Multi-Objective Portfolio Optimization and Rebalancing Using Genetic Algorithms with Local Search

Vishal Soam
School of Electrical Engineering
The University of Tokyo
Tokyo, Japan
Email: vishal@iba.t.u-tokyo.ac.jp

Leon Palafox
School of Electrical Engineering
The University of Tokyo
Tokyo, Japan
Email: leon@iba.t.u-tokyo.ac.jp

Hitoshi Iba
Grad School of Information, Science and Technology
The University of Tokyo
Tokyo, Japan
Email: iba@iba.t.u-tokyo.ac.jp

Abstract — The Portfolio Optimization problem is an example of a resource allocation problem with money as the resource to be allocated to assets. We first have to select the assets from a pool of them available in the market and then assign proper weights to them to maximize the return and minimize the risk associated with the Portfolio. In our work, we have introduced a new “greedy coordinate ascent mutation operator” and we have also included the trading volumes concept. We performed simulations with the past data of NASDAQ100 and DowJones30, concentrating mainly on the 2008 recession period. We also compared our results with the indices and the simple Genetic Algorithms approach.

I. INTRODUCTION

Various institutions, funds like mutual funds and hedge funds, rich individuals etc., use the Portfolios. In ideal market conditions when most of the stock prices keep rising and all the assets give you good returns, it is easy to choose an asset and invest in it. But usually this does not happen and all assets come with some risk associated with them. Moreover, although keeping the money in cash has almost zero market risk associated with it, according to “The time value of money” theory money loses value over the period of time. For example, 100 yen paid today to an individual has more value than 100 yen paid next year. Therefore, it is meaningful to invest money through some strategy. However, since market conditions are never really ideal and markets can change suddenly, it is also risky to invest the entire amount in just one or two assets. In this kind of ever changing market conditions, investing in a diversified Portfolio is wise.

Most of the Portfolio Optimization problems are based on Markowitz’s [1] Modern Portfolio Theory (MPT). It says that diversification can minimize the risk of a Portfolio. Although written in 1950s, researchers still use this model as a base to solve the problem of optimal distribution of capital in order to minimize risk and maintain a target return. On the other hand, although the MPT forms a strong base for this problem, it also ignores the real life constraints associated with it like changing the Portfolio as the market changes to keep the risk minimized and to maintain the targeted return.

The Portfolio Optimization problem has been receiving a lot of attention from the Evolutionary Computation community in the recent years. However, with a large number of assets involved in the problem, the search space tends to become huge. Moreover, the inclusion of real-life constraints like trading costs, lots constraint, trading volumes limit and so on makes the problem non-continuous and unsolvable by the numerical methods [3]. Methods like Genetic Algorithms have been showing good results. However, the problem is big and some of the works concentrate only on the selection of assets or single scenarios. In our work, we use our algorithm to manage the Portfolio for long term. We first select the assets and form our Portfolio. We then rebalance the Portfolio for multi scenarios. We performed the simulations and compared our results with the indices’ performance and the results of simple Genetic Algorithms.

II. THE PORTFOLIO OPTIMIZATION

The Portfolio Optimization problem falls under the traditional resource allocation problem. Here the job is to distribute a limited resource to a number of jobs while satisfying some utility functions [2]. Here, the limited resource is the capital available for investment and the jobs are the varied assets in which we invest this capital (like stocks, bonds, foreign exchange assets etc.).

However, Portfolio selection is not just a problem of finding attractive investments that we could solve by intuition. The behavior of a Portfolio can be quite different from the behavior of its individual components. Powerful optimizers like Genetic Algorithms exploit the covariance, the expected returns, the risks and other constraints to obtain an optimized Portfolio. We show a basic Genetic Algorithm in Fig. 1.

A. The Markowitz’s Model

The MPT says that it is not enough to look at the expected risk and return of any particular stock. Diversification, which is investing in various stocks, can reap a reduction in the risk.
involved and a better expected return. In other words, the risk in a diversified Portfolio will be less than the risk involved in holding any one of the individual asset.

It says that, if we have N assets available in the market, a Portfolio P can be defined as a set of N real valued weights \((w_0, w_1, w_2, ..., w_N)\). The restrictions on these weights are as follows [4]:

\[
\sum w_i = 1 \quad (1)
\]
\[
0 < w_i < 1 \quad (2)
\]

To evaluate the fitness of the Portfolio, estimated return and risk are used. We calculate them as follows:

\[
R_p = \sum r_i w_i \quad (3)
\]
\[
\sigma_p = \sum \sum \sigma_{ij} w_i w_j \quad (4)
\]

where N is the total number of assets in the Portfolio, \(r_i\) is the given estimated return of asset \(i\), \(w_i\) is the weight of the \(i\)th asset. Again, \(p\) is the total risk of the portfolio, \(ij, i \neq j\) is the covariance between \(i\) and \(j\) and \(ii = i^2\) is the deviation of the estimated return of asset \(i\). This is the general way of estimating the expected risk and return, although other works suggesting the use of other methods exist as well [5], [6].

We can either use these two utility measures separately or combined to determine the optimal Portfolio. The **Sharpe Ratio** is a combined way of using them and is defined as follows:

\[
S_r = \frac{E(r_p) - R_{riskless}}{\sigma_p} \quad (5)
\]

In equation (5), \(R_{riskless}\) is the risk-free rate (an ideal case which usually doesn’t exist in real-life), an asset which has zero risk and a low return rate. They can be thought of as money in cash or an asset like bonds of an economically stable country. Fig. 2 explains the relationship between these three utility measures.

B. Dynamic Financial Markets and Their Behavior

No asset’s price keeps on increasing or decreasing forever (bankruptcy could be final end though).

The MPT assumes the Portfolio Optimization problem as a static problem and says that the past or present of the Portfolio has no effect on the future of the Portfolio. However, economic changes, market changes, political changes, industry changes and various other changes in the policies makes this a dynamic problem.

Therefore, generating a Portfolio is not the end of the problem; it is indeed, the beginning and we need to change the Portfolio from time to time according to the changes in the market or assets. Thus, since the market is dynamic, so should be the Portfolio. We in our work, try to change the Portfolio in accordance to the market changes, in a way that it keeps the expected return maintained and also keeps the risk to its minimum level. We call it rebalancing of the Portfolio.

C. Previous Works

On one hand, there are works that use a single real valued array to keep the weights of the assets selected in the Portfolio [7], [8]. For just the selection of the assets for the Portfolio, binary array can be used.

On the other hand, the relatively newer approach is to use a combination of both real valued and binary arrays [9], [10]. Here, in the binary array, 0’s and 1’s represent the absence and presence of an asset in the Portfolio respectively. The real valued array is used to keep the weights assigned to the assets. This approach has shown better results than the approach using either of the two above mentioned arrays.

We have also seen works using a tree structure through Genetic Programming to generate a Portfolio and also rebalancing it through a ranking of assets [4], [11].

Trading Strategies is one other popular approach in the evolutionary computation community. This is where some strategy is build, based upon which assets are evaluated individually.
Later, based upon some rules it is decided if that particular asset should be included in the Portfolio. While some works have built rules related to moving averages of the assets [12], some others have used some Genetic Programming rules for the same [13], [14]. However, these Trading Strategies usually assign equal weights to all the assets after they have been selected for the Portfolio. While the selection of assets through these approaches might work well, the assigning of equal weights to all the selected assets will most likely not provide an optimized Portfolio to us.

III. OUR GENETIC ALGORITHM

In our work, we first select the assets for the Portfolio using their past data. We then generate an optimized Portfolio and rebalance it for many future months later.

Our motivation to implement greedy coordinate ascent mutation operator is to fine tune the weights of the assets in the Portfolio. In our preliminary approach, where we used simple mutation operator, did not give us bad results as such, but the Portfolio itself was a bit unrealistic and needed some improvements. For example, performing the simulations with NASDAQ100 dataset, our simple Genetic Algorithm resulted in a Portfolio including around 40-50 assets (depending on rebalancing) with many assets holding significantly small weights. These small weights not only make it difficult to buy the Portfolio but it also makes rebalancing difficult and troublesome (in real life, where you have to buy and sell assets with transaction costs for each and every transaction).

The next motivation is to introduce some strategy which takes into account the news which affects the market significantly. Since, the inclusion of all the relevant news is not possible, we came up with this idea of including the daily trading volumes, which indirectly reflects the changing sentiments of the market and the industry i.e. customers’ reaction to the latest news.

A. Representation of a Portfolio and Selection Strategy

While some works, mainly concentrating on the selection of the assets for the Portfolio only, use the binary array (0s and 1s) to show the absence or presence of an asset, some other researches use real valued array representations to accommodate the weights of the assets.

We have used both the array representations in our work 3. We apply crossover operator to both of these arrays in the later steps. After creating the population, in each of the successive generations, we select some of the individuals from the existing population to breed new individuals for the next generation. We select the individuals through a fitness based process and follow the strategy of selecting individuals with better fitness value (calculated by fitness functions) for breeding.

Roulette Wheel Selection and Tournament Selection are some of the most commonly used techniques. In our work, we use Sharpe Ratio as the fitness function to calculate the fitness of the individuals and then apply Deterministic Tournament Selection (DTS). DTS works well with small subsets of the population. It is appropriate for a parallel implementation since it breaks the selection problem into many small independent parts and thus speeds up the processing.

B. Crossover Operator

Now that we have the selected individuals, we have to apply some operators to try to change them to produce new offspring. Crossover usually contains the information of both the parents which form the offspring.

We used K-point crossover technique. Here, we randomly choose K-points in the genetic representation of the two parents. Then we produce the offspring by alternatively copying elements from the two parents, changing from one parent to the other at each of the k points. This is shown in Fig. 4. Here, we do the vector multiplication of the binary and real valued arrays for both the parents respectively, and use the resulting arrays for crossover. Finally we normalize the resulting array (the offspring) to hold Eq. 1

C. Greedy Coordinate Ascent Mutation Operator and Local Search

Mutation operator forms the offspring by altering the values of the genes in the array. This operator is more like exploration of the search space. We used a two-step mutation. In the first step, we randomly alter the gene values, binary values in the first sub step and real values in the second sub step, and then normalize the array (Note that for normalization we always use the vector product of binary array and the real valued array). We then evaluate it and apply the second step. The second step is explained in Fig. 5. Here, we try to alter one particular gene, both by adding some value and by subtracting the same value, while keeping all the other genes fixed. The individual
Algorithm 1

```
Sharpe = Sharpe(weight)
for (i=0; i<portfolio array size; i++)
    base = weight[i]
    weight[i] = weight[i] * alpha
    Plus_weight = weight / (1 + alpha)
    Sharpe_Plus = Sharpe(Plus_weight)
    weight[i] = 2 * alpha
    if (weight[i] < 0)
        weight[i] = 0
    Minus_weight = weight / (1 - base)
    else
        Minus_weight = weight / (1 - alpha)
    Sharpe_Minus = Sharpe(Minus_weight)
    return weight of max_Sharpe
```

Fig. 5: Guided local search co-ordinate ascent mutation operator

is then evaluated, after normalization, and the best of the three (original, one with added value and the one with subtracted value i.e. Sharpe, Sharpe_plus and Sharpe_minus) is chosen. Therefore, we chose the best possible value for a particular gene while all other genes’ values remain unchanged, in one step.

We then repeat the same process for all the other genes, which acts like a local search. While, this mutation operator is computationally a bit costly, it has shown better results than the normal mutation operators. It has also shown good results with the problem of the presence of insignificant weights in the final Portfolio; this mutation operator removes all the insignificant weights.

D. Changing the Portfolio: The Rebalancing Problem and Seeding

Markets are almost never stable. Markets are dynamic and keep changing which makes the Portfolio problem dynamic as well. Once we have generated the Portfolio, the problem is not over. Since the prices, and thus returns and risk, of the assets included in the generated Portfolio keep changing, there arises a need to keep changing the Portfolio accordingly, to keep the returns of the Portfolio stable and risk minimized.

However, buying or selling of an asset, both require transaction costs. Therefore, the objective of our problem changes slightly. Now we not only have to take care of the return and risk, but also to change the Portfolio in such a way that the change in the expected return of the rebalanced Portfolio is greater than the transaction costs we have to pay. We have seen the use Euclidean Distance as a cost measure in rebalanced Portfolios [11]. However, we have used the real-life technique to calculate transaction costs as shown in equation ??.

\[
f(x_i) = \begin{cases} 
1, & \text{if } f(x_i) > 0 \\
0, & \text{if } f(x_i) < 0 
\end{cases}
\]  \( (6) \)

Here, \( Cost_i \) is the transaction cost, \( T_i \) is the amount of ith asset traded, \( Cost_{fixed} \) fixed amount charged for transactions below a certain amount, \( T_{min} \), and \( c \) is some small \% of the total transaction value.

Seeding is another technique which we use to tilt the search in a particular direction. We have used seeding, which brings some of the best individuals from the scenario at time \( (t-1) \) to the scenario at time \( t \). Fig. 6 shows seeding.

E. The Concept of Trading Volumes

In real world, the world news plays a very important role in driving the prices of all sorts of assets. The political news, the news about some company declaring the dividends for a quarter/year, the forecast about the oil prices etc.: all of this news affects the prices of the assets related to them. Therefore, we believe that somehow all the world news should be incorporated in the system selecting and managing the Portfolio.

At the same time, the incorporation of all this news in the system can be extremely difficult. Which news to incorporate and how to build a system that makes a good use of this news while managing the Portfolio: this can be a tedious job.

However, as soon as some important news reaches the market, we can see its affect through the volume of the related asset traded in the market on that day or in the coming days. For example, if there is some news about a company that might go bankrupt in the coming days, most of the (all?) stockholders in the company start selling their stocks to avoid the loss. As a result, the volume of the asset traded in those days is much bigger than the average volume traded usually. This information is extremely important and gives us an idea about the trend of the prices of the assets in the coming days.

Therefore, since trading volumes have this important hidden information in them, we weight the returns accordingly taking into account the traded volume ratio concept. How to use trading volume ratio with the returns is shown in Fig. 7. Here, \( R_{c,t} \) Return of asset a on the ith day of period t, \( V_{it} \) is the trading volume ratio = \( V_{at}/V_{it} \). \( V_{at} \) is the traded volume on the ith day of period t, \( V_{it} \) is the average trading volume in the time period T. The volume ratio element inflates or deflates the return (risk) depending on the volume traded, providing us with a good idea of when to rebalance the Portfolio.
Algorithm 2

\begin{algorithm}
\begin{algorithmic}
\State for a period of time \( t \)
\State calculate average traded volume \( V_t \)
\State \( R_{at} = R_{at}*V_{at}/V_t \)
\end{algorithmic}
\end{algorithm}

Fig. 7: The Usage of Trading Volume Ratio with Returns

IV. TEST PROBLEMS AND EXPERIMENTS

In our work, we testify the proposed Greedy Coordinate Ascent Mutation Operator and the inclusion of the concept of trading volumes in the calculation of returns by several simulations.

We first evolved the Portfolio, and then rebalanced it for the next 18 scenarios, i.e. one and a half years (18 months) using our proposed algorithm, concentrating mainly on the time period of the recession of 2008 and the period following it, when most of the major assets showed losses, and compared them with the results from simple Genetic Algorithms and also the indices. The main aim is to check the working of the Greedy Coordinate Ascent Mutation Operator and the how the trading volumes influence the management of the Portfolio.

A. Datasets and Experimental Settings

We used two datasets for the simulations. The first one is the Dow Jones Industrial Average, which includes 30 assets in total. The data is a small sized set and we have included 29 out of the 30 assets in the simulations, due to non-availability of data of one of the assets. The second one if the NASDAQ100 data set, which has a total of 100 assets. The dataset is medium-large sized set and we have included all 100 assets in the simulations.

We used the data for the time period from January 2007 till December 2007 as the training data for the evolution of the first Portfolio. We then rebalanced the Portfolio for the time period of one and a half years from January 2008 till July 2009, i.e. 18 scenarios, rebalanced on the last day of each month.

Parameters are as follows: Number of generations 200, crossover rate 0.2, mutation rate 0.2, for seeding the elite size 100 individuals. We used the Japanese 10-year bonds with the rate of return of 1.035% as the riskless asset. In the mutation step, the parameter alpha is 0.005.

B. Experimental Results and Discussion

For both the cases, Dow Jones Industrial Average and NASDAQ 100, the results are shown in Fig. 8,9,10,11 and tables I and II. Here, “Full Portfolio” represents our Portfolio where both, Greedy Coordinate Ascent Mutation Operator and the trading volumes, are included. “Portfolio no News” represents the method in which Greedy Coordinate Ascent Mutation Operator is included while the trading volumes (the news concept) are not included, for comparison purposes. Therefore if we compare the results of “Simple GA” and “Portfolio no News”, we can see the impact of “Greedy Coordinate Ascent Mutation Operator”. And if we compare “Simple GA” and “Full Portfolio” results, we can analyze the impact of both the new points included in our algorithm. Fig. 89 and Table I show the results of the simulations made on the Dow Jones Industrial Average 30 assets (29 assets included). Fig. 1011 and Table II shows the results of the simulations made on the NASDAQ 100 assets (all 100 assets included). In Tables I and II, the weights less than 0.001 are considered insignificant weights.

In Fig. 8, the profits achieved by all the three methods have shown better results than the index itself. However, even in the difficult times of the 2008 recession, our Portfolio has outperformed all the other three cases, including the index.

Since just the profits cannot evaluate if a Portfolio is good or not (because of the risk involved), the comparison of the Sharpe Ratio is an important step. Fig. 9 shows the results of Sharpe ratio comparisons. In this case too, all the three methods have outperformed the Dow Jones index. Although, our Portfolio has shown better results than the index and the other two methods, the dynamic nature of the market and the market instability in the recession periods is visible in the Portfolio results. Portfolio jumps and falls quite often suggesting the change in the trading volumes in assets during that period. The search space seems very bumpy, and calls for better rebalancing strategies to make the Portfolio a bit more stable.

Table I shows the number of assets selected in the final Portfolio in July 2009. The Greedy Coordinate Ascent Mutation Operator in our Portfolio works very well and all the assets selected in the Portfolio have significant weights. On the other hand, simple GA works poorly here and a lot of assets with insignificant weights exist in the Portfolio. The presence of insignificant weights makes the management of the Portfolio cumbersome and expensive due to the trading costs related to them during the rebalancing problem.

In Fig. 10 as well, the Portfolio outperforms the index and the other two methods, and again the Portfolio profits are a bit bumpy with few jumps and falls. However, in Fig. 11, the Portfolio without trading volumes performs better than our Portfolio for few scenarios. This suggests that although the profits by our Portfolio have been better than the other two methods and the index, the risk has been higher than that of the Portfolio without trading volumes in some cases. However, the Sharpe ratio is comparatively stable than in the case of Dow Jones. This might be due to the difference in the composing assets of the two indices. The number of assets with insignificant weights in our Portfolio is again 0, while the simple GA has a lot of them.

V. CONCLUSIONS AND FUTURE WORKS

We used two new points in our work, namely the Greedy Coordinate Ascent Mutation Operator and the trading volumes. We proposed the inclusion of trading volumes in the Portfolio as an alternative of inclusion of world news, which usually has a strong impact on the financial markets. We performed the simulations with the data of Dow Jones Industrial Average.
TABLE I: The comparison of the number of assets with significant and insignificant weights selected for the last Portfolio in the month of July 2009 in case of Dow Jones Industrial Average 30 assets (29 assets included)

<table>
<thead>
<tr>
<th>Dow Jones</th>
<th>Number of assets with significant weights</th>
<th>Number of assets with insignificant weights</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simple GA</td>
<td>10</td>
<td>14</td>
</tr>
<tr>
<td>Portfolio no Vol</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>Portfolio</td>
<td>5</td>
<td>0</td>
</tr>
</tbody>
</table>
30 assets and the NASDAQ 100 assets for the difficult market situations during the recession of 2008. The results achieved by the simulations are very motivating. Our Portfolio not only performs better than the index and the simple GA, it also solves the problem of the availability of assets with insignificant weights. Therefore, our Portfolio can be a good alternative to other available methods for the problem of Portfolio Optimization.

However, the real application of our proposed method requires some modifications and some more works. One of the future works is to include the LOTS constraint, which says that it is not possible to divide an asset indefinitely. Other future work is the modification of the rebalancing strategy. In other words, rather than rebalancing the Portfolio on the last day of each month, we should rebalance it whenever an upcoming positive or a negative trend in the prices of the assets is detected.

### REFERENCES


