Optimising a FTSE MIB's portfolio using Multi-Objective Evolutionary Algorithms based on decomposition

Davide Liso

March 30, 2021

Abstract

Choosing the optimal weights to invest in individual stocks so as to obtain a portfolio that maximises returns and minimises risk has always been one of the main problems in mathematical finance. A class of optimisation algorithms called Multi-Objective Evolutionary Algorithms based on decomposition (MOEA/D) will be used in order to solve this task. In this project, those algorithms that are currently considered to be at the state-of-the-art will be used and compared. Specifically, MOEA/D-DE, MOEA/D-DEM, MOEA/D-GA, MOEA/D-Levy and NSGA-II, will be proposed as numerical methods to optimise a portfolio consisting of all 40 stocks of the FTSE MIB index.

1 Portfolio Optimisation Model

The most famous portfolio optimisation model was introduced by Markowitz [1]. This model assumes that investors would like to maximise return under a certain risk level or minimise the risk with a certain return level. It can be represented as a bi-objective problem as follows:

$$Max\ r_p = \sum_{i=1}^{N} w_i r_i \tag{1}$$

$$Min\ \sigma_p^2 = \sum_{i=1}^{N} \sum_{j=1}^{N} w_i w_j \sigma_{ij} \tag{2}$$

with constraints

$$\sum_{i=1}^{N} w_i = 1;\ 0 \le w_i \le 1$$

where N is the number of assets of the portfolio, w_i is the weight vector of the ith asset. σ^2 and r_p are the variance and the expected mean of portfolio returns respectively. σ_{ij} is the covariance of returns between asset i and asset j. This is the original form of Markowitz model adopted in this project, although there are several extensions of the original model which include transaction costs and other constraints, such as limits on the proportion of an asset in the portfolio.

2 Multi-objective optimisation model

MOEA/D algorithms consist in decomposing a multi-objective optimisation problem into several single objective optimisation problems and optimising them simultaneously. The decomposition method used here is called NBI-style Tchebycheff decomposition, explained in [2] and [4]. In MOEA/D, the optimisation problem is often presented as follows:

$$\min_x g^{te}(x|\lambda, z^*) = \max_{m=1,\dots,M} \{\lambda_m |f_m(x) - z^*_m|\} \tag{3}$$

$$\lambda = (\lambda_1, \dots, \lambda_m); \ \sum_{i=1}^{m} \lambda_i = 1) \tag{4}$$

$$z^* = (minf_1, \dots, minf_M) \tag{5}$$

where λ is the weight vector, z^*_m (also known as *reference point*) is the minimum value of the m-th objective. In this case, the GA operator is adopted in order to generate an offspring. Every population is is optimised as the solution of one single objective optimisation problem decomposed using one weight vector. During optimisation, the solution uses the information of its neighbor. The neighbor is defined as solu- tions of neighboring SOOPs, which can be computed using the closest Euclidean distance between weight vectors. When reproducing an offspring, MOEA/D selects parents from the neighbors of the current individual. Once the offspring are generated, it can be used to update all neighbors of the current individual. There are many variants of mutation methods of MOEA/D. Here I used the MOEA/D-DE, introduced by [3], which uses the DE and polynomial mutation operator into MOEA/D. Both are presented as follows

$$offspring = \begin{cases} x^i + F \cdot (x^j - x^k) & \text{if } uniform(0,1) < crossover\ rate \\ x^i & \text{if } uniform(0,1) \geq crossover\ rate \end{cases} \tag{6}$$

where x^i, x^j and x^k are parents. The NBI-style Tchebycheff decomposition method transfers vector optimisation into scalar optimisations with a normal vector λ of convex hull of minima (CHIM) and evenly distributed reference points $\{z^1,\ldots,z^N\}$ where F^1 and F^2 are the extreme points. Since we are dealing with a bi-objective optimisation problem, CHIM is the line between F^1 and F^2.

$$\min_x g^{tn}(x|\lambda, z^*) = \max_{m=1,2}\{\lambda_m f_m(x) - z_m^*|\} \tag{7}$$

$$\lambda = (\lambda_1, \lambda_2);\ \lambda_1 = |F_2^2 - F_2^1|; \lambda_2 = |F_1^2 - F_1^1| \tag{8}$$

$$z^i = a_i \cdot F^1 + (1 - a_i) \cdot F^2,\ a_i = \frac{N-i}{N-1} \tag{9}$$

The other variants used in this project have been also proposed and compared in Yifan He and Claus Aranha study. For all the details, check [5].

3 Experimental Results

In this project, the aim is to assess the performances of four variants of MOEA/D mutation methods and NSGA-II. The dataset used is available in the repository of my Github account[1]. It consists of all 40 stocks listed in the index from April/2019 to February/2021

In the following table, there is a summary of mutation methods adopted in this project. In order to evaluate how good MOEAs are, there are several metrics that consider convergence and diversity performances. The Maximum Spread (MS) [6] metric is adopted here as a diversification metric. A larger MS indicates a better performance. It is defined as follows:

$$\sqrt{\sum_{m=1}^{M}(\max_{i=1,\ldots,N} f_m^i + \min_{i=1,\ldots,N} f_m^i)^2} \tag{10}$$

where f_m^i is the m-th objective of i-th non dominated solution. The experiment is repeated 50 times. The parameters are the following: for all methods, popuulation size, neighbor size T, proportion σ and upper limitation n_r are 100, 20, 0.9 ,2 respectively (the same setting as in Li's paper[7]). The maximum population is set as 1000 and the stop criterion (diversity) is set when the variation of Maximum is not larger than 1e-05 for 100 continuous generations. The parameter β and a_0 of MOEA/D-Levy are equal to

[1] https://github.com/drag94/Optimising-FTSEMIB-s-portfolio-using-Multi-Objective-Evolutionary-Algorithms-based-on-decomposition/tree/main/dati%20ftsemib%20listino

0.3, 1e-05. Generally speaking, the parameters are the same as in [5].

Method	Mutation Formula
MOEA/D-DE	offspring $= x^i + F\cdot\,(x^j - x^k)$
MOEA/D-GA and NSGA-II	SBX crossover and polynomial mutation
MOEA/D-DEM	offspring $= x^i + F\cdot\,(x^j - x^k)$ and polynomial mutation
MOEA/D-Levy	offspring $= x^i\, +\, a_0\cdot(x^i - x^j) \otimes$ Levy(β)

Next figures are a visual demostration of how these kind of algorithms work. In this case, the NSGA-II is shown. Finally, the last two figures show the best set of portfolios and their relative Maximum Spread

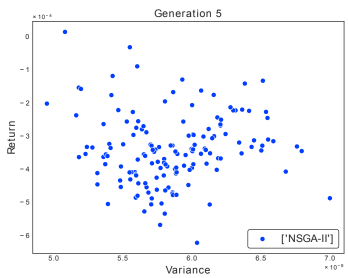

Figure 1: 5-th Generation

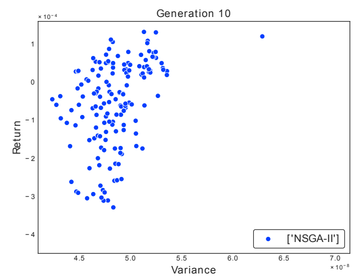

Figure 2: 10-th Generation

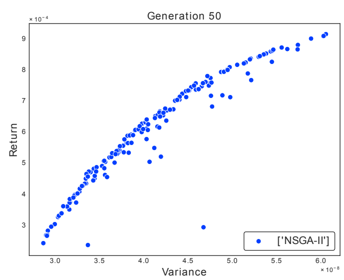

Figure 3: 50-th Generation

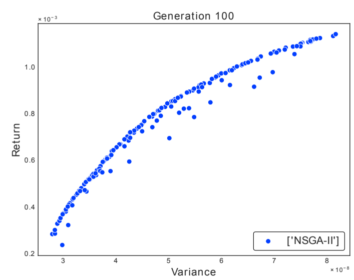

Figure 4: 100-th Generation

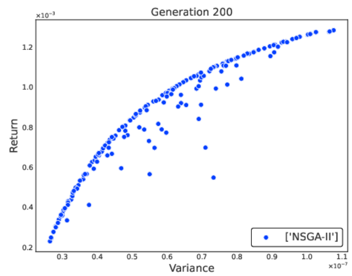

Figure 5: 200-th Generation

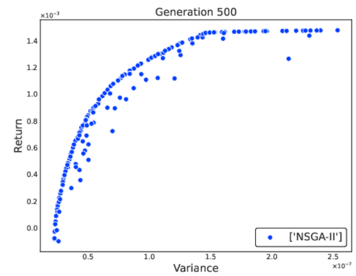

Figure 6: 500-th Generation

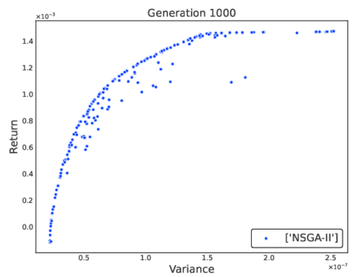

Figure 7: 1000-th Generation

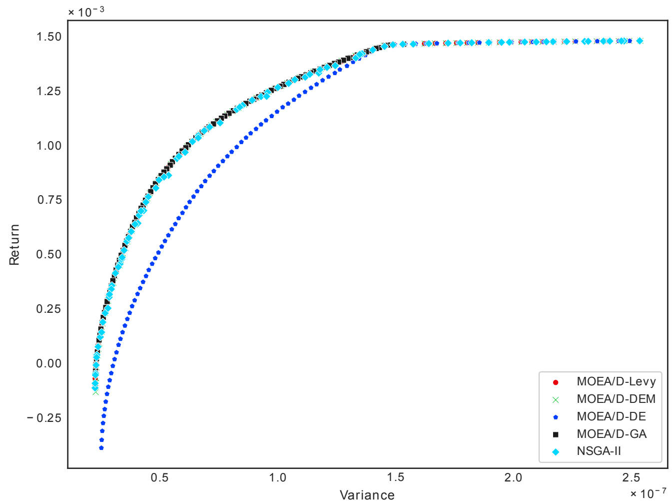

Figure 8: Final population FTSE MIB

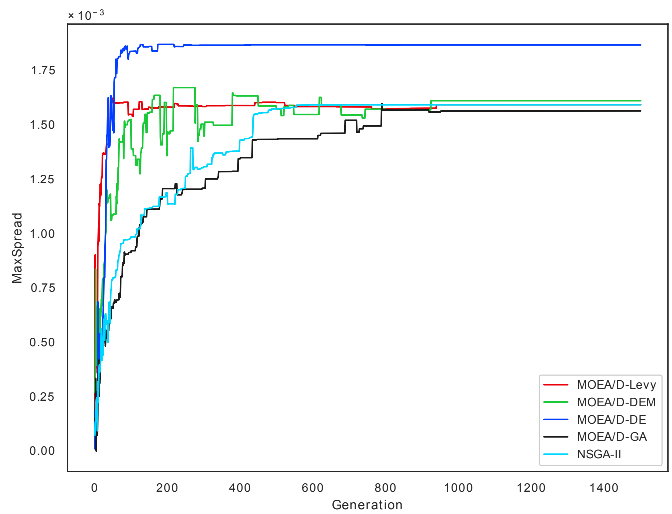

Figure 9: Maximum Spread FTSE MIB

4 Discussion and Conclusion

The results show good performances of all methods but MOEA/D-DE. Surprisingly, the worst method achieved the best Maximum spread, while the other ones achieved almost the same MS. This is certainly due to the particular composition of the dataset. In this experiment, the MS metric does not show the best method and it would be better to use multiple metrics that take into account diversity and convergence, or to use a compound metric that takes both into account, as in [8]. Furthermore, the model used in this project is fairly simple and does not consider transaction costs and there is no chance to short selling (due to the positive weights).

However, MOEA/Ds appear to be very promising as numerical methods of optimisation, according to the results of the experiment. In the future, other algorithms (belonging to the class of MOEA such as [9] and [10]) will be implemented and compared with those used in this project and there will be added to the plain model further constraints in order to make the whole process more realistic.

References

[1] H. Markowitz, “Portfolio selection,” The Journal of Finance, vol. 7, no. 1, pp. 77–91, 1952.

[2] Qingfu Zhang and Hui Li (2007). MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition. IEEE Transactions on Evolutionary Computation, 11: 712–731, doi:10.1109/TEVC.2007.892759.

[3] Hui Li and Qingfu Zhang (2009). Multiobjective Optimization Problems With Complicated Pareto Sets, MOEA/D and NSGA-II. IEEE Transactions on Evolutionary Computation, 13: 284–302,

[4] Zhang, Q., Li, H., Maringer, D. and Tsang, E. (2010). MOEA/D with NBI- style Tchebycheff approach for portfolio management, in IEEE Congress on Evolutionary Computation, IEEE, ISBN 978-1-4244-6909-3, pp. 1–8

[5] Yifan He, Claus Aranha (2020). Solving Portfolio Optimization Problems Using MOEA/D and Lévy Flight.

[6] Deb, K. (2001). Multi-objective optimization using evolutionary algorithms, John Wiley & Sons, ISBN 9780471873396

[7] Hui Li and Qingfu Zhang (2009). Multiobjective Optimization Problems With Complicated Pareto Sets, MOEA/D and NSGA-II. IEEE Transactions on Evolutionary Computation, 13: 284–302

[8] Coello Coello, C. A. and Reyes Sierra, M. (2004). A Study of the Paralleliza- tion of a Coevolutionary Multi-objective Evolutionary Algorithm, Springer, Berlin, Heidelberg, pp. 688–697

[9] Ertenlice, O. and Kalayci, C. B. (2018). A survey of swarm intelligence for portfolio optimization: Algorithms and applications. Swarm and Evolutionary Computation, 39: 36–52

[10] Ke, L., Zhang, Q. and Battiti, R. (2013). MOEA/D-ACO: A Multiobjective Evolutionary Algorithm Using Decomposition and AntColony. IEEE Trans- actions on Cybernetics, 43: 1845–1859