics as an undergraduate, and earned his M.S. and Ph.D. in Computer Science at the University of Virginia.

How to display Figures from the article

To see the Figures described in the article, run the *Binomial Market Model* program and set up specific displays as described. You can experiment with the program at the same time, and get more insight into how the model operates.

If the program isn't already running, start it by double-clicking its icon with the left mouse button. (Position the mouse sprite over the icon and click the left button twice in quick succession without moving the mouse.)

Figure [1] Binomial price series

To generate a binomial price series:

If *Binomial Market Model* is already running, click the **Reset** button in the main window. This enters some default parameter settings and puts the program into Price series mode. If the program isn't running, start it by double-clicking its icon.

Click the **Proceed** button in the Main window. The label of that button changes to **Plot**, and the **Day Hour:Min** display starts changing rapidly. Let the program run for as long as you wish, 100 or 200 days or more.

Click the **Plot** button to see the price series.

The X-axis of the graph represents days before the present simulated moment; the Y-axis is the stock price.

You can change the volatility and interest rate scroll bars at any time, even while the simulation is running. The simulator stops generating new prices when you click **Plot**, but you can continue by clicking **Proceed** again in either window. Click **Reset** to start a new series.

Short- and long-term moving averages are displayed in red and green, respectively. At the top of the plot you'll also see the calculated, recent short-term and long-term volatility values for the series at the time the plot is made. When Steps per sample is set to 1, these two numbers are almost always virtually identical to the volatility scroll bar setting in the Model parameters panel. When there are multiple steps per sample, the two volatility numbers often differ substantially.

Figure [2] Main window showing binomial parameters

This Figure is just the Main window of the *Binomial Market Model* program. Run the program and set it up as follows:

If *Binomial Market Model* is already running, click **Reset** in the main window. This enters some default parameter settings and puts the program into Price series mode. If the program isn't running, start it by double-clicking its icon.

The default settings of the binomial parameters correspond to the example in the article.

The up- and down-multipliers, and associated probabilities, are shown in the Model parameters panel of the Main window. To see how they change as a function of volatility and interest rate, adjust the scroll bars. The calculated values change as the settings are adjusted.

The time step size also figures into the model, so adjusting the Minutes per sample and Steps per sample scroll bars also affects the calculated up/down numbers.

Figure [3]   Binomial price map with probability bars

To see an interesting example of the binomial price map:

If *Binomial Market Model* is already running, click **Reset** in the main window. This enters some default parameter settings and puts the program in Price series mode. If the program isn't running, start it by double-clicking its icon.

Click the Distribution check box in the Simulation panel of the Main window. This changes the time scale of the Minutes per sample scroll bar to span days rather than minutes. The default is now 365 days (one year).

Adjust the Days per sample scroll bar to 30 days (one month). You may want to click the arrows at either end of the scroll bar for fine control of the setting.

Click the **Plot window** menu item in the Main window. This displays the Plot window without any graph.

In the Plot window, click the **Other graphs** menu item. Select **Price movement map** to display the graph.

The vertical axis of the graph is price; the horizontal axis spans 15 monthly price moves over a period of 1.2 years. Note how the map shows a pronounced skew up toward higher prices.

You can display this graph for any time scale and binomial parameter settings. The log-normal skew is not as apparent when the time scale is small.

Figure [4]   Binomial CDF vs Log-normal CDF

You can compare the binomial cumulative probability distribution function (CDF) with a log-normal CDF, and test the effect of the number of price steps per sample.

This display graphs the (cumulative) probability that the stock price will be less than or equal to each price on the X-axis when the "next" price sample is generated:

If the Binomial Market Model program is already running, click **Reset** in the main window. This enters some default parameter settings and puts the program into Price series mode. If the program isn't running, start it by double-clicking its icon.

Click the Distribution check box in the Simulation panel of the Main window. This changes the time scale of the Minutes per sample scroll bar to span days rather than minutes. The default is now 365 days (one year).

Leaving **Steps per sample** set to its default value of one, click the Proceed button. Let the simulation run for a while, say a hundred or more samples. You can watch the samples count up in the upper left corner of the Main window.

Click the **Plot** button to see the CDF. In this case, the binomial model can only generate two possible prices: one up move or one down move, corresponding to one step per sample. Therefore the binomial CDF is a step function with two ledges.

Now adjust Steps per sample to 500 by sliding its scroll bar control all the way to the right. Click Proceed and let the simulation run through, say, 100 samples. Click **Plot** again. Now the accumulated distribution looks much smoother. If you continue out to 1000 or more samples, the empirical distribution becomes quite smooth.

You can also see a systematic deviation between the blue line, corresponding to a true log-normal CDF, and the binomial data. If you adjust the volatilityscroll bar carefully until prob[Up] = prob[Down] (or as close to equality as you can get them) and re-run the distribution plot, the log-normal and binomial CDFs will essentially coincide after many data samples.

Figure [5] Proportion of paths that are runs of at least N steps

This graph takes quite a while to generate on 386 or 486SX computers - as much as 5 or 10 minutes. These computers have no floating-point processor, so the arithmetic is slow. On 486DX machines, the computation typically takes about 60 seconds.

To see the graph:

Adjust volatility, interest rate and time scale to any values you choose.

If the Plot window is not displayed, click the **Plot window** menu item in the Main window.

In the Plot window, select the Other graphs menu item. Choose **Runs vs. other paths** from the drop-down list, and wait for the graph to be drawn.

Figure [6]   Probability of runs with length at least N steps

This graph takes quite a while to generate on 386 or 486SX computers - as much as 5 or 10 minutes. These computers have no floating-point processor, so the arithmetic is slow. On 486DX machines, the computation typically takes about 60 seconds.

To see the graph:

Adjust volatility, interest rate and time scale to any values you choose.

If the Plot window is not displayed, click the **Plot window** menu item in the Main window.

In the Plot window, select the **Other graphs** menu item. Choose **Prob of long runs** from the drop-down list, and wait for the graph to be drawn.

Figures [7] and [8]

The binomial option price can be computed directly from the price movement map. To see this in action,

If the Plot window is not displayed, click the **Plot window** menu item in the Main window.

In the Plot window, select the **Option value** menu item.

Indicate in the pop-up whether you want to value a call or a put. Choose its strike price, and the European or American exercise style. Click OK.

A 15-move price map will be displayed, with the option values at expiration. (The expiration is always 15 moves in the future, with the time span of each price move derived from the binomial parameters in the Main window, as usual.

Now, each time you click the **Proceed** button, the program will back up one step in the price map and propagate the option values backward as described in the article.

Cumulative probability distribution function (CDF)

Let X be a random variable whose value, at every sampling, is within the range
$X_{\min} < X \le X_{\max}$. Then
$CDF(X,Y)$ is defined as the probability that a sample value of
X is less than or equal to
Y.

By this definition $CDF(X_{\min}) = 0$, and
$CDF(X,X_{\max}) = 1$ because it is certain that any sample value
X will be less than or equal to
$X_{\max}$.

For example, Figure [4] graphs the (cumulative) probability that the stock price will be less than or equal to each price on the X-axis when the "next" price sample is generated.
The Y-axis of Figure [4] runs from zero to one.

You can see that the probability of the random variable falling within any arbitrary sub-range of its possible values $a < X \le b$ is simply
$CDF(X,b) - CDF(X,a)$.

Normal and Log-normal CDFs (Cumulative Probability Distribution Functions)

A random variable has a *normal* CDF if its probabilities are defined by the well-known bell-shaped, or Gaussian, curve.

A random variable has a *log-normal* CDF if the natural logarithms of its sample values $\log_e(X)$ are normally distributed.

Combinations C(n,j)

The number of combinations of *n* things taken *j* at a time is

$$C(n,j) = \frac{n!}{j!(n-j)!}$$