An Evolutionary Computing Approach to
Financial Portfolio Management Based on
Growth Stocks & Sector/Industry Distribution

André Nicolau Fernandes
Instituto Superior Técnico
Universidade de Lisboa,
Lisboa, Portugal
Email: anfernandes10@gmail.com

Abstract—In this paper we propose an artificial intelligence based system to compose and manage financial portfolios. Financial knowledge and evolutionary algorithms are incorporated in the system to achieve a better performance when choosing the stocks to compose the portfolio. The main goal is to design a strategy which captures the most promising growing companies on the market. To achieve this goal, the system designed incorporate a fundamental approach and a market division by sectors/industries, using financial ratios and sector specific indicators. Several investing strategies were tested, and further validated using as measures the return, variance and extra risk. The simulations were conducted using date from the S&P 500 index for the periods January 2011 until December 2014. Real constraints are used to define an environment similar to the ones faced by real portfolio managers. The results demonstrated that combining evolutionary computation with financial ratios is a good solution to select the most promising companies. The division of the market by sectors also proved to be profitable. Simulations obtained returns above the market with slightly lower variances. The best results suggested that the most promising stocks are the ones presenting higher returns on equity (ROE), higher rate of revenue (RR) and price to earnings growth (PEG), and additionally, low debt ratios (DR).

1. Introduction

Stock markets have been around for centuries. The London Stock Exchange (LSE) and the New York Stock Exchange (NYSE) are examples of large and old stock exchange, been funded in 1773 and 1792 respectively. Over the last 50-60 years, the world suffered the Third Industrial Revolution, the so called Digital Revolution, marked by an exponential technology evolution in every possible fields. Of course, in this Information and Digital Age, the financial and economic world was not forgotten or neglected. On the contrary, the advances in the computer science were well applied in this subject. With the introduction of the internet and the possibility of worldwide knowledge in real time (globalization), the exchange markets had to change their ways of operating, introducing the remote stock trading. Also, nowadays, stock information is an easy thing to store and analyse. As result of the easy access to trading prices, historical prices and volume of transactions, new investing strategies and philosophies appeared.

Until then, investors based their stock choices on the companies’ fundamentals, like their profit, sales and net income [1]. Though in the current days it still represents the most common strategy, many investors and analysts give preference to the analysis of prices and trends, technique known as Technical Analysis. Technical analysts believe that by looking into historical data related with stocks and the overall market, they will be able to forecast the future at a certain degree, without studying the micro and macro economy [2]. Both philosophies have supporters and criticisms. Motivated by all this, the stock exchange become a popular research field among computer science people [3]. With all the information available, it starts to be difficult to manage and chose portfolios without the computers help. It may be impossible in practice to manage financial portfolios with a large amount of stocks without a computer. Computer scientists thought “what if we could create an automatic system to choose the best stocks and manage financial portfolios?”.

Mainly on the Artificial Intelligence field, specialist did a exhaustive research in order to combine both areas. With the extreme evolution in computation, solving large problems with dozens of variables in real time is a reality, opening the doors to tools such as evolutionary algorithms (EA), fuzzy logic [4], artificial neural networks (ANN) [5] or other learning guided approaches [6]. These are techniques based on the natural biological evolution. As Darwin stated, the fittest survives and seed the next generation. After several generations of natural selection and reproduction, the population converge on an optimal set of individuals.

In this project we created and tested a system that uses an EA, applied to the stock exchange. More concisely, to solve the problem of selecting the best stocks among the thousands available and manage the resulting portfolio, the system applies an evolutionary algorithm, mixing the economic and the artificial intelligence worlds. The algorithm
uses several financial ratios, calculated with the company’s fundamentals (Fundamental Analysis). The objective is to select companies exhibiting an enthusiastic growth potential, supported this selections on the industry and sector [7] in which the company is inserted, instead of choosing from the overall market. The main contributions made in this paper are: (1) Combine fundamental analysis of both companies and industries/sectors with Evolutionary Algorithms in order to develop an adaptive approach for investing; (2) investing decisions based on the industry/sector, investing only in the best companies from the best sectors instead of looking to the overall market; (3) explore indicators based on sector analysis, calculating fundamental ratios based on the companies inserted in each sector/industry; (4) explore new financial ratios based on fundamental analysis to determine if a company is overvalued, supported on current fundamental values and future projections of profits.

This paper is organized as follows: the Section 2 presents the state-of-the-art for the portfolio management and optimization using evolutionary techniques; Section 3 describes the system proposed and the strategies used; Section 4 presents and discusses the case studies and results; Finally, the conclusions and future work are displayed on Section 5.

2. Related Work

In the stock market world, problems are not linear, having an infinite set of possible solutions, each of them with pros and cons. The artificial intelligence gained relevance in the area of forecasting financial markets and optimization of funds in a portfolio, producing systems with the ability to combine thousands of variables in few seconds and produce optimal solutions. Some of the many solutions developed over the last decades are presented here, with special relevance to the ones using evolutionary algorithms. Nonetheless, some other solutions using different artificial intelligence techniques are given a brief overview, such as neural networks (ANNs), fuzzy logic systems and mixes of both with genetic algorithms.

2.1. State of the art of Genetic Algorithms

Yan & Clack in 2007 [8], based on the assumption that historical facts that caused crashes and market environments with great volatility repeat over time, proposed a genetic solution to achieve robustness when faced to an out-of-sample volatile environment. In their work, performance is tested across several environments, such as bearish and bullish market, and considering three possible evolution scenarios for the population, (1) Standard GP (SGP), (2) Multiple-Scenario evaluation in the last generation (MELG) and (3) Multiple scenario evolution (MEVO). The fitness function chosen by the authors is a unique-objective approach, the Sharpe Ratio, which is calculated with the ROI and the Risk Free Rate. The results showed that all three GP systems, compared to the portfolio and overall market index, presented better results when it comes to volatility. When it comes to robustness, MELG performed the best, also presenting greater returns in comparison to the portfolio index.

Kaucic in 2012 [9], developed a solution based on the technical analysis, this is, an evolutionary system supported on several technical indicators. The author deploys a solution composed of three different modules. The investment module, denotes the long-plus-short strategy for the portfolio. The technical module, where several technical indicators are applied to detect promising stock, such as rate of change (ROC), relative strength index (RSI), simple moving average (SMA), hull moving average (HMA), volume moving average (VMA), moving average convergence divergence (MACD) and on balance volume (OBV). The last module is the evolutionary learning module, where a genetic algorithm is used to select the best combination of technical signals, using the reward-to-risk ratio (ratio between the reward and risk of the portfolio) as performance measure. Using three distinct ways of calculating the risk of the portfolio, three evolved portfolios where created, Information Ratio (IR), Sortino Ratio (SR) and Omega ratio.

Soleimani, Golmakani, & Salimi in 2009 [12], proposed a model called Buy and Sell (B&S) using GP to generate trading rules: buy or sell. Using historical data for training, the authors managed to extract these rules from tree data structures. These trees are constructed carefully in order to let the assumptions of GP stainless. The model outperform the hold and buy strategy, although surpassed in extremely bullish markets.

Huang, Chang, Kuo, Lin, Hsieh, & Chang in 2012 [11], studied the selection of assets based on fundamentals of initial public offerings (IPO). Initially, the model ranks the stocks using fundamental indicators which calculate the profitability of the company, the liquidity, leverage, efficiency, etc. The genetic search model implemented seeks the simultaneous optimization of the input features, sorting indicator and weights assigned to each fundamental indicator. In this work, the average return of all portfolio is used to evaluate the performance. The results across several cross-validation scenarios proved that the approach provides the investor above average returns.

2.2. State of the art of portfolio optimization

Anagnostopoulos & Mamanis in 2011 [13] conducted an evaluation of five different MOEA considering the cardinality constrained portfolio problem. They named their
model as mean-variance cardinality portfolio optimization (MVCCPO), and justified the approach with the objective of avoiding extremal situations when choosing the assets, as an example, avoid excessively investing in a single asset. In their study the focus was to verify if any advantages can be extracted from the bi-objective cardinality and test it against the existing solutions of MOEAs on the MVCCPO model, being the first to prove their effectiveness. To achieve that goal, the authors tested five MOEAs: SPEA2, NSGA-II, PESA, NPGA2 and e-MOEA. Also, for comparison purpose, a single objective evolutionary algorithm (SOEA) was included in the study. As performance measures, the chosen approach calculates a linear normalization of the preferences generated by the algorithms. The results showed that the SPEA2 performed best in 75% of the tests made.

Sadjadi, Gharakhani, & Safari in 2011 [14], looked into optimization for cardinality constrained portfolios. The authors extended the Markowitz’s model to support some obligatory constraints: limit the number of holding assets, consider the return for each asset as uncertain, limit with upper and under bonds the investments in each asset, and no short selling allowed. In the GA, the selection mechanism used was the roulette wheel and the uniform selection, ensuring the diversity and improving the quality of the final solution. As performance measure, they use relative percentage deviation (RPD), which calculates the proximity between the optimal solution and the one generated by the GA. Results showed that there is not significant difference between nominal and robust methods regarding returns.

Silva, Neves, & Horta in 2014 [15], proposed a MOEA solution using both fundamental and technical indicators. Using a sever of fundamental ratios as debt ratio (DR), return on equity (ROE), profit margin (PM), price to earnings (PER), revenue growth (RG), and others, the authors constructed a portfolio with the most promising assets. Further, based on technical indicators, decisions were made concerning the market choose, the timing to entry and exit, stop loss triggers, and some portfolio typical constraints. Using a MOEA, the results obtained revealed that is a better investment technique than just following the index. The authors also concluded that the more valuable genes were the ROE, the rate growth of net income and the profit margin.

Mousavi, Esfahanipour, & Zarandi in 2014 [16], introduced an approach using multi-tree genetic programming to construct and manage a portfolio, this is, which assets to buy, when and how much. The rules are trinary, this is, the signal encoded on the rules can be buy, sell or “no trade”. For comparison purposes, the authors use three other approaches: a GA solution, a GP solution and the Buy and Hold strategy. The metrics are the Rate of Return (ROR) and the conditional Sharpe Ratio (CSharp). The advantages of the trinary rules system are reflected in good rates of return and high risk adjusted returns.

Table 1 presents a summary of the related work discussed above, where the principal aspects of the solutions are presented.

3. Solution’s Architecture

The proposed system attends two major problems. (1) The selection of promising growth stocks based on sector distribution; (2) make the decision whether to buy or sell over time. The approach to solve the given problems was found on the branch of artificial intelligence, the so called Evolutionary Algorithms (EA). Two different operations are contemplated. The first is charged with the filtration of stocks, electing the companies that exhibit a greater growth potential in each sector on a daily basis. The second decides when to buy/sell and the percentage of each asset on the portfolio. Both operations are based on fundamental analysis and optimized using an EA. That being said, this paper proposes four new methods focused on maximizing returns: (1) Combine fundamental analysis of both companies and industries with Evolutionary Algorithms; (2) pick the best companies from each sector, instead of looking only to the overall market; (3) explore several indicators based on sector analysis; (4) explore new indicators based on fundamental analysis.

3.1. Algorithm Architecture

The system architecture is constituted by four main modules: (1) the Picking Module (the GA), (2) the Portfolio Simulator, the (3) Data Processing Module and (4) the Fundamental Ratios Module. The interaction and data flow between the modules is presented in figure 1. The user provides the system with his preferences and constraints, the budget and investing strategies for example. Next, the system starts operating, calculating a set of fundamental ratios with the data gathered from companies in S&P 500 to understand their profitability, liquidity, debt, etc. The sectors performance is also calculated and stored in this phase. Having the fundamental information, the so called Picking GA starts its execution, generating several chromosomes (individuals) with different weights for each fundamental ratio obtained in the previous step. Each individual is submitted to a simulation using the portfolio simulator for a quality check. During the training period there is an evolution process. The survivors, this is, the stocks presenting higher returns, are selected to constitute the new generations. Once the training period is over, the individuals are sorted by performance. The best individuals are selected for a real life simulation (testing period). The system runs again the portfolio simulator with a out-of-sample period to measure the performance of the individuals in an unseen environment. Finally, the results are delivered to the user.

3.2. Optimization Process

Given a certain population in the GA, it is necessary to specify how the algorithm will select the individuals (the individuals’ representation is approached on section 3.6) that will generate the next generation. The chosen process is known as the “Natural Selection” methodology, which mainly consists in sorting the individuals according to their
TABLE 1: Overview over different solutions using evolutionary techniques

<table>
<thead>
<tr>
<th>Reference</th>
<th>Period</th>
<th>Markets</th>
<th>Algorithms</th>
<th>Fitness Functions</th>
<th>Portfolio Analysis</th>
<th>Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kaucic (2012)</td>
<td>25 Jan 2006 to 19 Jul 2011</td>
<td>DJI</td>
<td>GA</td>
<td>reward-to-risk</td>
<td>Technical Analysis</td>
<td>The three evolved portfolios largely surpassed the DJI index</td>
</tr>
<tr>
<td>Esfahanipour, Mousavi (2011)</td>
<td>22 Aug 2004 to 21 Aug 2008</td>
<td>TSE</td>
<td>GP</td>
<td>Risk Adjusted Return using Sharpe Ratio</td>
<td>Technical Analysis</td>
<td>Outperformed the Buy and Hold model. 10.85%</td>
</tr>
<tr>
<td>Huang, Chang, Kuo, Lin, Hsieh, Chang (2012)</td>
<td>2005 to 2011</td>
<td>SFI</td>
<td>GA applied to search model</td>
<td>Mean Fundamental Analysis</td>
<td>The model showed above average for first day returns</td>
<td></td>
</tr>
<tr>
<td>Soleimani, Golmakani, Salimi (2009)</td>
<td>NA</td>
<td>NA</td>
<td>GA, LINGO</td>
<td>RAR</td>
<td>Markowitz’s Model</td>
<td>11% for small scale problems, 7% for large scale problems, with 3.5% of risk</td>
</tr>
<tr>
<td>Anagnostopoulos, Mamanis (2011)</td>
<td>NA</td>
<td>Hang Seng, DAX100, FTSE100, S&amp;P500, NASDAQ</td>
<td>MOEAs (SPEA2, NSGA-II, PESA, NPGA2, e-MOEA)</td>
<td>Mean, Variance Markowitz’s Model</td>
<td>SPEA2 performed better among the five MOEAs and the SOEA tested</td>
<td></td>
</tr>
<tr>
<td>Sadjadi, Gharakhani, Safari (2011)</td>
<td>NA</td>
<td>Hang Seng, DAX100, FTSE100, S&amp;P100, Nikkei225</td>
<td>GA</td>
<td>Mean, Variance, RPD Markowitz’s Model</td>
<td>No significantly changes in returns by portfolios with different constraints when these hold with probability of 99%</td>
<td></td>
</tr>
</tbody>
</table>

fitness value and subsequently select some of them for reproduction. The fitness function in this work is the Return On Income (ROI), calculated with the output of the Portfolio Simulator. The ROI function is used to evaluate the gains obtained when investing during a specific period of time. Of course, the higher the returns are, the fitter the individual is, so the goal of the algorithm is to maximize the ROI. For the process of evolution, a key factor is the genetic variation. After the evaluation of the population, these operators are applied to the individuals. The two genetic operators used in this work were the crossover and the mutation.

3.3. Simulation Version

When should a company’s stock enter and leave the portfolio? The decision is made supported with some verifications to the system’s current state. If the conditions are the desired, then buy. If not, sell or do nothing. In this work, three different versions were implemented to deal with this problem. The difference between versions lies in the way that companies are packed into the “sell list” and “buy list”. The three version implemented are: Benchmark, Minimum Exit and Mandatory Timing. The Benchmark is the main version, as it is common to the other two version, this is, the verifications performed in this version are also performed in Minimum Exit and Mandatory Timing versions. The first verification made is if the company’s rate is higher than the buy signal. The company’s rate is calculated daily using the fundamental ratios’ values and the respective weights. Next, if the company is not in the portfolio already and the portfolio is not full, the company is added to the “buy list”. In the case where the portfolio is already complete, if the leap (difference between the current rank position and the last position of the company) is smaller than the minimum jump needed to entry the portfolio, the actions terminate. The minimum jump is one of the portfolio’s constraints, which represents the minimum leap necessary for the company to be considered for buying. The minimum jump constraint and other constraints/parameters are explained in detail on section 3.7. If the leap is bigger than the minimum jump, one last verification is required. If the rate of the worst company in the portfolio is smaller than the rate of the proposed company, then, the worst company is added to the “sell list”.
and the new company is added to the “buy list”. In Minimum Exit version, to reduce the number of transactions made, one more verification was added to the previous one. A company in the portfolio may only be sold if it has a worst rank than the proposed company and if it has fallen considerably from the previous ranking to the current ranking. The leap that the company made is calculated and compared to the minimum jump needed to exit the portfolio. If the leap is higher, the company is added to the “sell list” and the proposed company added to the “buy list.”

The Mandatory Timing versions introduces a new investment flow. Instead of free timing to buy and sell, in this version a new time constraint is applied. Stocks can only be sold if they were kept in the portfolio for \( X \) days. Even if the company performed the worst of all companies, it cannot be sold until the minimum date is not due. The number of days is given by the gene \( \text{time to sell} \).

3.4. Simulation Strategies

How much money should the simulator invest in each company? Three strategies were implemented and tested in this work. With Strategy A, the capital available is divided in equal portions by the companies in the “buy list”.

\[
\text{Investment}_c = \frac{\text{AvailableMoney}}{\text{BuyListSize}} \quad (1)
\]

Strategy B defines that if one sector is making more money than the others, more money should be invested in that sector.

\[
\text{Investment}_c = \frac{\text{AvailableMoney} \times \text{Performance}_s}{\text{BuyListSize}_s} \quad (2)
\]

\[
0.0 \leq \text{Performance}_s \leq 1.0 \quad (3)
\]

Where \( \text{Performance}_s \) is the weight represented by the sector \( s \) considering the entire index, and \( \text{BuyListSize}_s \) is the number of companies of the sector \( s \) in the “buy list”.

For last, Strategy C dictates that best companies shall have more money invested. The “buy list” is mainly sorted by sector’s quality, and by company’s rank on a second phase. With that said, the investing starts in the best company and finishes at the worst. In the end, to avoid having available money when it could be invested, if there is still money available, it is equally distributed for all companies in the portfolio.

\[
\text{Investment}_c = \frac{\text{AvailableMoney} \times \text{Performance}_s}{\text{BuyListSize}_s} \quad (4)
\]

In the end of the process, if money is still available, a reinforcement is performed following the equation:

\[
\text{ReinforceInvestment}_c = \frac{\text{AvailableMoney}}{\text{PortfolioSize}} \quad (5)
\]

3.5. Sector Picking Choices

Being this work a solution which innovates by using the sector division of the companies, this is a major choice. Every trimester, sectors must be ordered by their “value” or potential. To dictate which sectors are the best, many options can be considered. In this solution, four different indicators were experimented: (1) The revenue of each sector on the given trimester, calculated using the revenues of the companies from the sector; (2) the revenue growth in relation to the previous trimester; (3) the sector’s earnings per share, which is calculated using the net income and suggests which sectors are making more real money; (4) the earnings per share growth in relation to the prior trimester. The sector picking produces several rankings used by the simulator to decide which companies to buy. In figure 2 are represented the several rankings produced.

3.6. Chromosome Representation

The individuals, or chromosomes, are a simple array structure of real numbers. For the picking GA, the chromosome chosen has 19 genes. The genes indexed from
0 to 13 represent a weigh given to a specific financial ratio, this is, the importance that the ratio will have on the decision whether to buy or not the stock. All financial ratios used are further explained on section 3.8. Besides the weights, assigned to the financial ratios, six more genes are present. These genes represent constraints necessary to the well function of the software, information regarding: the portfolio maximum size; bounds to define the minimum necessary rank an equity needs to obtain to enter or exit the portfolio; the number of sectors to invest; the time to sell the stocks, this is, the minimum time the stock needs to remain in the portfolio until it can be sold. Furthermore, the last gene represents the metric defining the score that the companies need to rank in order to be an acceptable company. A representative image of the structure for the chromosome is displayed in figure 3.

![Figure 3: Genetic Algorithm Chromosome.](image)

### 3.7. Parameters/Constraints of the Trading System

The last seven genes of each chromosome, or individual, represent parameters of the system. This parameters are used by the Portfolio Simulator module, defining the behaviour of the system. Beside the parameters provided by the genes, some constraints are present to all individuals.

- **Portfolio’s Size** represents the maximum number of stocks that can be present in the portfolio at every moment. Notice that this is the maximum number, it does not mean that the portfolio cannot contain less stocks than the specified value. For example, in the first few days of simulation, usually the portfolio is not completely full.
- **Gain Stop** represents the percentage of profit considered “enough” for the algorithm to finish. Given a certain moment, if the portfolio’s profit is superior to the value on this gene, the algorithm proceed to close all positions and take the profit home. The acceptable profit percentage in this work is 50%.

The **Minimum Jump To Entry/Exit the Portfolio** represent the values to entry and exit the portfolio. Every simulation cycle, companies are selected and kept as possible companies to insert in the portfolio. This selection is based on their rank, calculated using the weights present in the other genes and the corresponding fundamental ratios. Before the algorithm makes the decision whether to buy or not the stock, it also considers the last rank of the company. For the decision to be positive, the following equation must be verified:

\[
Jump \geq MJ_{Entry} 
\]

\[
Jump = LR - AR
\]

Where \(MJ_{Entry}\) is the Minimum Jump To Entry value, varying between 20 and 30; \(LR\) is the last rank position of the company and \(AR\) is the actual rank position of the company. This verification ensures that the algorithm only invests in a company that presents real improvements and high chances of growth, leaving behind companies that are basically as they were in the last simulation cycle (even if they are doing slightly better). Analogous to the previous process, the exiting process also considers the last rank of the company. In this case, the equations to be verified is:

\[
Jump \geq MJ_{Exit} 
\]

\[
Jump = AR - LR
\]

Where \(MJ_{Exit}\) represents the Minimum Jump To Exit value, varying between 20 and 30. Contrary to the previous case, this verification ensures that the algorithm only sells companies that are presenting fairly poorer results when compared to the previous values. An illustration is provided to clarify the Minimum Jump to Entry action on figure 4 assuming a minimum jump value of 20.

The **Number of Sectors to Invest** represent the number of sectors available to invest. In this system, each simulation is limited to a certain number of sectors. This mean that the simulator can only select companies amongst the \(X\) best sectors. For each trimester, a new rank of sectors is calculated, and the companies from the worst sector are discarded from the possibilities list. This decision is based in two reasons: (1) prevent the algorithm from investing in only one sector. If this industry would collapse, the investor would lose all his money. (2) only buys from the best momentary sectors, avoiding at least the worst 3 every simulation.

The **Time to Sell** is the minimum amount of time that the asset must remain in the portfolio. The value’s unity is days. If
the algorithm chooses to buy a given stock, it cannot sell it right away, the following equations must be verified:

\[ DP \geq TTS \]  
\[ DP = BD - PD \]  
\[ 0 \leq BD \leq 365 \]  
\[ 0 \leq PD \leq 365 \]

Where \( TTS \) is the Time To Sell value, varying between 15 and 60 days; \( DP \) is the days passed between the bought and the present date; \( BD \) is the Buy Day and \( PD \) is the Present Day.

**Buy Signal** represents the minimum value that a company must score to be considered for buying. **Other Constraints**: **Long only constraint** indicates that in this work only long positions are allowed. **Transaction costs** are the costs associated to every buy and sell operation. In this work, the commission paid for each transaction is 0.3% of the its value. **No dividends** are considered in this solution.

### 3.8. Fundamental Ratios Modules

This module specifies several ratios that can be used in the GA to evaluate quantitatively a certain company. The objective is to determine the ratios that better suits the model, this is, that more accurately find highly growing companies. For this work, taking into account the final objective of finding growing companies, 12 ratios were selected and 2 new ratios were created. The two original ratios are first described, followed by the other twelve used.

After its calculus, to each ratio, given its value, a score was assign. This is meant to normalize all values and this way simplify the mix of the several ratios, ranking each company with a normalized data set. The score is a integer from 0 to 5.

#### 3.8.1. Revenue Rate (RR).

The Revenue Rate ratio is an original ratio created for this work. The main goal is to detect which companies are highly overvalued.

\[
Revenue \ Rate \ (RR) = \frac{Stock \ Price \times \text{Outstanding Shares}}{Revenue} \quad (14)
\]

The fraction numerator represents the trading value of the company, this is, if all stocks were sold at the momentary price, how much money would be received. The denominator represents the revenue generated by the company. A RR value below 1 means that the company’s market value is lower than the money being generated by the company’s activity, which suggests that the company’s business market value is undervalued and it will probably rise in the near future. On the other hand, if the RR value is too high, 10 for example, indicates that the company is selling 10 times over its worth in sales/revenue. In this situation, is mandatory to double check if the company has indeed revenue power to justify that value or if it is highly overvalued.

Assuming that the higher profit prospective a company can make a year is 20%, which is a hard reality even for the biggest corporations. A RR of 10, implies that the company’s market value is 10 times the revenue:

\[
Market \ Value \ (MV) = Stock \ Price \times \text{Outstanding Shares} \\
RR = \frac{MV}{Revenue} \land RR = 10 \\
\equiv MV = 10 \times Revenue.
\]

Considering the PER formula and the restrictions:

\[
PER = \frac{MV}{Profit} \land MV = 10 \times Revenue \land \\
Profit = 20\% \times Revenue \equiv PER = \frac{10 \times Revenue}{0.2 \times Revenue} = 50.
\]

Conclusion, a RR of 10 imply a PER value of 50, which is a bad indicator and a strong signal that the company is overvalued. To keep up with a RR value of 10, companies’ profits should round the 50%, a very unlikely scenario.

That being said, the objective is to find companies with profit margins rounding the 20% and presenting RR values around 4-5, which indicates that the company is not overvalued and has a very good profit rate.

#### 3.8.2. Price to Earnings Future (PER-F).

This ratio is the second original ratio created for this work. The objective of this ratio is to calculate the PER value 3 years from the day of its calculation.

\[
Price \ to \ Earnings \ Future \ (PER - F) = \frac{Stock \ Price_{Actual}}{EPS_{3 \ years}} \quad (15)
\]

\[
EPS_{3 \ years} = \frac{Net \ Income_{3 \ years}}{Outstanding \ Shares_{Actual}} \quad (16)
\]
\[ \text{PER} - F \equiv \frac{\text{Market Value}_{\text{Actual}} \times \text{Outstanding Shares}_{\text{Actual}}}{\text{Net Income}_{3 \text{ years}}} \]  

(17)

Considering:

\[ \text{Market Value} = \text{Stock Price} \times \text{Outstanding Shares}, \]

the PER-F ratio is given by:

\[ \text{PER} - F = \frac{\text{Market Value}_{\text{Actual}}}{\text{Net Income}_{3 \text{ years}}} \]  

(18)

The goal of this work is to find growing companies, and to achieve this goal, the PER ratio is very important and effective. Although, it may be misleading. By performing a simulation of the future, the investor can have an idea about the growing perspectives of the company. Also, is a good comparison metric between companies.

From the previous equations is possible to derive the basic idea of this ratio: assuming that the net income growth rate will be the same for the next 3 years, and assuming that the company’s market value is kept, what will be the PER value? For example, given two different companies, A and B. The market values are \( MV_A = 40 \) \& \( MV_B = 40 \). The net income is given by \( NI_A = 3 \) \& \( NI_B = 4 \). Calculating the PER with the previous values: \( \text{PER}_A = 13 \) \& \( \text{PER}_B = 10 \). By analysing the current PER values, company A is more attractive.

But what are the perspectives for the next 3 years? Considering a current net income growth rate of \( \text{NIGR}_A = 20\% \) \& \( \text{NIGR}_B = 5\% \), and assuming that this values will be maintained for the next 3 years, the PER-F is: \( \text{PER} - F_A = 7.7 \) \& \( \text{PER} - F_B = 8.6 \). Three years from now, under the same circumstances company B is slightly more attractive than company A.

Return on Equity (ROE) shows how profitable a company is by comparing the net income with the money shareholders have invested on the company, this is, reveals the percentage of profit that the corporation earned.

\[ \text{Return on Equity (ROE)} = \frac{\text{Net Income}}{\text{Average Shareholders’ Equity}} \]  

(19)

Return on Assets (ROA) shows how profitable a company is by comparing the net income with the company’s total assets. Is reveals if the management is employing the assets well in order to make money, this is, the higher the ROA, the better capacity to generate money.

\[ \text{Return on Assets (ROA)} = \frac{\text{Net Income}}{\text{Total Assets}} \]  

(20)

Current Ratio (CR) shows the ability of a company to cover current liabilities. It is the most popular ratio to evaluate the firm’s liquidity. An high value indicates that the assets of the company can easily cover the liabilities, this is, the short-term assets are readily to pay off the short-term liabilities. A positive high CR reduces the risk of the investment, however a lower CR may not necessarily be a bad indicator, it could mean that the company is using their assets to grow the business.

\[ \text{Current Ratio (CR)} = \frac{\text{Current Assets}}{\text{Current Liabilities}} \]  

(21)

Debt Ratio (DR) shows the dependency on leverage. The lower the DR, the less leverage the company is using, and the less risk the company have. A high percentage indicates an high debt-load, which means the company is currently more financed by money owed to others.

\[ \text{Debt Ratio (DR)} = \frac{\text{Total Liabilities}}{\text{Total Assets}} \]  

(22)

Earnings per Share (EPS) can serve as a profitability indicator, since it reports the portion of company’s profit allocated to each outstanding share of common stock, this is, represents an stock’s performance. The number of shares outstanding can change over the year, for example each six months. For that reason, when calculating the EPS, is good practice to use a weighted average number of shares outstanding over the reporting term.

\[ \text{Earnings per Share (EPS)} = \frac{\text{Net Income} - \text{Dividends on preferred stock}}{\text{Average Outstanding Shares}} \]  

(23)

Price to Earnings (PER) shows how many times a stock is trading per each dollar of EPS. Its the most reported and used ratio by investors. Given that it uses the EPS ratio, a low PER is preferred, as it indicates the number of years required to retrieve the investment. Although, a high PER can also be good news. High PER ratios may indicate positive expectations about the payout or substantiial growths on earnings compared to the overall market.

\[ \text{Price to Earnings (PER)} = \frac{\text{Stock Price}}{\text{Earnings per Share (EPS)}} \]  

(24)

Price to Earnings Growth (PEG) is a refinement of the PER. Differently from the PER, PEG takes into account the growth perspectives of the stock’s earnings. Supported by the PEG ratio, investors and analyst are given an idea of the stock’s valuation, this is, if it is over or undervalued, relative to the growth perspective. Being the unity the neutral value, less than one means that the stock’s price is being undervalued, and the oposite for values with higher PEG ratios.

\[ \text{Price to Earnings Growth (PEG)} = \frac{\text{Price to Earnings}}{\text{Earnings per Share (EPS)}} \]  

(25)

Price to Book Value (P/BV) expresses how much shareholders pay for the net assets of a company. It relates the the market value of the company to the value of a company’s assets on the balance sheet statement. This ratio gives to investors an indication about the under or overvaluation of the stock, showing how many times its intrinsic value a company’s stock is trading.

\[ \text{Price to Book Value (P/BV)} = \frac{\text{Stock Price per Share}}{\text{Average Outstanding Shares}} \]  

(26)

Profit Margin (PM) show the company’s profit margin. Net profit is usually the most mentioned ratio to discuss the
company’s profitability. It shows how much money can the company keep from the earnings made in sales.

\[ \text{Net Profit} = \frac{\text{Net Income}}{\text{Revenue} \times \text{NetSales}} \]  

(27)

Net Income Growth Rate (NIGR) represents the gain, positive or negative, from the last year net income. Through the calculation of the NIGR values for periods of 5-10 years, the investor can have a good perception of the future net profits. A NIGR ratio continuously growing at a stable rate for the last 5 years is a strong indicator of the company’s consistency and ability to produce profit, and so it is a very positive buy signal.

\[ \text{Net Income Growth Rate (NIGR)} = \frac{\text{Net Income}_{\text{Current Year}} - \text{Net Income}_{\text{Last Year}}}{\text{Net Income}_{\text{Last Year}}} \times 100 \]  

(28)

Revenue Growth Rate ratio (RGR) is similar to the NIGR ratio. The difference is that in this case, the revenue is used instead of the net income. With the revenue information is possible to measure how fast a company is expanding and making money. Calculating the RGR provides the investor with clear information about how much money is entering the company, due to sales, services or other company’s activities, and at which rate it is growing.

\[ \text{Revenue Growth Rate (RGR)} = \frac{\text{Revenue}_{\text{Current Year}} - \text{Revenue}_{\text{Last Year}}}{\text{Revenue}_{\text{Last Year}}} \times 100 \]  

(29)

3.9. Validation Measures

In this work, the valuation comprehends three main calculations: (1) the portfolio’s performance was evaluated using the fundamental indicator Return on Investment (ROI); (2) the portfolio’s volatility/risk, evaluated based on the calculation of the portfolio’s variance; (3) the risk adjusted associated to the investments made throughout the portfolio simulation is calculated using the Sharpe Ratio. The system validation is completed through the analysis of some quantitative measures, which will help the reader to understand the quality of the solution. The ROI consists in a simple profit formula showing how much percent the user got from his investment:

\[ ROI = \frac{\text{Final Return} - \text{Investment}}{\text{Investment}} \times 100 \]  

(30)

The variance is a probability distribution measure which measures how spread are the numbers in a data set. The calculation follows the formula:

\[ \sigma^2 = \frac{\sum (X_i - \mu)^2}{N} \]  

(31)

Where \( X_i \) is the portfolio value for day \( i \), \( \mu \) is the mean of the portfolio value for all simulation period and \( N \) is the number of the days in the simulation period. Applying the variance formula to this data set provides the investor with an idea of how disperse are the portfolio values from the mean daily return of the portfolio. An highly volatile portfolio is undesirable for most investors, as it means a great amount of uncertainty and risk about the portfolio’s value. The Sharpe ratio is a referenced metric used in finance to evaluate the risk/return of an investment. This ratio gives a perception to the investor about how much extra volatility he will have to deal with for the extra returns, this is, how much more risk is he taking for the higher returns.

\[ \text{Sharpe Ratio}(P) = \frac{R_P - R_F}{\sigma_P} \]  

(32)

Where \( R_P \) is the average rate of return of the portfolio \( P \), \( R_F \) is the Risk-Free rate and \( \sigma_P \) is the portfolio \( P \) standard deviation. Additionally to the previous performance measure, some metrics are also used to evaluate the several strategies employed, including: (1) the annual return percentage; (2) the maximum draw down; (3) the sector distribution, which evaluates if during the simulation the best sectors detain indeed more slots than the other; (4) the correlation with the market, called Beta metric.

4. Results

In the figure 5 are presented the general parameters of the environment set to the simulation process, both stated in the previous paragraph and others that are not entered/specified by the user, along with the parameters of the genetic algorithm.

![Figure 5: Simulation environment (left) and Genetic Algorithm configuration (right).](image)

In the figures 6 and 7, are exposed the returns (ROI) in three different abstraction levels. The best results per period are highlighted. In figure 6, the reader can see which way to order the sectors performed best given the combination Version + Strategy (sector ordering abstraction). As an example, using the version Benchmark and the strategy B, in 2014 the better results were achieved when the sectors were sorted by their EPS Growth. In figure 7, on the left side, the abstraction goes to the strategy level. The return value for the abstraction is calculated doing the mean return of the return obtained for the four possible sector sorts. On the right side on the figure, the abstraction goes to the version level, this is, the mean return of the version considering the three strategies.

From the tables represented in figures 6 and 7, is already possible to state a few comprehensive appreciations: (1) the Benchmark-A-Revenue Growth combination outperformed...
the other 35 combinations in 3 of the 4 periods. As it is the best combination, it will be more exhaustively explained as a specific case study; (2) in 3 out of 4 periods, the Benchmark-A achieved the best results; (3) in all 4 periods, the Benchmark version outperform the other two versions; (4) each period has its own “favourite” way to order the sectors; (5) in 2014, the results suggest that using the Earnings Per Share or the Earnings Per Share Growth for sector ordering provides better returns; (6) for bad years, like 2011, both other versions, Minimum Exit and Timing, present similar returns (slightly lower).

Figure 6: Best sector ordering for each combination Version-Strategy.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Benchmark</td>
<td>Revenue Growth</td>
<td>Revenue Growth</td>
<td>Revenue Growth</td>
<td>Revenue Growth</td>
<td>EPS Growth</td>
</tr>
<tr>
<td>Minimum Exit</td>
<td>Revenue Growth</td>
<td>Revenue Growth</td>
<td>Revenue</td>
<td>EPS</td>
<td></td>
</tr>
<tr>
<td>Mandatory Timing</td>
<td>Revenue Growth</td>
<td>Revenue Growth</td>
<td>Revenue</td>
<td>EPS</td>
<td></td>
</tr>
</tbody>
</table>

Figure 7: Best strategy for each version (left) and best version (right).

4.1. Case Study I - Benchmark A Revenue-Growth

The case study represents the best results obtained. The combination employed was: Version Benchmark + Strategy A + Sectors sorted by Revenue Growth. In the figure 8 the reader can observe the results concerning the case study I. For comparison purposes, the results for the S&P 500 index are also represented in this table, along with the improvement obtained when comparing with the results obtained in this simulation.

Figure 8: Case Study I - Validations results comparison.

<table>
<thead>
<tr>
<th>Measure</th>
<th>S&amp;P 500 Index</th>
<th>Avg. GA Execution</th>
<th>Difference</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROI (%)</td>
<td>0</td>
<td>13.41</td>
<td>13.99</td>
</tr>
<tr>
<td>Standard Deviation</td>
<td>0.05</td>
<td>0.008</td>
<td>0.007</td>
</tr>
<tr>
<td>Sharpe Ratio</td>
<td>-0.07</td>
<td>0.40</td>
<td>2.01</td>
</tr>
</tbody>
</table>

Figure 9: Case Study I - Variance/ROI graph for the 50 executions in the period 2013.

Looking into the genes of the chromosome, represented in figure 12 which triggered the best execution, and comparing with genes from the average execution, its possible to deduct a few conclusions: (1) A key factor for the positive Study I provides more stability and less risk. Finally, the Sharpe ratio registered for the case study are higher, meaning that when adjusted to risk, the Case Study I investment is more attractive. The values already presented are mean values calculated with the results obtained from 50 executions of the same simulation. In figure 9 is represented the graph for the four year period.

As the reader can see, the mean solution strongly surpasses the index, ending up with almost twice the index’s ROI. The best solution triples the index’s ROI. Of course, the best case is a one time situation, although it happened, so it means that is possible to achieve those results with the right combination of ratios, weights and strategies.

The mean variance of the executions for this case study also surpasses the index. In the figure 10 are exhibited the ROI-Variance results for all executions, along with the index result. The reader can observe that the variance values are very similar, but the ROIs produced by the case study are considerably higher. The case study proves to present better returns for the same risk, and in some cases better return with lower risk. In figure 11 are the measures and metrics for the S&P 500 index and the average GA execution.

Figure 10: Case Study I - Quantitative Metrics.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>S&amp;P 500 Index</th>
<th>Avg. GA Execution</th>
<th>Difference</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROI (%)</td>
<td>0</td>
<td>13.41</td>
<td>13.99</td>
</tr>
<tr>
<td>Standard Deviation</td>
<td>0.05</td>
<td>0.008</td>
<td>0.007</td>
</tr>
<tr>
<td>Sharpe Ratio</td>
<td>-0.07</td>
<td>0.40</td>
<td>2.01</td>
</tr>
</tbody>
</table>

Figure 11: Case Study I - Quantitative Metrics.
results is limit imposed on the company selection. Only 4 out of 10 sectors are available; (2) the ideal portfolio size shall be around 10-15 assets. Focusing the investments in a small number of companies from the best sectors is more lucrative than spread the investment to much; (3) the “winning” fundamental ratios are: debt ratio, price to earnings growth, return on equity, revenue growth rate and revenue rate. All of them are growth potential indicators, which implies that the best results were achieved when money was invested in companies presenting high growth potential. The Case Study I is proven to be a superior strategy. It is demonstrated that it offers higher profits and more stability than the S&P 500 index. From the ROI evolution chart (figure 9) the reader can observe that, despite the fact that both index and evolutionary strategy follow the same trends, the average execution of the evolutionary algorithm surpassed the index. This is justified with three reasons: (1) although the trends are similar, when the index curve is going up, the GA curve goes up further; (2) in several occasions, in a falling scenario, the GA curve can resist the down trend better than the index; (3) the GA performs intelligent investment decisions, this is, by choosing the best sectors, the ROI receives high boosts when in an upward trend, surpassing the S&P 500 index growth. Looking into the variance graphs (figure 10), is notorious the supremacy of the GA solution. Almost every executions provide more stability for the same revenue, being that an high percentage of them even provides higher revenue for lower variance (risk).

5. Conclusion

The analysis of the results obtained in this work allows the conclusion that the Evolutionary Algorithm plus Stock Market combination is a very powerful tool when it comes to the management of financial portfolios. The ability to solve and optimize problems with thousands of variables in a few seconds demonstrated great potential when applied to the buy/sell of companies’ assets. In this work the reader has at his disposal the results obtained from a software combining genetic algorithms with the American stock market, more precisely, companies from the S&P 500 index from 2010 to 2014. Several strategies were tested and documented. Dividing the market by industries or sector has proven to be a reliable and profitable game plan. Exploring sectors presenting higher revenue and revenue growth provides investors with good and more stable returns. Additionally, this strategy successfully selects the growing companies, based on fundamental ratios. Additionally, a few suggestions are left as possible improvements to the system: (1) extend the

<table>
<thead>
<tr>
<th>Genes</th>
<th>Chromosome Values</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Average (50 Executions)</td>
</tr>
<tr>
<td>Current Ratio</td>
<td>4.15</td>
</tr>
<tr>
<td>Debt Ratio</td>
<td>1.18</td>
</tr>
<tr>
<td>Earnings Per Share</td>
<td>1.64</td>
</tr>
<tr>
<td>Net Income Growth</td>
<td>3.23</td>
</tr>
<tr>
<td>Price to Book Value</td>
<td>2.05</td>
</tr>
<tr>
<td>Price to Earnings</td>
<td>1.29</td>
</tr>
<tr>
<td>Price to Earnings Growth</td>
<td>2.77</td>
</tr>
<tr>
<td>Profit Margin</td>
<td>4.86</td>
</tr>
<tr>
<td>Return on Assets</td>
<td>4.49</td>
</tr>
<tr>
<td>Return on Equity</td>
<td>2.81</td>
</tr>
<tr>
<td>Revenue Growth Rate</td>
<td>0.61</td>
</tr>
<tr>
<td>Revenue Rate</td>
<td>0.51</td>
</tr>
<tr>
<td>Price to Earnings Future</td>
<td>1.81</td>
</tr>
</tbody>
</table>

Figure 12: Case Study I - Chromosome example and best individual values.
Data Module to include market data from different indexes. (2) incorporate macroeconomic indicators and technical indicators in the algorithm. (3) extend the algorithm to allow different strategies during the same test. It would be of great profit if the algorithm could adjust its strategy to the current market situation instead of using the same strategy during the entire period; (4) improve the “environment” of the simulation. New features should be included, such as allow short selling and implement dividends distribution; (5) explore new risk measures and ratios besides the variance and Sharpe ratio;

References


