Automated Data Acquisition and Analysis of Stock Options

Lingyan Zhang

School of Computer Science

Mcgill University, Montreal

May 2001

A thesis submitted to the Faculty of Graduate Studies and Research

in partial fulfillment of the requirements of the degree of Master of Science

Abstract

This thesis describes a system that retrieves exchange-traded option quotes on stocks, which are specified by the user, from the website www.moneynet.com. The data obtained from the Web is analyzed to determine the "breakeven" stock prices. A breakeven stock price is the expiration-date stock price for call or put options, in order for the call or put holders as a group to break even. This information is very useful as it helps us to understand the stock markets and the opinions of stock traders. It is useful for expectation analysis as well. The system is written in Excel 2000 and the data obtained from the Web is displayed on Excel spreadsheets and is transformed into a format that can be analyzed by the program. The program consists of approximately 900 lines of VBA code and uses many advanced Excel built-in functions.

Résumé

Cette thèse décrit un système qui recherche des options cotées en bourse spécifiées sur des stocks par l'utilisateur sur le site www.moneynet.com. Les informations obtenues sur le Web sont analysées pour déterminer les seuils de rentabilité. Le "seuil de rentabilité" est le prix de stocks à la date d'expiration pour les options d'achat et les options de vente qui fait que tous les acheteurs d'option ne fassent ni gain ni perte sur leur position. Cette information est très utile puisqu'elle nous permet de comprendre le marché des stocks et reflète l'opinion des commerçants dans ce domaine. Aussi est-elle essentielle pour l'analyse de prévision. Le système est écrit en Excel 2000 et l'information obtenue sur le Web est arrangée sur des feuilles électroniques en Excel et transformée en un format pouvant être analysé par le programme qui, lui-même, consiste d' à peu près 900 lignes de code VBA et utilise plusieurs fonctions avancées intégrées à Excel.

Acknowledgments

First of all, I wish to express my sincere gratitude to my supervisor, Professor Monty Newborn, for his wonderful supervision. His constant encouragement and trust led me to this new and interesting area of study and his helpful suggestions have guided me throughout the course of this project. As well, this thesis would not have its present form without his invaluable comments on the earlier drafts.

I would also like to thank Delize Williams for proofreading my writing, Danielle Azar for translating the abstract into French, and Liqian Gao for her friendship. Their suggestions are always great sources of inspiration.

In addition, I would like to acknowledge the assistance of Bin Zhang, Jun Chen and Xing Yu in producing hardcopies of this document. This is truly appreciated.

And last but certainly not least, I wish to thank my husband, Feichi Long, as well as my parents, Junyi and Shuqin, for their love and support in my life.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Options have been traded for centuries, but they remained relatively obscure financial instruments until the introduction of exchange-traded options on the Chicago Board Options Exchange (CBOE) on April 26, 1973. Since then, the option markets have grown tremendously. Today, on CBOE alone, options are traded on over 1200 different stocks and the number is still increasing. In the United States, stock options are also traded on four other exchanges: the Philadelphia Exchange (PHLX), the American Stock Exchange (AMEX), the Pacific Stock Exchange (PSE), and the International Securities Exchange (ISE).

1.1 Stock Options

Stock Options are derivatives of underlying stocks, which means that each stock option derives its value from an actual stock [8]. Therefore, the price movement of the underlying stock directly influences the value of the stock option and the option's trading activity.

1.1.1 Basic Concepts

A **stock option** is the right to buy or sell an underlying stock at a specified price on or before a given date. The specified price is called the **strike price** and the given date is called the **expiration date** [8]. After the expiration date, the stock option expires and is worth nothing. Both the strike price and the expiration date are predetermined by the exchanges. A stock option contract involves a buyer and a seller. The buyer is called the option **holder** while the seller is called the option **writer**. To buy an option, the holder pays the writer the **premium**, the price of the option. Although the holder has the right, he has no obligation to use it. If the holder does use the right, this is called **exercising an option**. A single stock option contract generally is for 100 shares of the underlying stock.

There are two types of stock options: calls and puts. A **call option** gives the holder the right to buy while a **put option** gives the holder the right to sell. When a call option is exercised, the writer of the call has an obligation to sell the underlying stock to the holder at the strike price. When a put option is exercised, the writer has an obligation to buy the underlying stock from the holder at the strike price. **American options** can be exercised at any time up to the expiration date. **European options** can be exercised only on the expiration date itself [4]. All exchange-traded stock options are of the American style [8].

A call option holder hopes that the stock price will increase. If the stock price passes the strike price before the option's expiration date, he can exercise the call to buy the stock from the writer of the call at the strike price and sell it immediately to the market at a higher price, and thus earn the difference between the market price of the stock and the strike price. The holder of a put option hopes that stock price will decrease. If the stock price falls below the strike price before the option's expiration date, he can exercise the put to buy the stock from the market at a lower price and sell it to the seller of the put at the strike price, and thus earn the difference between the strike price and the market price of the stock. If the underlying stock doesn't move in the direction that the holder hopes before the option's expiration date (in other words, the stock price is lower than or equal to the strike price for a call option, or higher than or equal to the strike price for a put option), the holder would not exercise the option [9][10]. Taking the premium into account, Figures 1.1 and 1.2 show the profit and loss for an option holder from buying a call option and a put option, respectively.

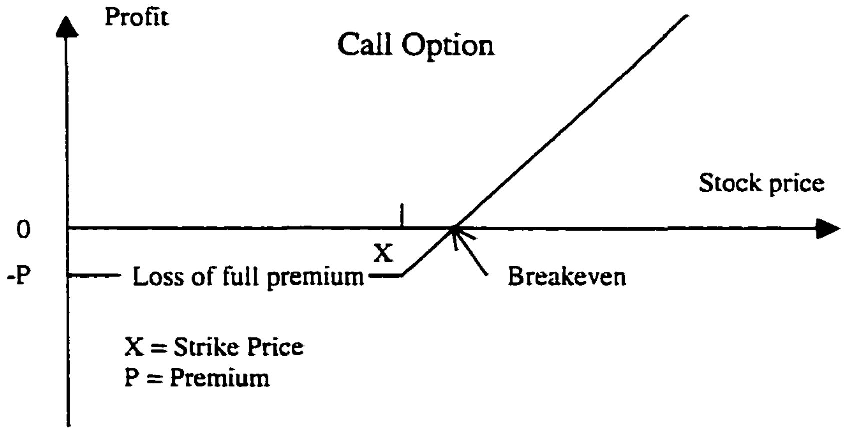


Figure 1.1 Profit from buying a call option

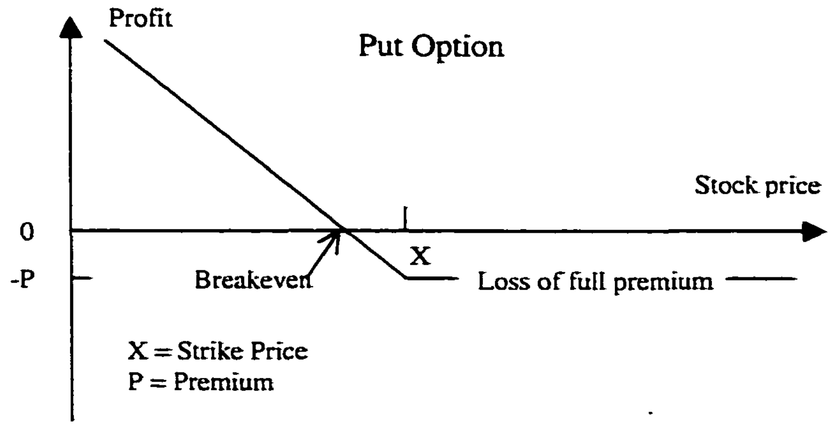


Figure 1.2 Profit from buying a put option

As can be seen from the figures, when buying a stock option, the most the holder can lose is the premium he pays for the option and he has an unlimited potential for profit.

A stock option can always be categorized in one of three ways: **in-the-money, at-the-money**, sometimes also referred to as on-the-money, or **out-of-the-money**. If S is the stock price and X is the strike price, a call option is in-the-money when S > X, at-the-money when S = X, and out-of-the-money when S < X. A put option is in-the-money when S < X, at the money when S = X, and out-of-the-money when S > X. Obviously, only when an option is in the money, it will be exercised [8].

Exercising options is not the only thing option holders can do. In fact, most option holders trade their options on exchanges instead of exercising them [1]. In other words, they resell their rights to buy or sell to other people and take the profit directly on the options themselves. A stock option's total value consists of two components: intrinsic value and time value. Options will have intrinsic value only if they are in-the-money. The **intrinsic value** of an in-the-money option is equal to the difference between the option's strike price and the market price of the underlying stock. Clearly, the deeper a stock option is in-the-money, the higher an intrinsic value it will have. Any value in an option above its intrinsic value is its **time value**. The more time a stock option has before its expiration, the higher its time value will be since additional time improves the chances that the option will make the desired move [13][14].

1.1.2 Option Quotes

Option quotes available on the Internet include detailed information on options' trading. Table 1.1 shows a typical option quotation from the Internet. This quotation refers to the trading of IBM calls on April 25, 2001. Due to space limitations, only options with strike prices between $100 and $120 are shown in the table. An option can be referred to by its symbol or by the stock on which it trades, the expiration month and the strike price. For example, option "IBMET" can also be referred as May 2001 IBM call with a strike price of $100.

Expiration Month Strike Price Option Type Option Price

IBM CALLS									
Symbol	**Description**	**Last Trade**	**Net Chg**	**Vol**	**Open Interest**	**Daily High**	**Daily Low**	**Date**
IBMET	IBM **May1** 100. C	15.60	+1.80	507	9610	15.60	14.50	04/25
IBMEA	IBM May1 105. C	11.50	+1.50	4257	14989	11.50	9.60	04/25
IBMEB	IBM May1 110. C	7.50	+1.20	778	14052	7.80	5.90	04/25
IBMEC	IBM May1 115. C	4.50	+0.80	1775	13457	4.60	3.40	04/25
IBMED	IBM May1 120. C	2.50	+0.65	3407	12131	2.50	1.60	04/25
IBMFT	IBM **Jun1** 100. C	16.20	0.00	0	44	0.00	0.00	04/24
IBMFA	IBM Jun1 105. C	11.90	+0.10	1	1593	11.90	11.90	04/25
IBMFB	IBM Jun1 110. C	10.00	+1.50	107	347	10.00	9.30	04/25
IBMFC	IBM Jun1 115. C	6.90	+0.90	1131	457	6.90	5.60	04/25
IBMFD	IBM Jun1 120. C	4.50	+0.80	1062	732	4.60	3.40	04/25
IBMGT	IBM **Jul1** 100. C	19.10	+1.50	80	5324	19.10	16.70	04/25
IBMGA	IBM Jul1 105. C	15.90	+1.60	303	11073	15.90	14.30	04/25
IBMGB	IBM Jul1 110.0 C	12.10	+1.00	44	12536	12.10	10.80	04/25
IBMGC	IBM Jul1 115.0 C	9.50	+1.00	739	12962	9.50	7.70	04/25
IBMGD	IBM Jul1 120.0 C	7.00	+0.70	62	19610	7.00	5.70	04/25
IBMJT	IBM **Oct1** 100. C	22.10	+0.20	10	2434	22.10	22.10	04/25
IBMJA	IBM Oct1 105. C	19.50	+1.50	11	988	19.50	19.00	04/25
IBMJB	IBM Oct1 110. C	16.20	+1.10	9	1469	16.30	15.80	04/25
IBMJC	IBM Oct1 115. C	13.20	+0.80	4	1711	13.20	13.00	04/25
IBMJD	IBM Oct1 120. C	11.00	+0.70	555	3850	11.10	9.90	04/25

Table 1.1 Stock option quotation of IBM calls from www.moneynet.com on April 25, 2001

The expiration month in Table 1.1 is the month in which a stock option's expiration date occurs. The precise expiration date is the Saturday immediately following the third Friday of the expiration month [8]. Since the exchanges close on weekends, for option traders, the last trading day is the third Friday of the expiration month.

All options of the same type, either calls or puts, are referred to as an **option class**. IBM calls in Table 1.1 are one option class while IBM puts are another. In one option class, options with the same expiration date and strike price form an **option series**, whose trading activity is recorded in Table 1.1 in one row [8].

The value of *Last Trade* is the price of an option to buy one share of underlying stock at the last trade. The values of *Daily High* and *Daily Low* are the highest and the lowest trading prices during the day. As mentioned earlier, one option contract is for 100 shares of the underlying stock. Therefore, an option contract costs 100 times the price shown. The value of *Vol* is the trading volume, which is the number of option contracts that have been traded during the day. This reflects how heavy the trading activity is on the single stock-option series. The value of *Open Interest* is the number of outstanding contracts, up to and including the quoted day, which have not been exercised. Open interest is a more accurate measure of cumulative demand for an option than is trading volume [12]. A unit of open interest represents two parties: a holder and a writer. Open interest will increase by one contract when both holder and writer initiate a new position, thus creating a new contract. Open interest will decrease by one contract if both holder and writer close an old position, thus liquidating an existing contract. Open interest will stay the same if an old trader passes off his position to a new trader, such as an old holder sells his right to a new holder [16].

In Table 1.1, within the same expiration month, there are different predetermined strike prices. Therefore, option traders can choose one to trade with. Conversely, from their trading, a clue of their opinions about the future stock price can be obtained.

It can also be seen, in Table 1.1, that options with different expiration months trade at the same time. At any given time, the expiration months with which stock options trade depend on the trading cycle of the options, which is assigned by the exchanges. Stock options are on a January, February, or March cycle. The January cycle consists of the months of January, April, July, and October. The February cycle consists of the months of February, May, August, and November. The March cycle consists of the months of March, June, September, and December. For the current month, if the expiration date has not been reached, options trade with expiration dates in the current month, the following month, and the next two months in its own cycle. If the expiration date has passed, options trade with expiration dates in the next month, the month after next, and the next two months of the expiration cycle [8]. For the example in Table 1.1, the quoted date, April 25, 2001, has already passed the expiration dates in that month, which is April 20, 2001. Since IBM is on a January cycle, IBM options currently trade with expiration dates in May 2001, June 2001, July 2001, and October 2001. In summary, an option with an expiration month in its trading cycle can be listed as early as eight months before its expiration month while an option with the other expiration months can be listed two months before its expiration month.

In addition to these regular-term stock options, longer-dated stock options known as Long-term Equity AnticiPation Securities or LEAPS, are also available on some stocks. All equity LEAPS expire in the month of January up to three years in the future. A new

LEAPS is initially listed after the expiration date in May for options on a January cycle, in June for options on a February cycle, or in July for options on a March cycle [4].

1.2 Purpose of the Thesis

The purpose of this thesis was to build a system that retrieves daily option quotes from the Web and analyzes them [15]. Because stock options require a relatively smaller initial investment than their underlying stocks and they also have unlimited profit potential, they provide investors with effective ways of managing investment risks [8]. Therefore, most studies on stock options focus on option pricing and the development of trading strategies. However, the motivation of this thesis is to assess investors' opinions about the underlying stocks from their option trading activities. What makes this possible is the special relationship between stock options and their underlying stocks [11]. That is, the stock options are derivatives of their underlying stocks.

Through analyzing stock option quotes, we found that investors' opinions can be quantitatively measured by computing what we call "breakeven" stock prices. The breakeven stock price is the expiration-date stock price for call or put options in order for call holders or put holders, as a group, to break even. Breakeven stock prices can be very helpful in understanding the current stock markets and are essential information for expectation analysis as well.

1.3 Expectation Analysis

Expectation analysis is a study on evaluating the opinions and sentiments of the investing community. The accurate interpretation of this evaluation can help investors predict the important turning points of a stock trend, therefore, develop their investment decisions and make a profit from their trading. Expectation analysis is a supplement to the traditional approaches to stock analysis: fundamental analysis and technical analysis. Fundamental analysis uses factors such as earnings, dividends, and economic projections to forecast stock prices. Technical analysis focuses on historical price patterns and volume characteristics to predict stocks' future performance [12].

In recent years, expectation analysis has proven to be a very effective and critical approach in stock trading. More and more investors and financial analysts use expectation analysis to help predict the movement of the stock price.

1.4 Assumptions of the Thesis

Stock options can be exercised or traded on the exchanges at any time before their expiration dates (since the exchange-traded options are of American style.) This fact brings great complexity in the stock option analysis and makes the analysis more difficult. In order to conduct the analysis, we assume the stock options will not be exercised before their expiration dates. In fact, studies show that it is never optimal to exercise an American call option on a non-dividend paying stock prior to expiration. There are only some circumstances under which the early exercise of an American put option on such a stock is optimal [8]. Thus, today's outstanding option contracts will remain until their expiration dates, even through their holders may change. In addition, the following is assumed:

- There are no transactions costs and commissions during option trading.
- Risk-free interest rate is constant and remains the same for all maturities.

1.5 Outline of the Thesis

This introductory chapter has reviewed basic stock option concepts that will be used in this thesis, indicated the purpose of this thesis, and discussed the assumptions of the study.

The following chapter reviews the approaches that can be used to retrieve valuable information on investors' opinions from option trading, including implied volatilities and put/call ratios. Contrary opinion theory is often applied in the interpretation of this information [12]. More details on this theory are presented as well.

Then, Chapter 3 introduces the approach used to calculate breakeven stock prices for call or put options. To illustrate the methodology, two examples are presented.

Chapter 4 documents the design and implementation of the Breakeven Stock Price (BSP) system. BSP is an Excel application, which downloads stock-option quotes from the Web and displays this raw data along with the breakeven stock prices. Because of the source of option quotes, BSP can only be applied to stock options trading on US exchanges. For those trading on the other exchanges, a little modification needs to be done before using the system.

Next, Chapter 5 outlines how BSP is to be used and discusses the outcome obtained.

Finally, Chapter 6 summarizes the thesis and speculates about possibilities for future work.

Chapter 2

Option-Based Information Retrieval

In this chapter, the current available approaches that are used to retrieve information on investors' opinions from option trading are reviewed, including implied volatilities and put/call ratios. Since contrary opinion theory is often used to interpret the information, it is presented first.

2.1 Contrary Opinion Theory

The great contrarian Humphrey B. Neill, in his book *The Art of Contrary Thinking* [12], presented a new idea of thinking: contrary thinking. According to Neill, "The 'crowd' is most enthusiastic and optimistic when it should be cautious and prudent; and is most fearful when it should be bold" and "because a crowd does not think, but acts on impulses, public opinions are frequently wrong." So a smart trader should avoid thinking like the crowd, but instead should adopt a contrarian or opposite viewpoint to that of the crowd. But, Neill also states "The public is right during the trends but wrong at both ends." Therefore, contrarians don't always go against the vast consensus of the investing public, but rather, they try to find the extremes.

When the public is optimistic about the market, they buy, and as a result, the stock price goes up. The public is right all the way up until the optimism reaches an extreme point. At that time, most of the buying power has already been used to purchase the shares. Then, when the inevitable profit taking occurs and there is not enough money on the sidelines to absorb the selling, sellers must offer their shares at a lower price, and the stock price moves down. So, the public is wrong at the top. The same is true when the public is pessimistic about the market. They are right all the way down and wrong again at the bottom.

Contrarians believe that excessive option speculation on one side of the market often marks classic turning points for individual stocks. So contrary opinion theory is to identify ends and to behave opposite to the public. Generally, the more optimistic the majority is, the more significant the top is; the more pessimistic the majority is, the more significant the bottom is [12].

2.2 Implied Volatilities

Volatility is a measure of the fluctuation of a stock on which an option trades. Stocks whose price ranges are large are said to have a high volatility. Those whose ranges are small are said to have a low volatility [12]. Historical volatility can be easily calculated from the past trading data by using a standard deviation-based formula, while future volatility is difficult to calculate directly because stock prices in the future cannot be observed. Alternatively, implied volatility is the option market's best assessment of the expected future volatility of the underlying stock [2][6], and it is obtained by plugging the actual option price, the stock price, the strike price, the time to expiration, the dividend, and the risk-free interest rate into the option pricing equations, such as the Black-Scholes model. Then the implied volatility is calculated [8].

Since there are six factors affecting the price of a stock option, which are the current stock price, the strike price, the time to expiration, the volatility of the stock price, the risk-free interest rate, and the dividends expected during the life of the option, option-pricing models are usually functions of these six variables [5]. Among these six variables, except for the volatility, the others can all be observed or calculated from the markets. This makes it possible to solve the model backwards from the observed option price to determine what the volatility input must be.

To illustrate this approach, the Black-Scholes model, which is the traditional option-pricing model, is used as an example. The Black-Scholes model is valid only under the following assumptions:

- No dividends and contributions are paid during the lifetime of option.
- Options are of the European style, which cannot be exercised before expiration.
- Risk-free interest rate is known and remains constant.
- The stock price is lognormally distributed.

Using the following notation,

S: current stock price,
X: strike price of option,
T: time to expiration of option in years,
r: risk-free interest rate,
σ: volatility of stock price,
c: value of call option,
p: value of put option,

the equation for the price of call option is,

$$c = SN(d_1) - Xe^{-rT} N(d_2),$$

and the equation for the price of put option is,

$$p = Xe^{-rT} N(-d_2) - SN(-d_1),$$

where

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

$$d_2 = \frac{\ln(S / X) + (r - \sigma^2 / 2)T}{\sigma \sqrt{T}} = d_1 - \sigma \sqrt{T},$$

and *N(x)* is the cumulative probability distribution function for a variable that is normally distributed with a mean of zero and a standard deviation of 1 [8].

In the equations above, although it is not possible to arrange the equations so that σ is expressed as a function of the other variables, after plugging in the other variables, an iterative search procedure can be used to find the implied σ.

The easiest search method is the method of bisection. The search begins with two trial volatilities, σ_{Low} and σ_{Hign}, with which two option prices bracketing the actual option value are calculated from the model. Next, the range is bisected by defining the midpoint of two trail volatilities: $\sigma_{Mid} = (\sigma_{Low} + \sigma_{High}) / 2$, and the option value with volatility σ_{Mid} is computed. If the new option value is greater than the market value, the range between σ_{Low} and σ_{Mid} is bisected next. Otherwise, the range between σ_{Mid} and σ_{Hign} is bisected. Each iteration cuts the possible range for σ in half, and the procedure continues until the desired accuracy is obtained [7].

A second common approach to computing implied volatility is Newton-Raphson search. Let σ_{Trial} denote some initial trial volatility, $c_{observed}$ denote the observed option price from the market, and let $v = \left.\frac{\partial c}{\partial \sigma}\right|_{\sigma=\sigma_{Trial}}$ be the Vega, which is the rate of change of the option price with respect to the volatility of the underlying stock, evaluated at σ_{Trial}. Then calculate $\Delta = (c_{observed} - c(\sigma_{Trial}))/v$, set the new value of σ_{Trial} to $\sigma_{Trial} + \Delta$, and repeat the process, until convergence to the desired accuracy is attained [7]. Newton-Raphson search can achieve convergence in a very few steps, much faster than bisection, but it requires calculation of both the option price and its Vega at each iteration, which requires so much extra computation that it may not be faster in practice.

A more sophisticated approach is to calculate several implied volatilities from options on the same stock with different strike prices, and then give different weight to each one of them. Thus a composite implied volatility for the stock is calculated, which is called weighted implied volatility. The amount of weight given to each implied volatility in this calculation is determined by the sensitivity of the option price to the volatility, which is measured by Vega. The more sensitive the option price is to the volatility, the more weight the implied volatility gets [8].

Implied volatilities show option traders' opinions about markets. Tracking trends in the implied volatility of the options on individual stock can often help pinpoint significant turning points in a stock. When a sharp spike appears in the implied volatility reading during a decline in the underlying stock (this happens usually because of the increase of the option price as the demand for puts by fearful investors reaches extreme levels), according to the contrary opinion theory, it indicates that the market is ready to turn around and rally, therefore sending a bullish signal. By contrast, when implied volatility reading is low, a bearish signal is sent [12].

2.3 Put/Call Ratios

The put/call ratio is the ratio of the trading volume of puts to the trading volume of calls over the same period for the same underlying stock [12]. The computation of the put/call ratio is straightforward because options' trading volumes can be obtained directly from the market.

The put/call ratio is a good indicator of investors' opinions. A low put/call ratio caused by heavy call trading, indicates that the investors are optimistic. Conversely, a high put/call ratio caused by heavy put trading, indicates that the investors are pessimistic. According to contrary opinion theory, when the ratio is quite low, too many traders are bullish, consequently the market tends to fall, or at least consolidate within a bull trend, and a bearish signal is sent. When the ratio is very high, too many traders are bearish, the market tends to rally, and a bullish signal is sent. The lower the ratio, the more bullish the market is thought to be. The higher the ratio, the more bearish the market is thought to be. Generally speaking, a put/call ratio of 0.75 or higher is considered bullish. A reading of 0.40 or below is considered bearish. Anything in between is considered neutral [4].

Because trading volumes record trading activity during one day, the put/call ratio can only capture investor opinions in a short term. On the other hand, open interest provides a more accurate measure of the overall demand for an option. So put/call open interest ratios are often used to confirm the signals sent [12].

By identifying extremes in public opinions, put/call ratios offer one of the most valuable tools to tell where the major market turning points are likely to occur.

Chapter 3

Breakeven Stock Prices

This chapter introduces the approach used to compute breakeven stock prices for calls and puts. The breakeven stock prices are expected expiration-date stock prices in order for call holders or put holders as a group to break even and can be interpreted as option holders' opinions about the stock prices on the options' expiration dates. First, the methodology is presented. Then, to illustrate this approach, a couple of examples are given where the breakeven price is calculated.

3.1 Methodology

As mentioned in Chapter 1, the cost of buying one option contract is 100 times the quoted option price. Using Q to denote the quoted option price, the cost would be $100 * Q$. If X is the strike price of an option and S is the expiration-date stock price, the payoff from a call contract at expiration is

$$\max(S - X, 0) * 100.$$

The payoff from a put contract at expiration is

$$\max(X - S, 0) * 100.$$

For a call holder to break even, the equation (3.1) must hold:

$$\max(S-X,0)*100=100*Q. \qquad (3.1)$$

For a put holder to break even, the equation (3.2) must hold:

$$\max(X-S,0)*100=100*Q. \qquad (3.2)$$

From equation (3.1), if $S \geq X$, S can be rewritten as $Q+X$ and is the breakeven stock price for the call. Only when the expiration-date stock price is greater than S, can the call holder make a profit. From equation (3.2), if $X \geq S$, S can be rewritten as $X-Q$ and is a breakeven stock price for the put. Only when the expiration-date stock price is lower than S, can the put holder make a profit. By computing S for a call or a put, the call holder or the put holder's opinion about the underlying stock can be measured.

Using a similar approach, the average breakeven stock price for the group of holders who buy the same class of option contracts on one stock with the same expiration date can be calculated. The total cost of these contracts is $\sum_{i=1}^{N} 100*Q_i*OI_i$. Here, i enumerates various strike prices. Q_i is the quoted option price and OI_i is the open interest corresponding to the option series. If these options are calls, the payoff from them at expiration is

$$\sum_{i=1}^{N} \max(S-X_i,0)*100*OI_i.$$

If these options are puts, the payoff from them at expiration is:

$$\sum_{i=1}^{N} \max(X_i-S,0)*100*OI_i.$$

For the call holders as a group to break even, the following equation must hold:

$$\sum_{i=1}^{N} \max(S - X_i, 0) * 100 * OI_i = \sum_{i=1}^{N} 100 * Q_i * OI_i . \qquad (3.3)$$

For the put holders as a group to break even, the following equation must hold:

$$\sum_{i=1}^{N} \max(X_i - S, 0) * 100 * OI_i = \sum_{i=1}^{N} 100 * Q_i * OI_i . \qquad (3.4)$$

It is difficult to rewrite equation (3.3) and equation (3.4) in order that S can be expressed as a function of the other variables. But a similar iterative search procedure, as used in the calculation of implied volatilities, can be employed to find the value of S. That is, different values of S can be chosen, and then adjusted, so that in every cycle, S will get closer to the true value.

From the option payoffs, it can be seen that trading options involves certain risks as option holders may lose their entire investment or a portion of it. Alternatively, potential option holders can always put the money into banks to gain risk-free interest. Therefore, if they choose to buy options, the holders have reasons to expect a higher return rate from option markets than the risk-free interest rate. When using P to represent the current value of premiums (the money that option traders invest in option markets), T to represent the time to the option's expiration in years, and r to represent the risk-free interest rate, the resulting value of the premiums at the time of expiration is Pe^{rT}. Taking this factor into consideration, the corresponding revisions are made to equation (3.3) and equation (3.4). For call options, this equation must hold:

$$\sum_{i=1}^{N} \max(S - X_i, 0) * 100 * OI_i = \sum_{i=1}^{N} 100 * Q_i * OI_i * e^{rT} . \qquad (3.5)$$

And for put options, this equation must hold:

$$\sum_{i=1}^{N} \max(X_i - S, 0) * 100 * OI_i = \sum_{i=1}^{N} 100 * Q_i * OI_i * e^{rT} . \qquad (3.6)$$

From equation (3.5) and equation (3.6), the breakeven stock price S for calls or puts can be obtained by inference, which reflects this group of option holders' overall opinion about the stock price on the expiration date of their options. Then, from the difference between the breakeven stock price and the current stock price, we can have a clue of how optimistic or pessimistic the option holders are. Furthermore, we can compare option holders' opinions on different stocks. To do so, rather than the absolute difference (between the breakeven stock price and the current stock price), the percentage of the difference with respect to the current stock price is compared.

$$percentage = \frac{breakevenstockprice - currentstockprice}{currentstockprice} * 100$$

Breakeven stock prices can be obtained as long as the options on that stock are actively traded. Because of the trading cycle of options, at any given time, options on one stock trade with different expiration dates. If the options quotes are divided according to their expiration dates and option classes, it is possible to obtain two breakeven stock prices, one for calls and the other for puts, for each expiration date.

3.2 Examples

In this section, option quotes of IBM on April 06, 2001 are used to illustrate how to calculate a breakeven stock price for calls and a breakeven stock price for puts.

The options used in the examples are July 2001 IBM options and the exact expiration date is July 20, 2001. The days between the quoted date (Apr. 06, 2001) and the expiration date (July 20, 2001) are 105, so T, the time to expiration in years, is 105/365 = 0.288. Suppose the risk-free interest rate r is 0.05, e^{rT} in equation (3.5) and equation (3.6) would be:

$$e^{rT} = e^{0.05*0.288} = e^{0.014} = 1.014488$$

3.2.1 Breakeven Stock Price for Calls

Table 3.1 shows the option quotes of July 2001 IBM calls. The cost of buying calls with the strike price of \$65 is \$35.2 * 100 * 138 = \$485,760. In the same way, the cost of buying other call option series can be calculated. The sum of these costs is the total value of July IBM calls.

Strike Price	Call Price	Open Interest
65.0	35.200	138
70.0	31.200	293
75.0	25.500	651
80.0	23.000	9097
85.0	19.800	1942
90.0	15.400	971
95.0	11.700	1641
100.0	9.300	3456
105.0	7.200	11298
110.0	5.500	10655
115.0	4.100	4416
120.0	2.800	10229
125.0	1.950	9055
130.0	1.400	4892
135.0	1.100	3337
140.0	0.800	6175

Table 3.1 Option quotes of July IBM calls on April 6, 2001 from www.moneynet.com

The payoffs from calls with the strike price of $65 on the expiration date depend on the expiration-date stock price. Assuming the expiration-date stock price is $100, which is greater than the strike price, the payoff is ($100-$65)*100*138 = $483,000. Similarly, the payoffs from other call option series can be calculated. The sum of these payoffs is the total value of July IBM calls on the expiration date. Table 3.2 shows the result when the expiration-date stock price is $100.

As can be seen in Table 3.2, the total value of July calls is $56,438,785 at the quoted date. If this money earns the risk-free interest rate, its value on the expiration date would be $56,438,785 * 1.014488 = $57,256,470. The total value of the call options must be equal to $57,256,470 on the expiration date for the call holders to break even. With the expiration-date stock price of $100, the total value of calls is much less than the goal. Because payoff from calls is an increasing function of stock price, the expiration-date stock price must have a higher value than the trial value of $100. After trying different stock prices, the stock price of $111.876 gives a total call value of $57,256,639, which is greater than $57,256,470. Because the precision of stock prices is set to 0.001, the next low value to be tried is $111.875. This yields a total call value of $57,252,625. The total value of the call options is less than $57,256,470 again. Therefore, the expiration-date stock price of $111.876 is considered to be the breakeven stock price for July IBM calls. Table 3.3 shows the result with the stock price of $111.876.

Strike Price	Call Price	Open Interest	Cost (CP*OI)	Payoff
65.0	35.200	138	485760	483000
70.0	31.200	293	914160	879000
75.0	25.500	651	1660050	1627500
80.0	23.000	9097	20923100	18194000
85.0	19.800	1942	3845160	2913000
90.0	15.400	971	1495340	971000
95.0	11.700	1641	1919970	820500
100.0	9.300	3456	3214080	0
105.0	7.200	11298	8134560	0
110.0	5.500	10655	5860250	0
115.0	4.100	4416	1810560	0
120.0	2.800	10229	2864120	0
125.0	1.950	9055	1765725	0
130.0	1.400	4892	684880	0
135.0	1.100	3337	367070	0
140.0	0.800	6175	494000	0
		Total:	56438785	25888000

Table 3.2 Payoff from July IBM calls with a trial stock price of $100

Strike Price	Call Price	Open Interest	Cost (CP*OI)	Payoff
65.0	35.200	138	485760	646888
70.0	31.200	293	914160	1226967
75.0	25.500	651	1660050	2400628
80.0	23.000	9097	20923100	28997597
85.0	19.800	1942	3845160	5219319
90.0	15.400	971	1495340	2124160
95.0	11.700	1641	1919970	2769352
100.0	9.300	3456	3214080	4104346
105.0	7.200	11298	8134560	7768505
110.0	5.500	10655	5860250	1998878
115.0	4.100	4416	1810560	0
120.0	2.800	10229	2864120	0
125.0	1.950	9055	1765725	0
130.0	1.400	4892	684880	0
135.0	1.100	3337	367070	0
140.0	0.800	6175	494000	0
		Total:	56438785	57256639

Table 3.3 Payoff from July IBM calls with a trial stock price of $111.876

3.2.2 Breakeven Stock Price for Puts

The approach to calculate a breakeven stock price for puts is similar to the one for calls except for a few minor differences. Table 3.4 shows the option quotes of July 2001 IBM puts.

Strike Price	Put Price	Open Interest
65.0	1.450	665
70.0	2.100	6601
75.0	2.600	4819
80.0	3.700	5654
85.0	5.000	5447
90.0	6.400	7109
95.0	8.400	4077
100.0	10.500	7081
105.0	13.800	5440
110.0	16.700	2979
115.0	19.900	3312
120.0	24.100	6490
125.0	28.100	457
130.0	32.200	182
135.0	37.400	175
140.0	42.300	149

Table 3.4 Option quotes of July IBM puts on April 6, 2001 from www.moneynet.com

The cost of buying puts with the strike price of \$65 is \$1.45 * 100 * 665 = \$96,425. In the same manner, the cost of buying other put option series can be calculated. The sum of these costs is the total value of July IBM puts.

The payoffs from puts with the strike price of \$65 on the expiration date also depend on the expiration-date stock price. For illustration purposes, a trial stock price of \$100 is used again, which is greater than the strike price of \$65. So there is no profit from this option series. For puts with the strike price of \$110, the payoff is (\$110-\$100)*100*2979 = \$2,979,000. Using the same method, the payoffs from other put option series can be calculated. The sum of these payoffs is the total value of July IBM puts at expiration. Table 3.5 shows the result when the expiration-date stock price is $100.

Strike Price	Put Price	Open Interest	Cost (CP*OI)	Payoff
65.0	1.450	665	96425	0
70.0	2.100	6601	1386210	0
75.0	2.600	4819	1252940	0
80.0	3.700	5654	2091980	0
85.0	5.000	5447	2723500	0
90.0	6.400	7109	4549760	0
95.0	8.400	4077	3424680	0
100.0	10.500	7081	7435050	0
105.0	13.800	5440	7507200	2720000
110.0	16.700	2979	4974930	2979000
115.0	19.900	3312	6590880	4968000
120.0	24.100	6490	15640900	12980000
125.0	28.100	457	1284170	1142500
130.0	32.200	182	586040	546000
135.0	37.400	175	654500	612500
140.0	42.300	149	630270	596000
		Total:	60829435	26544000

Table 3.5 Payoff from July IBM puts with a trial stock price of $100

Table 3.5 shows that the total value of the puts is $60,829,435 at the quoted date. If this money earns the risk-free interest rate, the value of this money at expiration would be $60,829,435 * 1.014488 = $61,710,731. The total value of the puts on the expiration date must be equal to $61,710,731 for put holders to breakeven. With the expiration-date stock price of $100, the total profit is much less than the goal. Because payoff from puts is a decreasing function of stock price, a smaller expiration-date stock price should be chosen to meet the goal. After trying different stock prices, the expiration-date stock price of $88.167 gives a total put value of $61,712,268, which is bigger than $61,710,731 and the closest to it. Therefore, the stock price of $88.167 is considered to be the breakeven stock price for July IBM puts. Table 3.6 shows the result with the expiration-date stock price of $88.167.

Strike Price	Put Price	Open Interest	Cost (CP*OI)	Payoff
65.0	1.450	665	96425	0
70.0	2.100	6601	1386210	0
75.0	2.600	4819	1252940	0
80.0	3.700	5654	2091980	0
85.0	5.000	5447	2723500	0
90.0	6.400	7109	4549760	1303080
95.0	8.400	4077	3424680	2785814
100.0	10.500	7081	7435050	8378947
105.0	13.800	5440	7507200	9157152
110.0	16.700	2979	4974930	6504051
115.0	19.900	3312	6590880	8887090
120.0	24.100	6490	15640900	20659617
125.0	28.100	457	1284170	1683268
130.0	32.200	182	586040	761360.6
135.0	37.400	175	654500	819577.5
140.0	42.300	149	630270	772311.7
		Total	60829435	61712268

Table 3.6 Payoff from July IBM puts with a trial stock price of $88.167

Chapter 4

Design and Implementation of BSP

The BSP system is a tool that can be used to retrieve specified stock option quotes and calculate breakeven stock prices for calls and puts. In this chapter, first, the data source that BSP uses to retrieve the data and conduct the analysis is presented. Then, the system design and implementation are described in considerable detail. Finally, some extensions of BSP are also presented.

4.1 Data Source and Excel Web Query

To calculate breakeven stock prices using the approach described in the last chapter, data, such as quoted option price, open interest and strike price, are needed. The best data source is daily option quotes. Option quotes can be obtained from the financial section of a daily newspaper. However, this is not very useful to a computer program as it is not in an electronic format. Before the popularization of the Internet, it was difficult to find such data for the public outside of large financial and government organizations. This obstacle was also the reason why analysis on option markets was difficult to conduct. With the advent of the Internet, things have changed dramatically and now free, delayed option quotes can be found on the Internet. This makes our study possible.

The web pages seen through web browsers are actually HTML files, which not only include the data which can be seen, but also plenty of information about the presentation of the data in the form of tags. To analyze the data on a web page, the HTML file has to be parsed in order to split tags and real data, and the raw data returned in a meaningful format. Because HTML tags include only presentation information and do not include any information on the meaning of the data they mark, the raw data can only be understood according to its position in the file. This makes data on web pages quite difficult to analyze.

Fortunately, we do not need to write a program to parse HTML files. There is a lot of software available to extract data from a web page. One is MS Excel. Excel provides a built-in function, Web query, which can retrieve the data on a web page and return it to an Excel worksheet. A Web query only extracts the raw data as seen in browsers, and not the entire contents of the HTML file. It displays the data on a worksheet, inserting different fields into different cells. Thus, a Web query provides a solution to the problem of data acquisition. In addition, in the 2000 version, Excel enhances this function by enabling users to retrieve only part of the data on a web page instead of all of the data on that page. Figure 4.1 is a screen shot of Web query dialog window.

An Excel Web query provides users great flexibility when retrieving data. As can be seen in Figure 4.1, after giving the address of the web page containing the interesting data, a user has choices to retrieve the entire page or only one or more specific tables on the web page. He can also decide how much formatting to keep. Tables, defined with the <TABLE> tags in HTML files, are often used by web designers to organize the content of a web page. Option quotes are presented in a tabulated format, which usually utilizes tables. Thus, a Web query can be effectively used to retrieve the quotes. Only the tables containing option quotes need to be fetched, and not the entire web page on which the quotes appear. The advantage of retrieving a specific table over an entire page is that the changes on the other parts of the web page have no effect on the data in the table. Moreover, a Web query can map every field in a table to a cell on a worksheet. This makes the output very organized and understandable. Not much cleanup work is necessary.

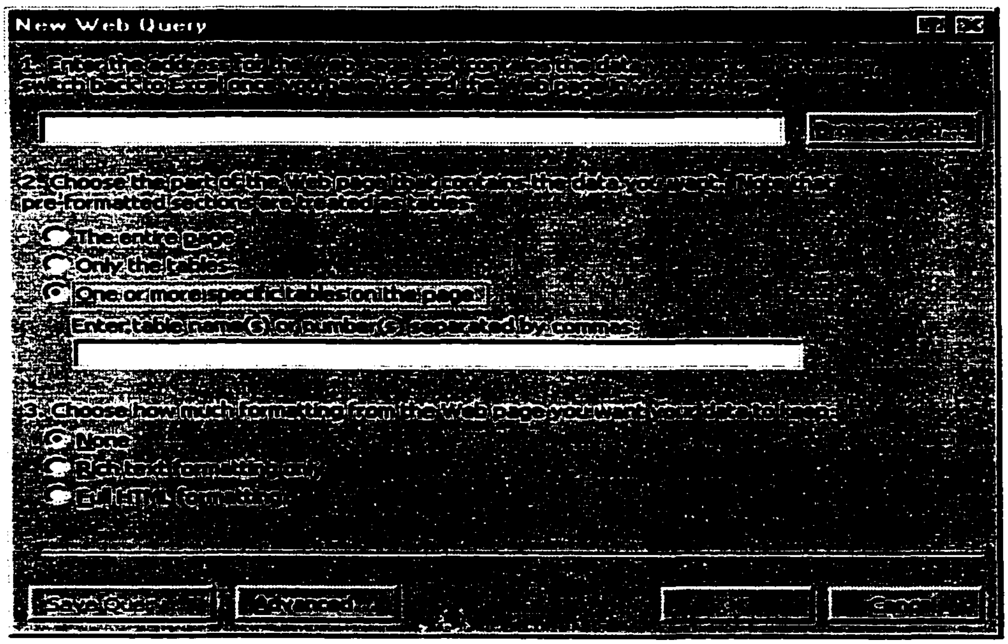


Figure 4.1 Web query in Excel 2000

For the purpose of this study, any website which provides accurate and completed daily option quotes can be considered as a data source. But, another issue is also important when choosing a website, and that is, using fewer Web queries to get required data. BSP is designed to get all available option quotes on one stock with different expiration dates in order to calculate as many breakeven stock prices as possible. Some websites present the quotes on different web pages according to the option's expiration dates. To get all the available quotes, multiple web pages need to be fetched from the website. This means that several Web queries have to be initiated. Processing Web queries takes relatively longer than processing data. In order to improve the performance of the system, the number of Web queries has to be limited to as few as possible. For this reason, websites which can provide the option quotes on one stock in bulk are the preferred choices. The financial website of Reuters, www.moneynet.com, is one of them and is the one BSP uses. This website allows users to query option quotes with expiration dates from the current month up to 4 years later. Quotes for one option class are returned in one web page. Therefore, on one stock, all option quotes can be obtained with two Web queries, one for calls, the other for puts.

4.2 Overview of BSP

BSP is an Excel macro developed in the Microsoft Visual Basic for Applications (VBA) programming language. Macros can automate complex tasks on Excel objects by storing a series Excel commands that, later, users can execute as a single command. An Excel file is in fact an Excel workbook, which contains worksheets. On a worksheet, there are many cells, each referred to by its row number and column number. Excel not only provides a convenient approach to retrieve data from a web page, but also is commonly used to maintain and manage financial data because Excel programs are simple to operate and require little preparation before data entry, compared to the setup of database tables [3]. In addition, Excel imposes a strict row and column format on the data, making data easy to refer to and to organize. Excel provides capabilities for data computation and analysis as well. Because of all these features, BSP is developed in Excel.

BSP downloads option quotes on as many stocks as a user requests. The option quotes on each stock are displayed on a separate worksheet. These worksheets are named with the symbol of the stock on which the options trade. On the last sheet, named "result", the percentage difference of breakeven stock prices from current stock prices for all stocks are displayed, one stock per row. The program can only be run using Excel 2000 or later versions.

The functionality of BSP can be divided into two independent parts: data acquisition and data analysis, as shown in Figure 4.2. The first part is the basis of the second part. It collects daily option quotes from the Web, transforms the data, and prepares the data for the analysis in the second part. The second part is the essence of BSP; it extracts the data, conducts the analysis, computes the breakeven stock prices, generalizes the results and presents them in an understandable format.

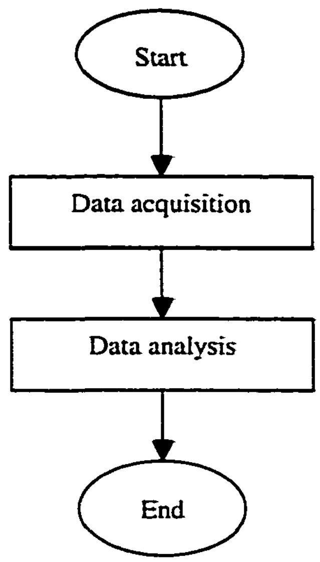


Figure 4.2 Overview of BSP

4.3 Data Acquisition

This part of the program retrieves the desired option quotes from the Web and prepares the data to be analyzed. The structure of the data acquisition is illustrated in Figure 4.3.

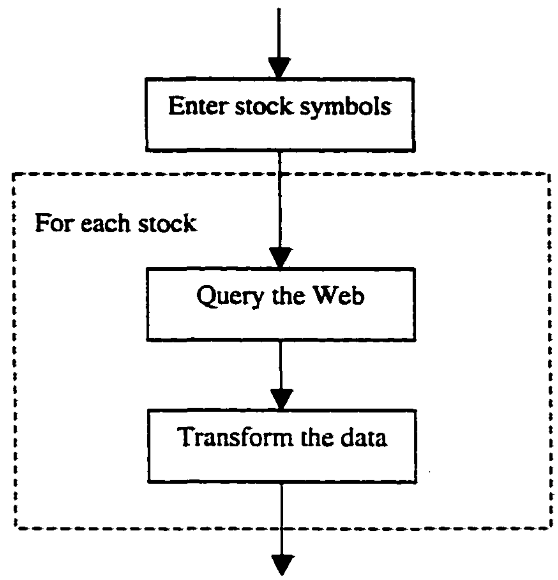


Figure 4.3 Structure of data acquisition

4.3.1 Enter Stock Symbols

To obtain option quotes, the program needs to know the symbols of the stocks on which the options trade. Figure 4.4 displays the window designed to prompt users for this data.

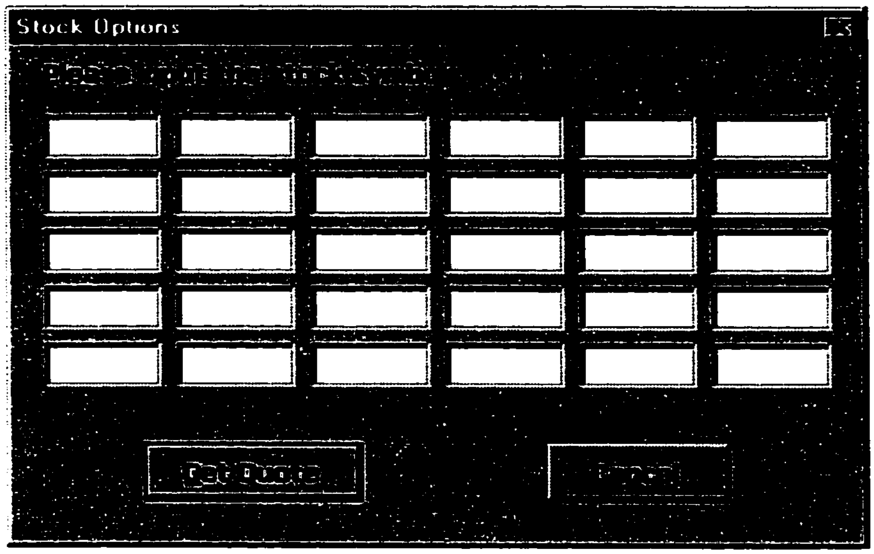


Figure 4.4 Window to input stock symbols

The program allows a user to select up to 30 stocks. A user can enter the symbols in any field. When implementing this function, a string array "symbols" is used to store the input symbols.

Upon clicking on the **Get Quote** button, the program will proceed to query the web for option quotes on every selected stock. If one clicks on the **Cancel** button, the program will terminate.

4.3.2 Query the Web

Prior to executing the Web query, the program must first know the address of the web page from which the quotes will be extracted, as well as the numbers of the quote tables needed. The following example is the address of the web page containing the option quotes on IBM with an expiration date from April 2001 to May 2001:

```
http://www.moneynet.com/content/MONEYNET/OptionChain/OCQuote.asp
?ExtraParms=&OCReturnURL=&SYMBOL=ibm&MinMonth=04&MinYear=2001
&MinPrice=&Type=CALL&MaxMonth=05&MaxYear=2001&MaxPrice=
```

It can be seen, by changing the stock symbol, option class, and the start month and the end month of the query period, the quotes on other stocks or in other time periods can be extracted. Stock symbols are obtained from users in the window shown in Figure 4.4. Option classes are call and put. The time period is set from the current month to the December of three years later, which is long enough to cover all expiration dates. In addition, the table numbers for stock price and option quotes are 7 and 9, respectively. No formatting is kept since it is irrelevant for the computing of the data.

The program forms a Web query for every option class and executes it to extract the quotes. Because of the default setting, Excel takes fractions as dates. For example, Excel will interpret "1/8" as "8-Jan". To correct this error, the WebDisableDateRecognition property of a Web query is set to "true."

The table containing a stock price has more information than the stock price, as shown in Figure 4.5. This extra information is irrelevant to the study. For the presentation purposes, the stock price is preserved and the other fields are deleted.

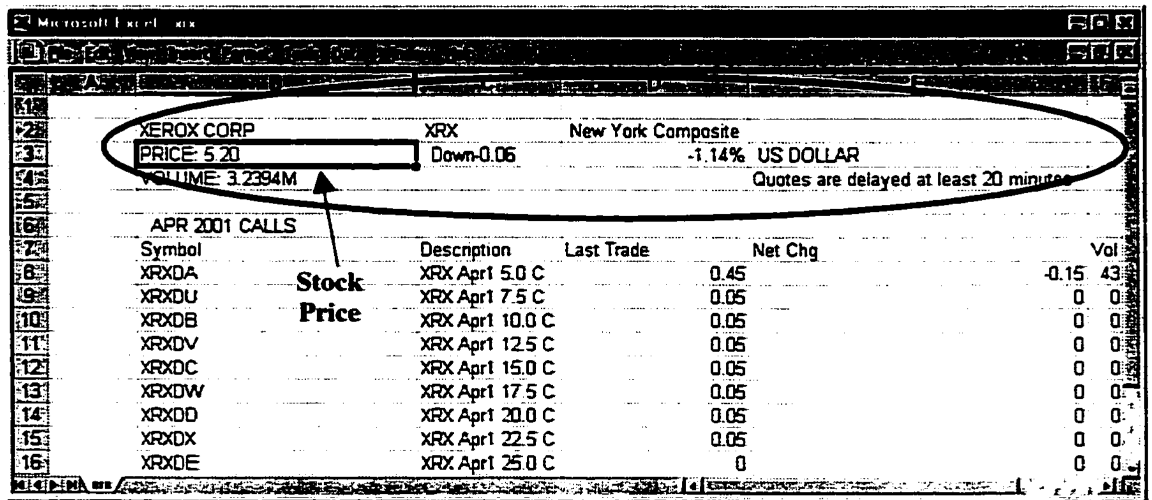
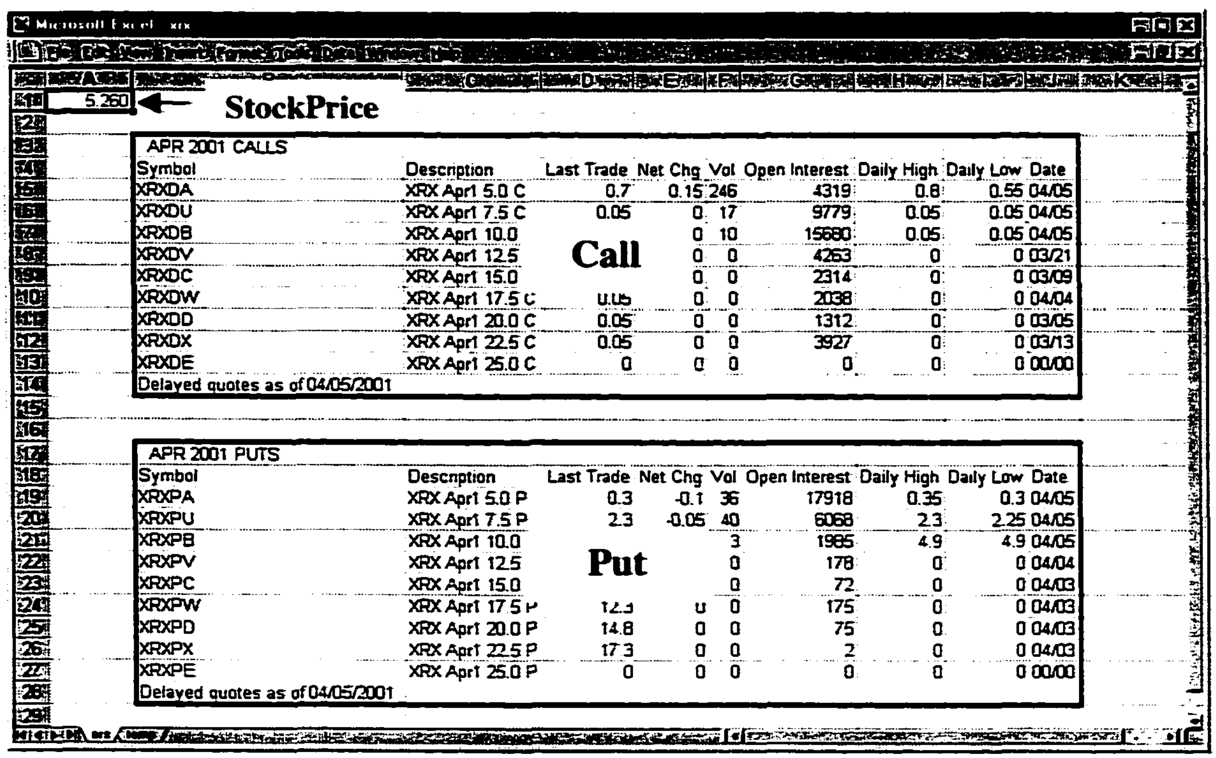


Figure 4.5 Stock price cleanups

The returned option quotes on one stock are stored on one worksheet, which is given the name of the stock symbol. On the worksheet, the cells storing the stock price, the quotes for calls and the quotes for puts are given the name of **StockPrice**, **Call**, and **Put** respectively, as shown in Figure 4.6.

Excel refers to data according to its position: the row number and the column number. Consequently, if the worksheet layout is modified in any way, for example, rows are inserted and deleted, the majority of cell references in the code would have to be changed accordingly. However, a name is associated with the data, rather than the position of the data. Naming data gives the added advantage of allowing a program to refer to the data by its name, with no regard to its location on the worksheet. If the name is for an area of data, e.g. **Call** in Figure 4.6, data can be referred by the name and the relative row and column numbers in the named area. Therefore, naming data makes the program easier to maintain.



APR 2001 CALLS

Symbol	Description	Last Trade	Net Chg	Vol	Open Interest	Daily High	Daily Low	Date
XRXDA	XRX Apr1 5.0 C	0.7	0.15	246	4319	0.8	0.55	04/05
XRXDU	XRX Apr1 7.5 C	0.05	0	17	9779	0.05	0.05	04/05
XRXDB	XRX Apr1 10.0		0	10	15680	0.05	0.05	04/05
XRXDV	XRX Apr1 12.5		0	0	4263	0	0	03/21
XRXDC	XRX Apr1 15.0		0	0	2314	0	0	03/09
XRXDW	XRX Apr1 17.5 C	[illegible]	0	0	2038	0	0	04/04
XRXDD	XRX Apr1 20.0 C	0.05	0	0	1312	0	0	03/05
XRXDX	XRX Apr1 22.5 C	0.05	0	0	3927	0	0	03/13
XRXDE	XRX Apr1 25.0 C	0	0	0	0	0	0	00/00

Delayed quotes as of 04/05/2001

APR 2001 PUTS

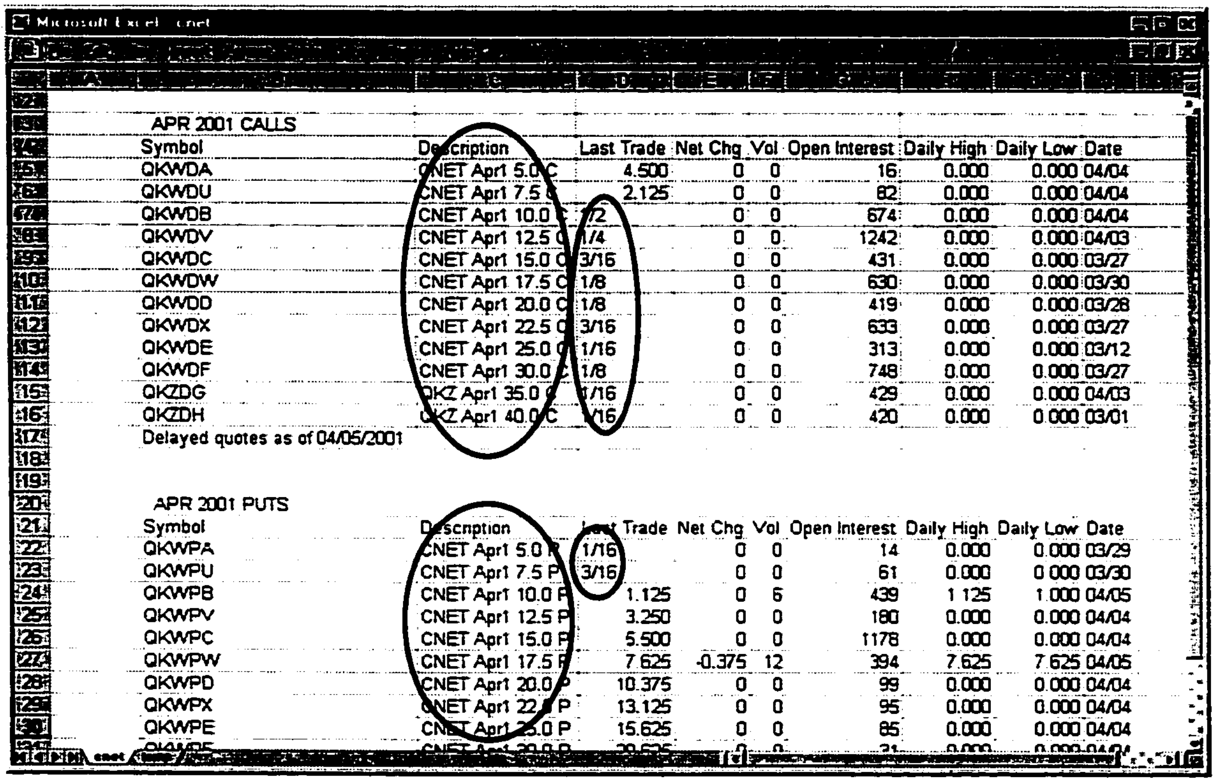
Symbol	Description	Last Trade	Net Chg	Vol	Open Interest	Daily High	Daily Low	Date
XRXPA	XRX Apr1 5.0 P	0.3	-0.1	36	17918	0.35	0.3	04/05
XRXPU	XRX Apr1 7.5 P	2.3	-0.05	40	6068	2.3	2.25	04/05
XRXPB	XRX Apr1 10.0			3	1985	4.9	4.9	04/05
XRXPV	XRX Apr1 12.5			0	178	0	0	04/04
XRXPC	XRX Apr1 15.0			0	72	0	0	04/03
XRXPW	XRX Apr1 17.5 P	[illegible]	[illegible]	0	175	0	0	04/03
XRXPD	XRX Apr1 20.0 P	14.8	0	0	75	0	0	04/03
XRXPX	XRX Apr1 22.5 P	17.3	0	0	2	0	0	04/03
XRXPE	XRX Apr1 25.0 P	0	0	0	0	0	0	00/00

Delayed quotes as of 04/05/2001

Figure 4.6 Quote sheet naming scheme

4.3.3 Data Transformation

The quotes from the Web are quite integrated and consistent. Not much cleanup work on the content is needed. However, as shown in Figure 4.7, the data is not ready for programs to process as yet.



APR 2001 CALLS

Symbol	Description	Last Trade	Net Chg	Vol	Open Interest	Daily High	Daily Low	Date
QKWDA	CNET Apr1 5.0 C	4.500	0	0	16	0.000	0.000	04/04
QKWDU	CNET Apr1 7.5 C	2.125	0	0	82	0.000	0.000	04/04
QKWDB	CNET Apr1 10.0 C	1/2	0	0	674	0.000	0.000	04/04
QKWDV	CNET Apr1 12.5 C	1/4	0	0	1242	0.000	0.000	04/03
QKWDC	CNET Apr1 15.0 C	3/16	0	0	431	0.000	0.000	03/27
QKWDW	CNET Apr1 17.5 C	1/8	0	0	630	0.000	0.000	03/30
QKWDD	CNET Apr1 20.0 C	1/8	0	0	419	0.000	0.000	03/28
QKWDX	CNET Apr1 22.5 C	3/16	0	0	633	0.000	0.000	03/27
QKWDE	CNET Apr1 25.0 C	1/16	0	0	313	0.000	0.000	03/12
QKWDF	CNET Apr1 30.0 C	1/8	0	0	748	0.000	0.000	03/27
QKZDG	QKZ Apr1 35.0 C	1/16	0	0	429	0.000	0.000	04/03
QKZDH	QKZ Apr1 40.0 C	1/16	0	0	420	0.000	0.000	03/01

Delayed quotes as of 04/05/2001

APR 2001 PUTS

Symbol	Description	Last Trade	Net Chg	Vol	Open Interest	Daily High	Daily Low	Date
QKWPA	CNET Apr1 5.0 P	1/16	0	0	14	0.000	0.000	03/29
QKWPU	CNET Apr1 7.5 P	3/16	0	0	61	0.000	0.000	03/30
QKWPB	CNET Apr1 10.0 P	1.125	0	6	439	1.125	1.000	04/05
QKWPV	CNET Apr1 12.5 P	3.250	0	0	180	0.000	0.000	04/04
QKWPC	CNET Apr1 15.0 P	5.500	0	0	1178	0.000	0.000	04/04
QKWPW	CNET Apr1 17.5 P	7.625	-0.375	12	394	7.625	7.625	04/05
QKWPD	CNET Apr1 20.0 P	10.375	0	0	99	0.000	0.000	04/04
QKWPX	CNET Apr1 22.5 P	13.125	0	0	95	0.000	0.000	04/04
QKWPE	CNET Apr1 25.0 P	15.625	0	0	85	0.000	0.000	04/04

Figure 4.7 Data transformation

There are two format issues to be considered. One is that some fields in the quotes, such as *Last Trade* in the figure, are in the format of fractions. Because of the default setting of Excel, even though the cells have been formatted as numbers with the precision of 0.001, the queried data cannot be converted from fractions to decimals. For programs to calculate these numbers, extra conversion work is required to solve this problem.

The other concern is the strike price, an important variable in the computation of breakeven stock prices. In the quotes obtained, a strike price appears as a part of the big field *Description*, which also includes an exercise month and an option class. Before carrying on the analysis, the fields in the column *Description* have to be separated. By using the Excel function of Text to Columns, the column *Description* is split into four columns: *Description*, *Month*, *Strike Price*, and *C/P*. However, the original field *Description* is not perfectly constructed and so extra cleanup work is still needed to obtain the required output from the Text to Columns function.

4.4 Data Analysis

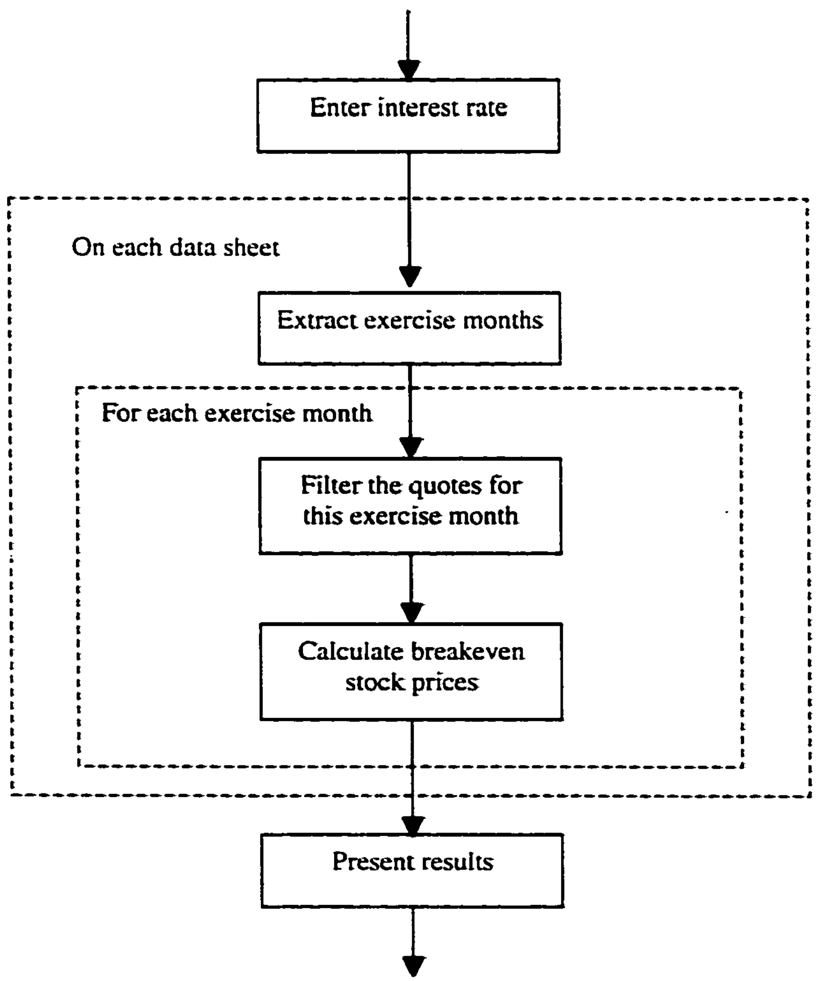


Figure 4.8 Structure of data analysis

After data acquisition and data cleanup, the data is ready to be processed by a program. So the system proceeds to the second stage, data analysis. In this stage, the program first extracts the exercise months with which options currently trade, then filters the quotes for every exercise month, calculates the breakeven prices for both calls and puts, and then finally generalizes and presents the results. The two worksheets, "call" and "put," are added temporarily to the workbook to store the calculated results. The structure of the data analysis is shown in the Figure 4.8.

4.4.1 Enter Interest Rate

To simplify the calculation, the risk-free interest rate mentioned in chapter 3, is assumed to be same for all maturities. This interest rate is used to calculate the future value of current money and must be obtained first. The program uses a default value of 5%, but also gives users an opportunity to enter what they observed from the markets. A dialog window, as shown in Figure 4.9, asks for the input from users.

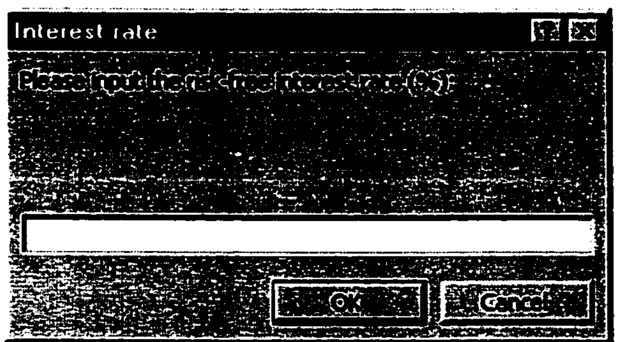


Figure 4.9 Window to input the interest rate

The system validates the input from a user. If the input is not a number, an error message pops up. Otherwise, after clicking on the **OK** button, the system takes the rate entered and resumes the execution. If the **Cancel** button is selected, the system will use the default rate of 5% instead and display a message window, as in Figure 4.10, to inform the user as well.

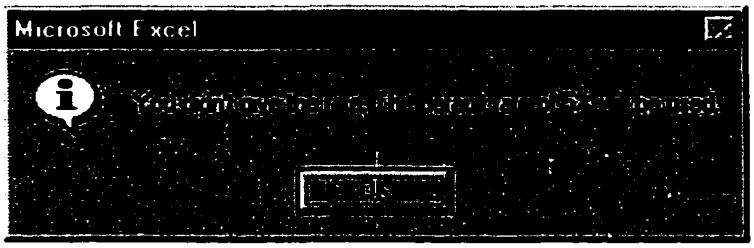


Figure 4.10 Message window about the interest rate

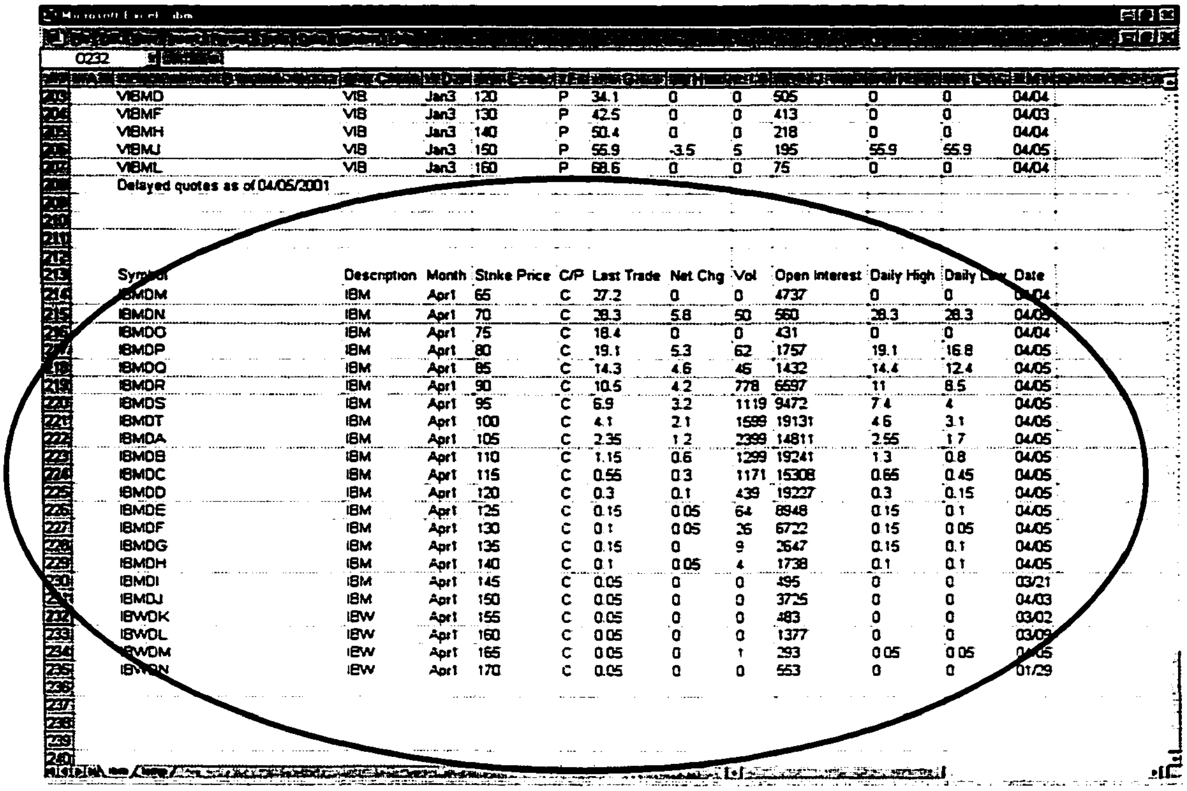
4.4.2 Extract Exercise Months

On one stock option quotes sheet, all quotes with different exercise months are combined together. To calculate a breakeven stock price, the quotes with the same exercise month need to be extracted first. Therefore, it has to be known with which exercise months the options are trading. The program browses the quotes and records the different exercise months. Since options trade on different cycles, an option in the January cycle trades with different exercise months than the one in the February cycle. So the program has to perform this extraction on every quote sheet.

4.4.3 Filter Option Quotes

Having the exercise months, the program can extract the quotes with each of these months from the bulk of quotes in order to perform the calculation. To implement this function, the program uses the Excel function of Advanced Filter and copies the filtered quotes at the end of worksheet, five rows after the **Put** area. The conditions for Advanced

Filter are exercise months and option classes. The Figure 4.11 below is an example of quotes extraction. The filter source is the quotes for options on IBM on April 5, 2001. The conditions for the filtering are "Apr1" and "C".



203	VIBMD	VIB	Jan3	120	P	34.1	0	0	505	0	0	04/04
204	VIBMF	VIB	Jan3	130	P	42.5	0	0	413	0	0	04/03
205	VIBMH	VIB	Jan3	140	P	50.4	0	0	218	0	0	04/04
206	VIBMJ	VIB	Jan3	150	P	55.9	-3.5	5	195	55.9	55.9	04/05
207	VIBML	VIB	Jan3	160	P	68.6	0	0	75	0	0	04/04
208	Delayed quotes as of 04/05/2001											
209												
210												
211												
212												
213	Symbol	Description	Month	Strike Price	C/P	Last Trade	Net Chg	Vol	Open Interest	Daily High	Daily Low	Date
214	IBMDM	IBM	Apr1	65	C	27.2	0	0	4737	0	0	04/04
215	IBMDN	IBM	Apr1	70	C	28.3	5.8	50	560	28.3	28.3	04/05
216	IBMDO	IBM	Apr1	75	C	18.4	0	0	431	0	0	04/04
217	IBMDP	IBM	Apr1	80	C	19.1	5.3	62	1757	19.1	16.8	04/05
218	IBMDQ	IBM	Apr1	85	C	14.3	4.6	46	1432	14.4	12.4	04/05
219	IBMDR	IBM	Apr1	90	C	10.5	4.2	778	6697	11	8.5	04/05
220	IBMDS	IBM	Apr1	95	C	6.9	3.2	1119	9472	7.4	4	04/05
221	IBMDT	IBM	Apr1	100	C	4.1	2.1	1599	19131	4.6	3.1	04/05
222	IBMDA	IBM	Apr1	105	C	2.35	1.2	2399	14811	2.55	1.7	04/05
223	IBMDB	IBM	Apr1	110	C	1.15	0.6	1299	19241	1.3	0.8	04/05
224	IBMDC	IBM	Apr1	115	C	0.55	0.3	1171	15308	0.65	0.45	04/05
225	IBMDD	IBM	Apr1	120	C	0.3	0.1	439	19227	0.3	0.15	04/05
226	IBMDE	IBM	Apr1	125	C	0.15	0.05	64	8948	0.15	0.1	04/05
227	IBMDF	IBM	Apr1	130	C	0.1	0.05	36	6722	0.15	0.05	04/05
228	IBMDG	IBM	Apr1	135	C	0.15	0	9	2647	0.15	0.1	04/05
229	IBMDH	IBM	Apr1	140	C	0.1	0.05	4	1738	0.1	0.1	04/05
230	IBMDI	IBM	Apr1	145	C	0.05	0	0	495	0	0	03/21
231	IBMDJ	IBM	Apr1	150	C	0.05	0	0	3725	0	0	04/03
232	IBWDK	IBW	Apr1	155	C	0.05	0	0	483	0	0	03/02
233	IBWDL	IBW	Apr1	160	C	0.05	0	0	1377	0	0	03/09
234	IBWDM	IBW	Apr1	165	C	0.05	0	1	293	0.05	0.05	04/05
235	IBWDN	IBW	Apr1	170	C	0.05	0	0	553	0	0	01/29

Figure 4.11 Quotes extraction for "Apr1" IBM calls

4.4.4 Calculate Breakeven Stock Prices

As introduced in Chapter 3, to calculate a breakeven stock price, the values of options at the quoted date and the payoffs from the options at expiration must be obtained first. Three new columns, *Cost*, *Payoff*, and *Stock Price*, are added to the filtered area for this purpose. In addition, a cell for the sum of Cost and another for the sum of Payoff are added. The formulas of these new cells are shown in Figure 4.12. Due to space limitations, only the relevant columns are shown. The others are hidden.

New Working Columns

Microsoft Excel - ibm

	E	G	J	N	O	P
212						
213	Strike Price	Last Trade	Open Interest	Cost	Payoff	Stock Price
214	65	27.2	4737	=Last Trade * 100 * Open Interest	=IF(Stock Price-Strike Price>0, (Stock Price-Strike Price) * 100 * Open Interest, 0)	117.5
215	70	28.3	560	=Last Trade * 100 * Open Interest	=IF(Stock Price-Strike Price>0, (Stock Price-Strike Price) * 100 * Open Interest, 0)	=$P$214
216	75	18.4	431	=Last Trade * 100 * Open Interest	=IF(Stock Price-Strike Price>0, (Stock Price-Strike Price) * 100 * Open Interest, 0)	=$P$214
217	80	19.1	1757	=Last Trade * 100 * Open Interest	=IF(Stock Price-Strike Price>0, (Stock Price-Strike Price) * 100 * Open Interest, 0)	=$P$214
218	85	14.3	1432	=Last Trade * 100 * Open Interest	=IF(Stock Price-Strike Price>0, (Stock Price-Strike Price) * 100 * Open Interest, 0)	=$P$214
219	90	10.6	6597	=Last Trade * 100 * Open Interest	=IF(Stock Price-Strike Price>0, (Stock Price-Strike Price) * 100 * Open Interest, 0)	=$P$214
220	95	6 9	9472	=Last Trade * 100 * Open Interest	=IF(Stock Price-Strike Price>0, (Stock Price-Strike Price) * 100 * Open Interest, 0)	=$P$214
221	100	4.1	19131	=Last Trade * 100 * Open Interest	=IF(Stock Price-Strike Price>0, (Stock Price-Strike Price) * 100 * Open Interest, 0)	=$P$214
222	105	2.35	14811	=Last Trade * 100 * Open Interest	=IF(Stock Price-Strike Price>0, (Stock Price-Strike Price) * 100 * Open Interest, 0)	=$P$214
223	110	1.16	19241	=Last Trade * 100 * Open Interest	=IF(Stock Price-Strike Price>0, (Stock Price-Strike Price) * 100 * Open Interest, 0)	=$P$214
224	115	0.65	15308	=Last Trade * 100 * Open Interest	=IF(Stock Price-Strike Price>0, (Stock Price-Strike Price) * 100 * Open Interest, 0)	=$P$214
225	120	0.3	19227	=Last Trade * 100 * Open Interest	=IF(Stock Price-Strike Price>0, (Stock Price-Strike Price) * 100 * Open Interest, 0)	=$P$214
226	125	0.16	8948	=Last Trade * 100 * Open Interest	=IF(Stock Price-Strike Price>0, (Stock Price-Strike Price) * 100 * Open Interest, 0)	=$P$214
227	130	0.1	6722	=Last Trade * 100 * Open Interest	=IF(Stock Price-Strike Price>0, (Stock Price-Strike Price) * 100 * Open Interest, 0)	=$P$214
228	135	0.16	2647	=Last Trade * 100 * Open Interest	=IF(Stock Price-Strike Price>0, (Stock Price-Strike Price) * 100 * Open Interest, 0)	=$P$214
229	140	0.1	1738	=Last Trade * 100 * Open Interest	=IF(Stock Price-Strike Price>0, (Stock Price-Strike Price) * 100 * Open Interest, 0)	=$P$214
230	145	0.05	495	=Last Trade * 100 * Open Interest	=IF(Stock Price-Strike Price>0, (Stock Price-Strike Price) * 100 * Open Interest, 0)	=$P$214
231	160	0.05	3725	=Last Trade * 100 * Open Interest	=IF(Stock Price-Strike Price>0, (Stock Price-Strike Price) * 100 * Open Interest, 0)	=$P$214
232	165	0.05	483	=Last Trade * 100 * Open Interest	=IF(Stock Price-Strike Price>0, (Stock Price-Strike Price) * 100 * Open Interest, 0)	=$P$214
233	160	0.05	1377	=Last Trade * 100 * Open Interest	=IF(Stock Price-Strike Price>0, (Stock Price-Strike Price) * 100 * Open Interest, 0)	=$P$214
234	165	0.05	293	=Last Trade * 100 * Open Interest	=IF(Stock Price-Strike Price>0, (Stock Price-Strike Price) * 100 * Open Interest, 0)	=$P$214
235	170	0.05	553	=Last Trade * 100 * Open Interest	=IF(Stock Price-Strike Price>0, (Stock Price-Strike Price) * 100 * Open Interest, 0)	=$P$214
236				=SUM(N214:N235)	=SUM(O214:O235)	
237						

Total Value (TV) Total Payoff (TP) Expiration-date Stock Price

The breakeven stock price is the expiration-date stock price making $TV * e^{rT} = TP$.

Figure 4.12 Formulas of working cells

The example in Figure 4.12 is for call options. For put options, the formulas in the column of *Payoff* would be:

IF(Strike Price-Price>0, (Strike Price-Price) * 100 * Open Interest, 0).

The rest remains the same. As can be seen, the formulas use many column labels. To enable Excel to accept column labels in formulas, the corresponding property of the workbook is set to "true":

ActiveWorkbook.AcceptLabelsInFormulas = True.

The program then adjusts the value of *Expiration-date Stock Price* in order to fulfill the goal. The goal is to ensure that the value of *Total Payoff* (TP) is equal to or at most, slightly larger than the value of *Total Value* (TV) times e^{rT}. Here, r is the risk-free interest rate obtained from users and T is the time to expiration in terms of years. When the goal is reached, the value of *Expiration-date Stock Price* is the breakeven stock price for the options.

The algorithm used to adjust the value of *Expiration-date Stock Price* is the Method of Bisection. First, the lowest strike price is regarded as the lower bound of the breakeven stock price and the highest strike price is regarded as the upper bound of the breakeven stock price. Because of the distribution of the trading, these assignments are considered to be safe and reasonable. The midpoint between the lower bound and the upper bound is used as a trial stock price. Then according to the relationship of $TV * e^{rT}$ and TP with the trial stock price, the lower bound or the upper bound is replaced by the trial stock price. As a result, the price range is cut in half. This procedure continues with the new lower bound and upper bound until $TV * e^{rT} = TP$ or a relative precision of 0.001 is obtained.

The flow chart in Figure 4.13 illustrates the algorithm for stock price adjustment for calls. TV_I represents the value of *Total Value* (TV) times e^{rT}.

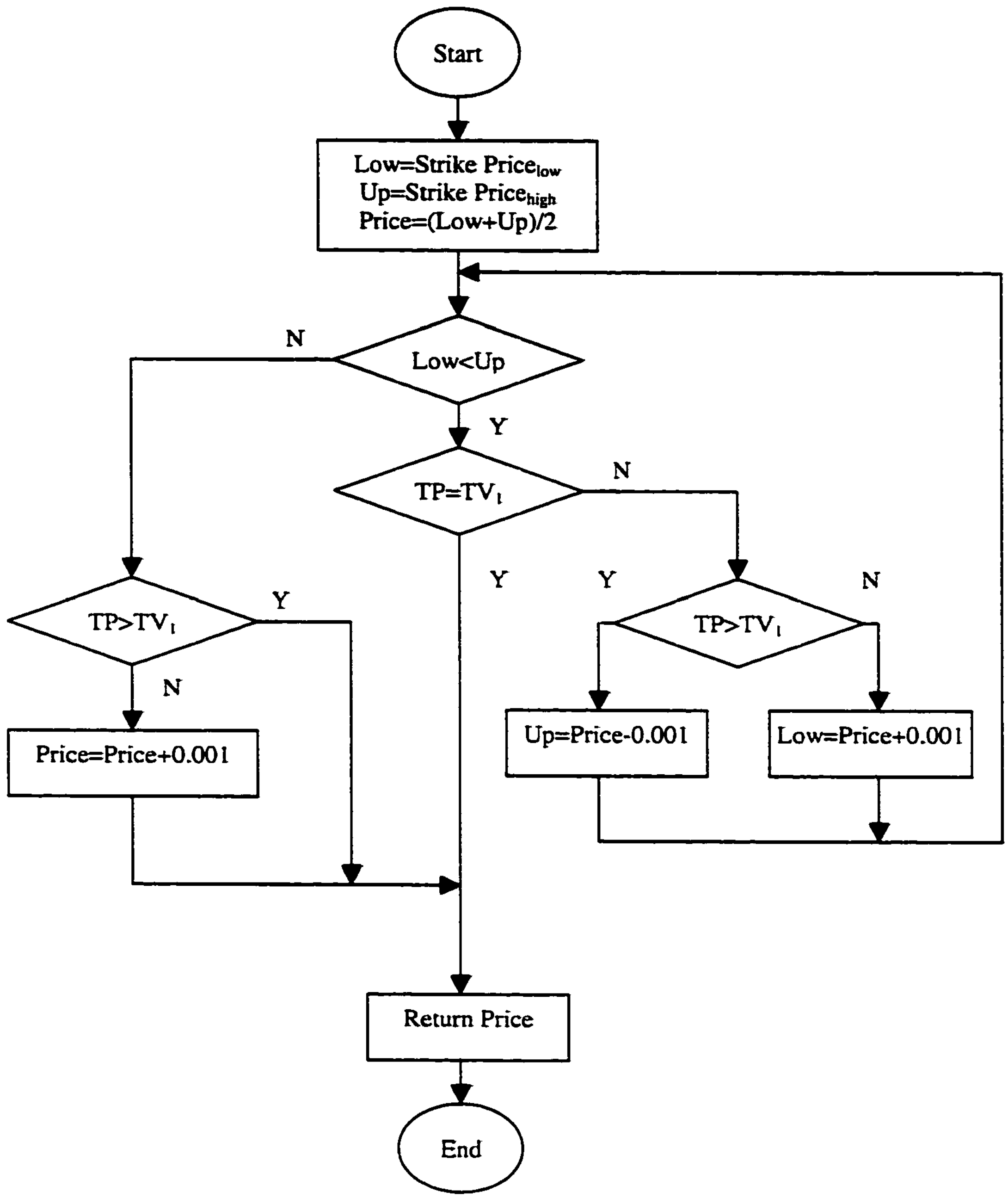


Figure 4.13 Algorithm for stock price adjustment for calls

Unlike the payoff from calls, which is an increasing function of stock prices, the one from puts is a decreasing function of stock prices. Therefore the algorithm used to adjust the stock price for puts is a little different from the one for calls, whose flow chart is shown in Figure 4.14.

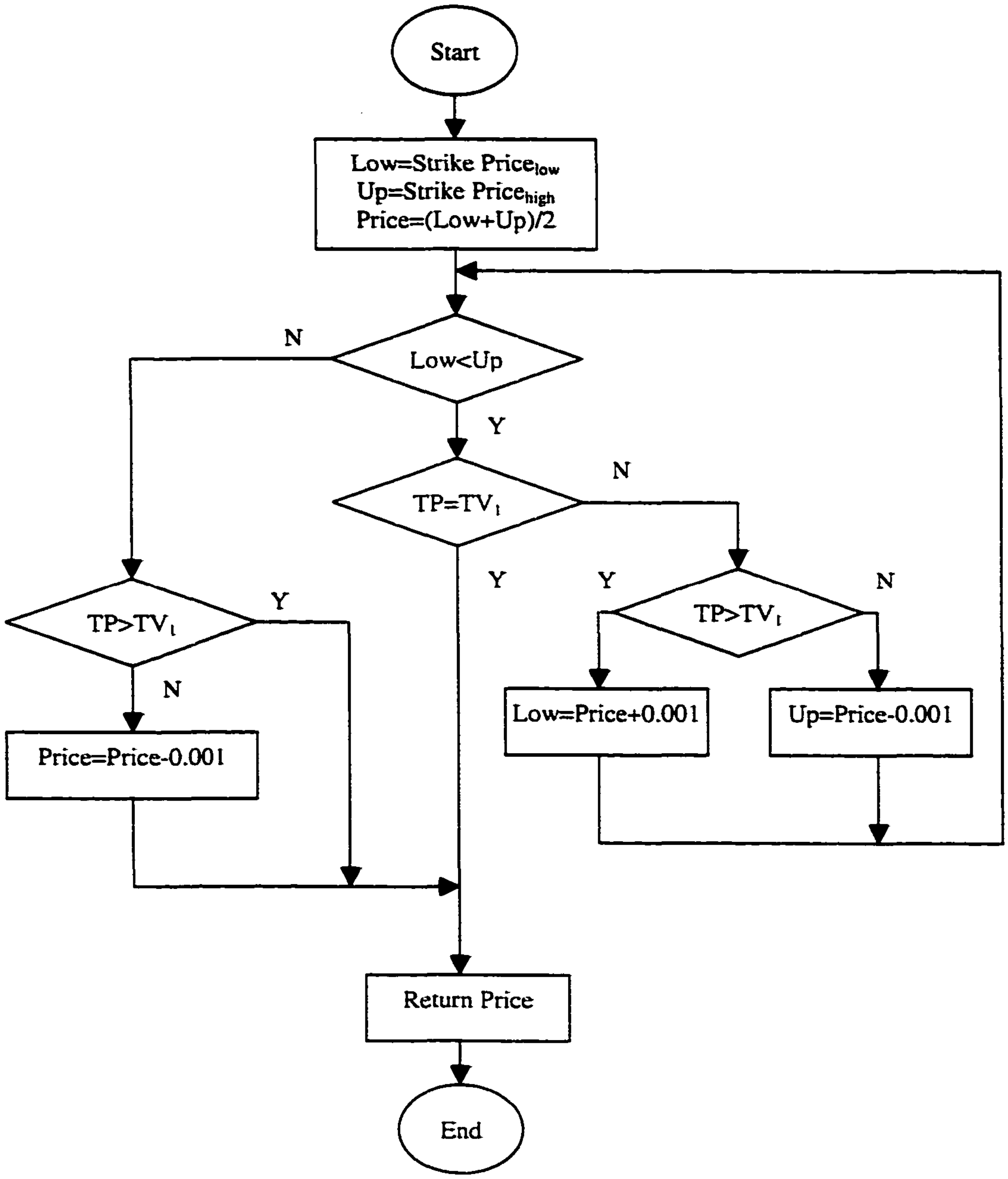


Figure 4.14 Algorithm for stock price adjustment for puts

4.4.5 Present Results

After calculating breakeven stock prices from option quotes with different exercise months, the system stores these prices on two different worksheets, "call" and "put", according to the source of the information. To present these breakeven stock prices, BSP uses the following layout on both sheets:

Company	Stock Price	Month1	Month2	Month3
Stock symbol	Current stock price	Breakeven price1	Breakeven price2	Breakeven price3

Table 4.1 "call" and "put" worksheet layouts

The breakeven stock prices for one stock are listed in one row. Even though there is a particular expiration day in an exercise month, the system still uses only months to present the expiration date, as is done in option quotes.

Breakeven stock prices are option holders' opinions about the underlying stocks. If a user enters many stocks in the window shown in Figure 4.4, option holders' opinions on every stock can be obtained at the same time by computing the breakeven stock prices. Comparing the holders' opinions about different stocks can reveal the most interesting information. Since the fluctuation of $1for the price of $80 has a very different meaning from the fluctuation of $1 for the price of $2, the system generalizes breakeven stock prices by calculating the percentages of the their differences from the current stock prices with respect to the current stock prices:

$$percentage = \frac{breakevenstockprice - currentstockprice}{currentstockprice} * 100 .$$

In theory, the percentages obtained for calls should be positive because call holders are optimistic about the markets, while the percentages obtained for puts should be negative because put holders are pessimistic about the markets.

To better present the results, two worksheets "call" and "put" are further combined into one, named "result," whose layout is shown in Table 4.2. Two columns of percentages are under one month title. The information for calls has a light turquoise background, so that it can be easily differentiated from the information for puts. Thus, users can easily see and compare the scope of changes in breakeven stock price for calls and puts away from the current stock prices. In addition, some formatting is done to facilitate easy reading, such as freezing the title rows and columns as well as enlarging the font size of cells.

Company	Stock prices	Month1		Month2		Month3	
Stock symbol	Current stock price	Percentage1 for calls	Percentage1 for puts	Percentage2 for calls	Percentage2 for puts	Percentage3 for calls	Percentage3 for puts

Table 4.2 "result" worksheet layout

4.5 An Alternative to BSP

BSP asks users to enter stock symbols in every execution. If a user analyzes options on certain stocks regularly, he may find this task to be quite tedious. To improve system performance, another approach is also implemented as an alternative to BSP, called Quick BSP (QBSP). QBSP does not ask for any stock symbols from a user when executing the program. Instead, it asks the user to store all the stock symbols in which he is interested into a text file, before running the program. The user should type only one stock symbol per line, name the text file "symbols" and store this file in the directory "C:\." An example of this text file is shown in Figure 4.15.

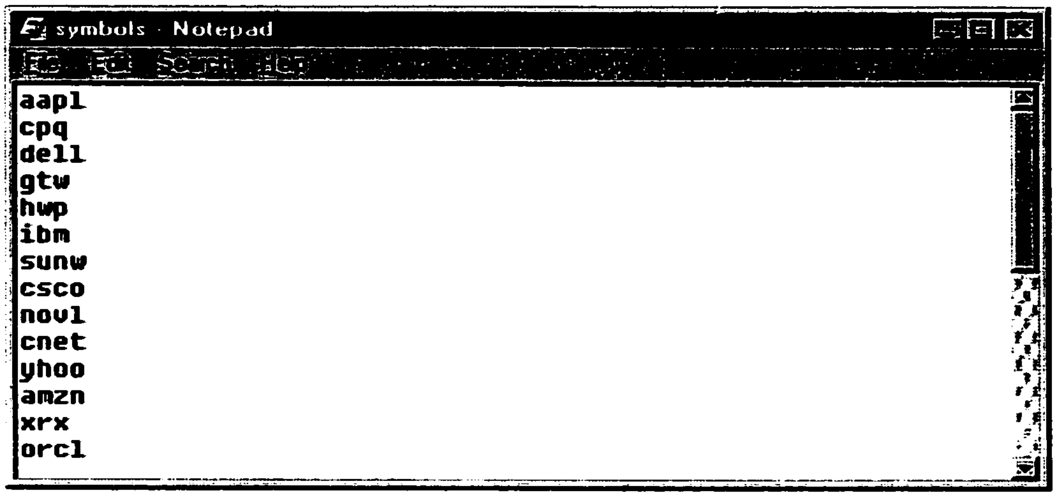


Figure 4.15 An example of "C:\symbols" file .

Moreover, in QBSP, there is no limit on the number of stock symbols users can enter, compared with the maximum of 30 in BSP. QBSP achieves this by resizing the array storing symbols after every 30 elements. These are the only differences between BSP and QBSP. The rest of the program's structure remains the same.

Since the data acquisition function is independent from the data analysis function, it can be separated from the system and used alone. In BSP, the data-acquisition function is called DataAcquisition, whereas in QBSP, it is called QDataAcquisition. During the period of this study, no historical option quotes have ever been found available on the Web. These two programs provide users with a simple mechanism to collect daily option quotes so that they can establish their own data source and conduct further research.

4.6 Error Handling

BSP and QBSP have the capability to handle some common errors. In either of the following situations, a critical error message window will be displayed and the programs will be subsequently terminated.

- A user gives an invalid stock symbol.
- A user gives a stock symbol for which there are no options trading.
- A user tries to run QBSP or QDataAcquisition without the file "C:\symbols."
- A user enters one stock symbol more than once in one execution.
- There is no Internet connection.

Chapter 5

Manual on the Systems

BSP, QBSP, DataAcquisition, and QDataAcquisition are stored in the workbook "StockOption.xls" as macros. They can be distributed with the workbook. This chapter documents the setup of the systems and running steps of the macros. Also an example of BSP execution is discussed in detail.

5.1 Installation

The macros can only run in an environment in which MS Excel 2000 and a web browser have been previously installed. To install the Stock Option Analysis systems, copy the file "StockOption.xls" into the startup folder of Excel.

Excel has two startup folders. One is the default startup folder, whose location depends on the operating system used and the way the system has been configured. It could be in either of these two directories:

- Root\OS\Profiles\User_name\Application Data\Microsoft\Excel\XLStart
- Root\Program Files\Microsoft Office\Office\XLStart

The other startup folder is the alternative startup folder. To define the alternative startup folder, click on **Tools** on Excel menu bar, select **Options** and click on the **General** Tab. In the Alternate startup file location box, enter the path for the alternative startup folder. By putting a workbook in the Excel startup folders, the macros in the workbook are available to all the opened Excel files.

5.2 An Example of BSP Execution

In this section, the details of an example running of BSP on April 6, 2001 are documented.

1. Open Excel.
2. Press ALT+F8, or, point to **Macro** on the **Tools** menu, and then click **Macros**, to bring up the macro list. The list may look different from the one shown in Figure 5.1 as there may be other files containing macros opened on the user's computer. But, the macros in Figure 5.1 should be found in the list.

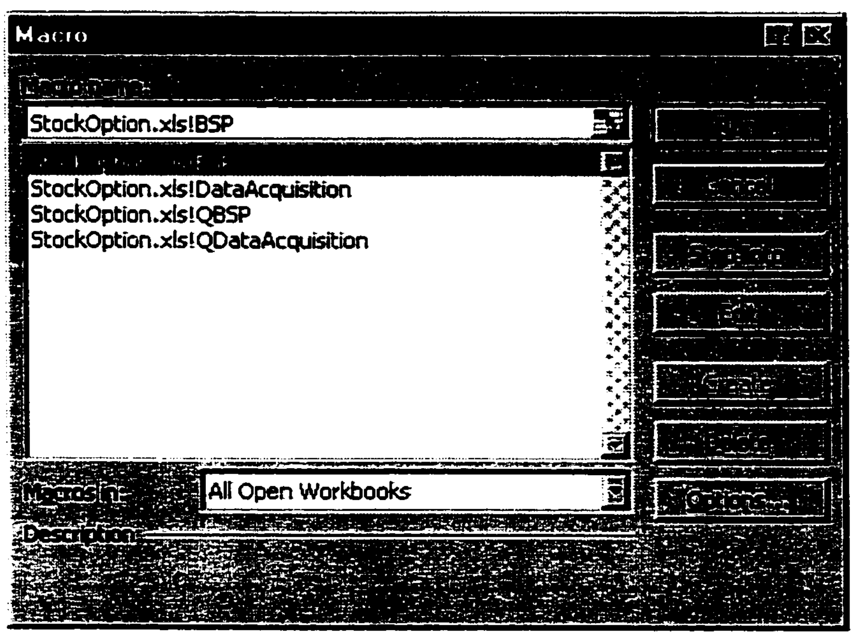


Figure 5.1 Macro list window in Excel

3. Click on StockOption.xls!BSP to highlight it.

4. Press **Run** button to run BSP.

5. A dialog window will appear and ask the user to input the symbols of stocks on which the options trade. Type in the symbols in any field and press **Get Quote** button. The Figure 5.2 shows an example of the input.

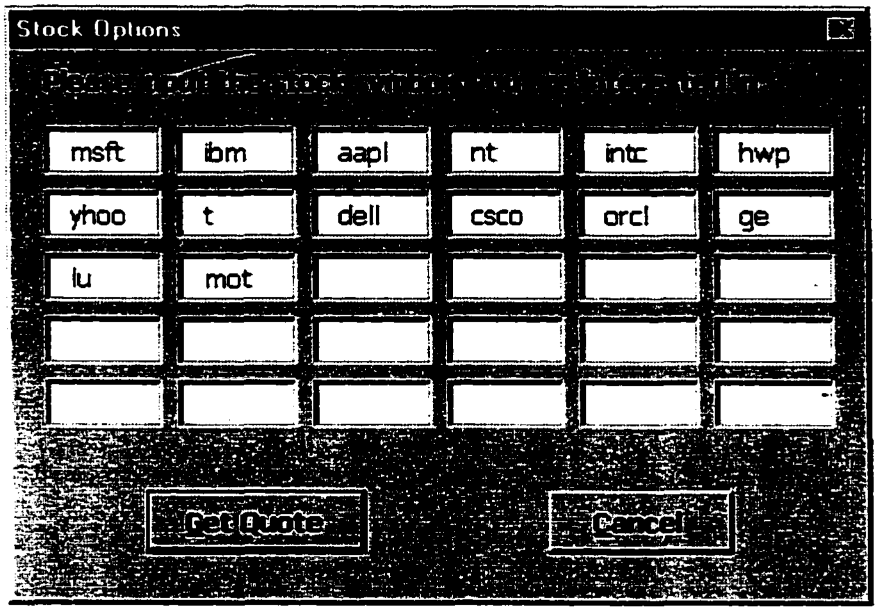


Figure 5.2 An example of the symbols input

6. Then, the program goes to the Web and retrieves the quotes. This process takes from a couple of seconds to more than ten minutes to execute, depending on the Internet connection used and the number of stock symbols entered.

7. Next, another window appears and asks the user to input the interest rate. An interest rate of 6% is entered in the dialog box shown in Figure 5.3 as an example. Click on **OK** button to continue the program.

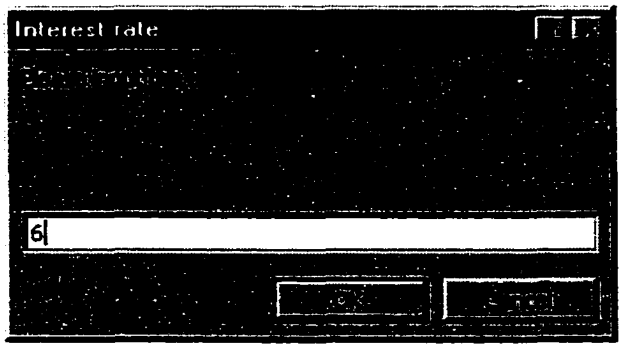


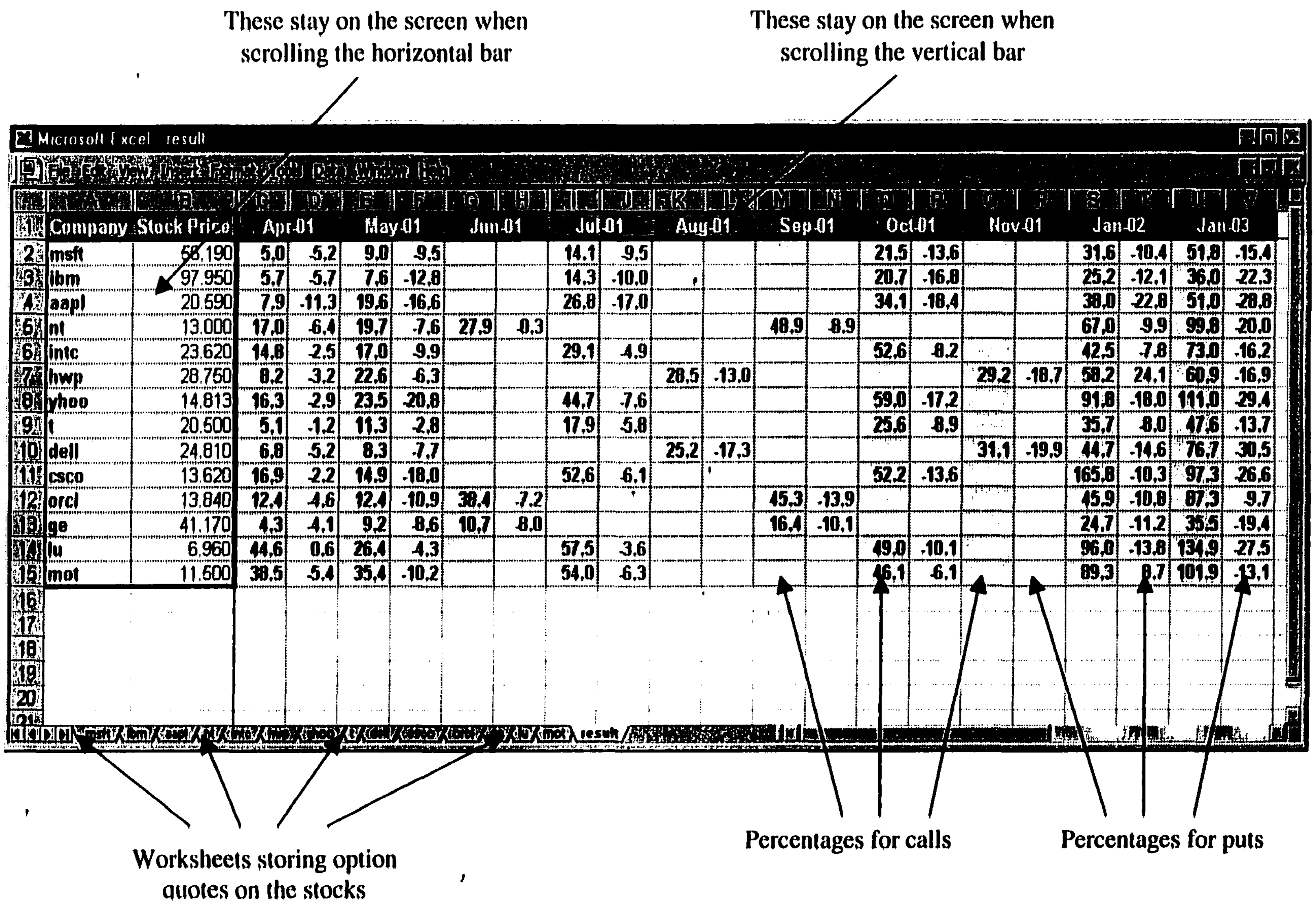
Figure 5.3 An example of the interest rate input

8. Finally, the program stops running and returns the result window, as shown in Figure 5.4.

The "result" sheet presents all the information obtained by BSP system. The numbers are the percentages by which the breakeven stock prices are different from the current stock prices. Numbers in the shaded fields are for calls, which indicate the scope of the upward movement of stock prices. The other numbers are for puts, which show the scope of the downward movement of stock prices. The empty cells exist because no options with that exercise month are trading at the moment. Users can scroll the horizontal bar and the vertical bar to see more information if needed. When scrolling, the company name, stock price, and months will remain fixed on the screen.

As can be seen in Figure 5.4, in the Apr-01 (April 2001) column, a positive figure appears in the downward change for Lucent (lu), which suggests that even put holders expect the stock price to go up. In the May-01 (May 2001) column, Apple Computer (aapl) has relatively larger price changes in both upward and downward movement than Dell Computer (dell). This suggests that Apple's stock is more volatile than Dell's. Lucent Technology (lu) has a larger upward change than downward change. This indicates there is more pressure to push the stock price up than down. In contrast, Cisco (csco) has a larger downward change than upward change, which suggests that is more pressure to drag the stock price down than push it up.

A user can also click on other worksheets to see the options' quotes. For example, click on the worksheet with the name "msft," and then the option quotes on Microsoft (MSFT) will show as in Figure 5.5.



Company	Stock Price	Apr-01		May-01		Jun-01		Jul-01		Aug-01		Sep-01		Oct-01		Nov-01		Jan-02		Jan-03	
msft	58.190	5.0	-5.2	9.0	-9.5			14.1	-9.5					21.5	-13.6			31.6	-10.4	51.8	-15.4
ibm	97.950	5.7	-5.7	7.6	-12.8			14.3	-10.0					20.7	-16.8			25.2	-12.1	36.0	-22.3
aapl	20.590	7.9	-11.3	19.6	-16.6			26.8	-17.0					34.1	-18.4			38.0	-22.8	51.0	-28.8
nt	13.000	17.0	-6.4	19.7	-7.6	27.9	-0.3					48.9	-8.9					67.0	-9.9	99.8	-20.0
intc	23.620	14.8	-2.5	17.0	-9.9			29.1	-4.9					52.6	-8.2			42.5	-7.8	73.0	-16.2
hwp	28.750	8.2	-3.2	22.6	-6.3					28.5	-13.0					29.2	-18.7	58.2	24.1	60.9	-16.9
yhoo	14.813	16.3	-2.9	23.5	-20.8			44.7	-7.6					59.0	-17.2			91.8	-18.0	111.0	-29.4
t	20.500	5.1	-1.2	11.3	-2.8			17.9	-5.8					25.6	-8.9			35.7	-8.0	47.6	-13.7
dell	24.810	6.8	-5.2	8.3	-7.7					25.2	-17.3					31.1	-19.9	44.7	-14.6	76.7	-30.5
csco	13.620	16.9	-2.2	14.9	-18.0			52.6	-6.1					52.2	-13.6			165.8	-10.3	97.3	-26.6
orcl	13.840	12.4	-4.6	12.4	-10.9	38.4	-7.2					45.3	-13.9					45.9	-10.8	87.3	-9.7
ge	41.170	4.3	-4.1	9.2	-8.6	10.7	-8.0					16.4	-10.1					24.7	-11.2	35.5	-19.4
lu	6.960	44.6	0.6	26.4	-4.3			57.5	-3.6					49.0	-10.1			96.0	-13.8	134.9	-27.5
mot	11.500	38.5	-5.4	35.4	-10.2			54.0	-6.3					46.1	-6.1			89.3	8.7	101.9	-13.1

Figure 5.4 Result sheet from the example execution

56.190

APR,2001-DEC,2004 CALLS

Symbol	Description	Month	Strike Price	C/P	Last Trade	Net Chg	Vol	Open Interest	Daily High	Daily Low	Date
MQFDF	MQF	Apr1	30.0	C	26.500	-0.250	10	1880	26.500	25.750	04/06
MQFDG	MQF	Apr1	35.0	C	21.750	0.000	0	571	0.000	0.000	04/05
MQFDH	MQF	Apr1	40.0	C	16.625	-0.125	10	16579	16.625	16.625	04/06
MQFDI	MQF	Apr1	45.0	C	11.000	-1.125	14	2738	12.000	11.000	04/06
MSQDJ	MSFT	Apr1	50.0	C	7.125	-0.625	230	23045	8.125	6.625	04/06
MSQDK	MSFT	Apr1	55.0	C	3.375	-0.625	6413	45368	4.125	3.000	04/06
MSQDL	MSFT	Apr1	60.0	C	1.188	-0.375	2790	79818	1.500	0.375	04/06
MSQDM	MSFT	Apr1	65.0	C	0.375	-0.063	1224	45127	0.500	0.250	04/06
MSQDN	MSFT	Apr1	70.0	C	0.125	0.000	678	32410	0.125	0.063	04/06
MSQDO	MSFT	Apr1	75.0	C	0.125	0.063	13	30016	0.125	0.063	04/06
MSQDP	MSFT	Apr1	80.0	C	0.063	0.000	14	17418	0.063	0.063	04/06
MSQDQ	MSFT	Apr1	85.0	C	0.063	0.000	7	7782	0.063	0.063	04/06
MSQDR	MSFT	Apr1	90.0	C	0.063	0.000	0	22437	0.000	0.000	03/26
MSQDS	MSFT	Apr1	95.0	C	0.063	0.000	0	10856	0.000	0.000	03/28
MSQDT	MSFT	Apr1	100.0	C	0.063	0.000	0	11719	0.000	0.000	03/26
MSQDA	MSFT	Apr1	105.0	C	0.063	0.000	0	1704	0.000	0.000	04/04
MSQDB	MSFT	Apr1	110.0	C	0.063	0.000	0	1023	0.000	0.000	03/20
MSQDC	MSFT	Apr1	115.0	C	0.063	0.000	0	419	0.000	0.000	04/03
MSQDD	MSFT	Apr1	120.0	C	0.063	0.000	0	481	0.000	0.000	02/13
MSQDE	MSFT	Apr1	125.0	C	0.063	0.000	0	1280	0.000	0.000	02/13
MSQDF	MSFT	Apr1	130.0	C	0.063	0.000	0	604	0.000	0.000	04/04
MSQDG	MSFT	Apr1	135.0	C	0.063	0.000	0	4420	0.000	0.000	02/07
MQVDH	MQV	Apr1	140.0	C	0.063	0.000	0	330	0.000	0.000	04/04
MQVDI	MQV	Apr1	145.0	C	0.000	0.000	0	0	0.000	0.000	00/00
MQVDJ	MQV	Apr1	150.0	C	0.000	0.000	0	0	0.000	0.000	00/00
MQVDL	MQV	Apr1	160.0	C	0.000	0.000	0	0	0.000	0.000	00/00
MQVDN	MQV	Apr1	170.0	C	0.000	0.000	0	0	0.000	0.000	00/00
MQVDP	MQV	Apr1	180.0	C	0.000	0.000	0	0	0.000	0.000	00/00
MQFED	MQF	May1	20.0	C	36.750	0.000	0	45	0.000	0.000	04/05
MQFEX	MQF	May1	22.5	C	34.250	0.000	0	0	0.000	0.000	04/05
MQFEE	MQF	May1	25.0	C	31.875	0.000	0	0	0.000	0.000	04/05
MQFEF	MQF	May1	30.0	C	27.500	5.000	10	161	27.500	27.500	04/05
MQFEG	MQF	May1	35.0	C	22.125	0.000	0	435	0.000	0.000	04/05
MQFEH	MQF	May1	40.0	C	17.000	-0.375	3	204	17.000	17.000	04/06
MQFEI	MQF	May1	45.0	C	12.500	-0.375	14	702	12.500	12.000	04/06

Figure 5.5 Quote sheet for MSFT from the example execution

5.3 Running QDataAcquisition

With the help of the Task Scheduler available in Windows, the macro of QDataAcquisition can run automatically every day. This provides a convenient way to get daily option quotes without the user being directly involved. To accomplish this, a dummy workbook is needed. The following specifies what needs to be done:

1. Create an Excel workbook and give it a name. For example, "Autoquote.xls."
2. Then, with the file Autoquote opened, on the **Tools** menu, point to **Macro**, and click on **Visual Basic Editor** to open VB editor window.
3. Double click on **ThisWorkbook** in the VBAProject of Autoquote and write a book open event, which will be activated when the Autoquote file is opened. The code is similar to the following:

```
Private Sub Workbook_Open()

    Workbooks.Add
    Application.Run "StockOption.xls!QDataAcquisition"

    Dim filename As String
    filename = "Quote" & Month(Date) & "-" & Day(Date) & "-"
    & Year(Date)
    ActiveWorkbook.SaveAs ("\" & filename)
    Application.Quit

End Sub
```

File Name

File Location

If this event runs on Apr. 06, 2001, the stock option quotes on that day are saved as "Quote04-06-2001" and stored in the root directory. However, a user can choose a different filename scheme and the location to save the file by changing the corresponding part in the code.

4. If the user has a digital signature, sign the project of Autoquote by clicking **Digital Signatures** on the **Tools** menu, and then add him to the list of trusted sources. If he does not have a digital signature, set the security level to low in the workbook window by pointing to **Macro** in the **Tools** menu, then clicking on **Security,** and on the **Security Level** tab, choose the option of low level. These settings can prevent the user from being asked to enable or disable macros.
5. Save the Autoquote file.
6. Open the Task Scheduler in Windows. Schedule "Autoquote.xls" to run on weekdays at the time preferred.

As a result, the QDataAcquisition runs automatically at the scheduled time. After the system finishes running, the resulting Excel file is saved with a name following the name convention specified in the code and Excel is closed. To prevent the book open event from automatically running, hold down SHIFT key while opening the file "Autoquote.xls."

Chapter 6

Conclusions and Future Work

This chapter commences with a summary of the work that has been accomplished and is followed by suggestions for further studies based on this thesis.

6.1 Summary

This thesis introduces the approach to compute the breakeven stock prices for call or put options in order to assess the investors' opinions about the stock markets and describes the design and implementation of the system BSP in detail. BSP retrieves specified stock option quotes from the Web and analyzes the quotes obtained to calculate the breakeven stock prices for call or put options.

A stock option is the right to buy or sell the underlying stock at a predetermined strike price at a future time. Because the special relationship between stock options and their underlying stocks, from the strike prices with which option holders are trading, a clue of their opinions about the stocks in the future can be obtained. A breakeven stock price is the expected expiration-date stock price for call or put options in order for the call or put option holders as a group to break even. By computing the breakeven stock prices, option holders' opinions are quantitatively measured.

The breakeven stock prices, presenting investors' opinions about stock markets, are useful for users to gain an insight of the stock markets so that the users can have a better understanding of stock movements. Especially when comparing the breakeven stock prices between different stocks or between stocks in different industry sectors, interesting information can be obtained.

BSP is developed in VBA for Excel. It utilizes the Excel workbook model and many Excel built-in functions. BSP can only be run using Excel 2000 or a later version because of the advanced built-in functions used in the program. BSP has an alternative, QBSP, which provides a different user interface. Both BSP and QBSP have sub functions, DataAcquisition and QDataAcquisition, which can be used alone to retrieve option quotes from the Web. Moreover, by using the Task Scheduler to schedule QDataAcquisition, the specified stock option quotes can be automatically retrieved from the Web and saved in an Excel file on a daily basis.

6.2 Future Work

BSP or QBSP only computes breakeven stock prices from one day's option trading. If the breakeven stock prices can be recorded everyday for a period of time, changes on these prices could be observed. Thus, it would be possible to conduct another analysis to explore the relationship between the breakeven stock prices and the current stock markets. This study would help us understand the reason behind the changes in breakeven stock prices, and therefore, will help us determine the factors that influence option holders to change their expectations. As a result, it would be clearer whether a breakeven stock price is a contrarian indicator.

In the implementation of BSP, it has been assumed that the interest rate from the risk-free market is the same for all maturities. This is reasonable for regular options because they take 8 months or less to mature. However, for LEAPS, they can take more than two years. Consequently, a better risk-free interest rate is expected. It would be nice to confine the computational model and use different rates for different lengths of time.

Bibliography

[1] Achelis, Steven B. "Technical Analysis from A to Z." <http://www.equis.com/free/taaz/inttechnicalanalysis.html>.

[2] Canina, Linda, and Stephen Figlewski. "The Information Content of Implied Volatility." *Review of Financial Studies*. Volume 6, Issue 3, pages 659-681.

[3] Chi, E.H.-H., P. Barry, J. Riedl, and J. Konstan. "A Spreadsheet Approach to Information Visualization Information." *Proceedings of the 1997 IEEE Symposium on Visualization*, Page(s): 17 -24, 116. 1997.

[4] Chicago Board of Option Exchange. <http://www.cboe.com>.

[5] Cox, John C., and Mark Rubinstein. "A Survey of Alternative Option-Pricing Models." *Option Pricing: Theory and Applications*. Pages 3-33. New York University, January 18-19, 1982.

[6] Dumas, B., J. Fleming, and R.E. Whaley. "Implied Volatility Functions: Empirical Tests." *Proceedings of the IEEE/IAFE 1996 Conference on Computational Intelligence for Financial Engineering*. Pages 199-233. 1996.

[7] Figlewski Stephen. "Forecasting Volatility." *Financial Markets, Institutions & Instruments*. Volume 6, Number 1. New York University Salomon Center, 1997.

[8] Hull, John C. Options, Futures, and Other Derivatives. Third edition. Prentice Hall, 1998.

[9] Investment Learning Center. <http://www.cboe.com>.

[10] Investment Research Institute. <http://www.options-iri.com>.

[11] Malz, Allan M.. "Using Option Prices to Estimate Realignment Probabilities in the European Monetary System." *Reserve Bank of New York Staff Reports*. No. 5. Sep. 1995.

[12] Schaeffer Bernie. The Option Advisor. John Wiley & Sons, Inc., 1997.

[13] Schaeffer Bernie. "Getting Started With Stock Options." <http://www.schaeffersresearch.com/option/getting_started.asp>.

[14] Schwage, Jack D. Getting Started in Technical Analysis. John Wiley & Sons. Inc., 1999.

[15] Wuthrich, B., V. Cho, S. Leung, D. Permunetilleke, K. Sankaran, and J. Zhang. "Daily Stock Market Forecast from Textual Web Data." *1998 IEEE International Conference on Systems, Man, and Cybernetics*. Volume 3, pages 2720 –2725. 1998.

[16] "Understanding Volume & Open Interest." <http://futures.tradingcharts.com/learning/volume_open_interest.html>.