Tanzania Journal of Engineering and Technology 2022, **41**(**2**):141-150
OPEN ACCESS articles distributed under Creative Commons Attribution Licence [CC BY-ND]
Websites: https://www.ajol.info/index.php/tjet, https://tjet.udsm.ac.tz

ISSN 1821-536X (**print**); ISSN 2619-8789 (**electronic**)
DOI: 10.52339/tjet.v41i2.787

Full Length Research Paper

Efficient and Simple Heuristic Algorithm for Portfolio Optimization

Collether John

Department of Electronics and Telecommunications, University of Dar es Salaam,
P. O. Box 33335, Dar es Salaam
Corresponding author: nyalusicj@yahoo.com; coletha@udsm.ac.tz

ABSTRACT

Markowitz model considers what is termed as standard portfolio optimization. The portfolio optimization problem is a problem which based on asset allocation and diversification for maximum return with minimum risk. Thus, the standard portfolio optimization problem happens when the constraints considered are budget and no-short selling. In reality however, portfolio optimization has realistic constraints to be incorporated such as holding sizes, cardinality and transaction cost. When realistic constraints are added into portfolio optimization problem, it becomes too complex to be solved by standard optimization methods which in this case turns to be an extended portfolio optimization problem. Markowitz solution and the standard methods like quadratic programming become inapplicable. With such limitation, heuristic methods are usually used to deal with this extended portfolio optimization problem. Therefore, this paper proposes a heuristic algorithm for the extended portfolio optimization problem. It is a hill climbing algorithm named Hill Climbing Simple (HC-S) which is then validated by solving the standard Markowitz model. In fact, the proposed algorithm is benchmarked with the quadratic programming (QP), which is a standard method. By benchmarking HC-S with QP, it showed that HC-S can attains similar accurate solutions. Also, HC-S demonstrated to be more effective and efficient than threshold accepting (TA), an established algorithm for portfolio optimization since HC-S find solutions with significant higher objective value and require less computing time as compared to standard methods.

ARTICLE INFO

Submitted: **May 25, 2021**

1st revision: **December 27, 2021**

Second revision: **June 20, 2022**

Accepted: **June 25, 2022**

Published: **June 30, 2022**

Keywords: *Heuristic, Hill Climbing Algorithm, Portfolio Optimization Problem, Threshold Accepting Algorithm.*

INTRODUCTION

Markowitz's standard portfolio optimization model (Markowitz, 1959; Markowitz, 1952) is a mathematical framework for describing and assessing return and risk of a portfolio of assets, using returns, volatilities and correlations. Markowitz introduced what is known as the mean-variance principle, whereby future returns are regarded as random numbers and expected value (mean) of the returns $E(r)$ and their variance (whose square root is called standard deviation/ risk) capture all the information about the expected outcome and the likelihood and range of deviations from it (Markowitz, 1959; Markowitz, 1952).

To solve the portfolio optimization problem means to find the portfolio weights i.e., how to distribute the initial wealth across the available assets in order to meet the investor's objectives and constraints (Maringer, 2008; Markowitz, 1959; Markowitz, 1952). The most important constraints are budget and return constraints since they characterize the main part of the portfolio problem (Di Tollo and Roli, 2008). The return constraint is considered when the investor requires a certain level of profit from his investment with minimum risk. The budget constraint is also taken when the investor has to invest all the capital in the portfolio. However, return constraints can only be satisfied using a historical portfolio (Sharpe, 2000; Korn 1997; Markowitz, 1959; Markowitz, 1952).

Although the Markowitz model is a well-defined optimization problem, there exists no general solution for the optimization problem, because of the non-negativity constraint on the asset weight. Though the Markowitz model cannot be solved analytically, numerical methods exist by which the model can be solved for a given set of parameters (Maringer, 2008 Winker, 2001; Sharpe, 2000; Gilli and Këllezi, 2000). The capacity of these traditional standard methods relies on strong assumptions and simplifications, which do not reflect the real market situations (Sharpe, 2000). For reliable results that reflect the constraints of the real market situations, alternative optimization techniques like heuristic algorithms are usually used to deal with the extended portfolio optimization problem. Lwin and Qu, (2013) proposed a hybrid heuristic algorithm for tackling real market constrained portfolio problem. Mercangöz and Eroglu (2021) give the implementing steps of the GA heuristic to solve a portfolio optimization problem. Milhomen and Dantas (2020) did a comprehensive review of the exact and heuristic methods, used to solve the portfolio optimization problem. They found that, attention should be given to input parameters/data of optimization models for best optimization results. Doering et al (2019) reviewed the current state and future trends of the use of higher-level heuristics (metaheuristic) for portfolio optimization and risk management. Meanwhile, Silva et al. (2019) applied particle swarm heuristic approach to solve the multi-objective portfolio optimization problem. On the other hand, Meghwani and Thakur (2018) used multi-objective evolutionary algorithms to solve a tri-objective portfolio optimization model with risk, return and transaction cost as the objectives. Arriaga and Valenzuela-Rendó, (2012) proposed a simple hill climbing algorithm called steepest ascent hill climbing algorithm which gave similar results as the complex evolutionary algorithm. Moreover, Kalayc et al. (2020) present an efficient hybrid metaheuristic algorithm that combines critical components from continuous ant colony optimization, artificial bee colony optimization and genetic algorithms for solving cardinality constrained portfolio optimization problem. Other researchers who applied heuristic algorithms to deal with portfolio problem are (Aranha and Iba 2009; Maringer, 2008; Aranha and Iba 2008; Crama and Schyns, 2003; Schaerf, 2002; Gilli and Këllezi, 2000). They apply heuristic optimization techniques like simulated annealing (Kirkpatrick *et al*, 1983), local search (example tabu search (Schaerf, 2002)) and threshold accepting (Dueck and Scheuer, 1990). The most established heuristic algorithm used in extended portfolio optimization problem being threshold accepting (Gilli and Schumann, 2012; Gilli and Schumann, 2010; Winker and Maringer, 2007; Winker, 2001; Gilli and Këllezi, 2000; Dueck and Winker, 1992). Heuristic techniques seek to converge to the optimum in the course of a search, by repeatedly generating new solutions and testing them. They are flexible and not so restricted to certain forms of constraints (Gilli and Winker, 2008; Winker, 2001; Gilli and Këllezi, 2000). Actually, heuristic techniques operate on an iterative principle that includes stochastic elements in generating new candidate solutions and/or in deciding whether these replace their predecessors – while still incorporating some mechanism that prefers and encourages improvements (Maringer, 2008; Winker and Maringer, 2007). The stopping criterion is usually a fixed number of steps or if the quality of the solution does not improve after a given or specified number of iteration or both (Winker and Maringer, 2007).

In this paper, a heuristic algorithm is designed and investigated for portfolio optimization problem. The produced algorithm is tested to solve the standard portfolio optimization problem. Using the proposed algorithm, it is found to be more effective and efficient than threshold accepting (TA), an established algorithm for portfolio optimization since HC-S find solutions with significant higher objective value and require less computing time as compared to standard methods.

METHODS AND MATERIALS

Design of the proposed HC-S algorithm

The designed hill-climbing algorithm is denoted as HC-S. Here HC stands for Hill Climbing, S stands for Simple search of neighbourhood. In each step, the algorithm attempts to improve a current solution by changing the relative weight of a single asset. **ThP** stands for Threshold Percentage. It refers to the size of a step in the proposed hill climbing method. In HC-S, ThP was fixed to 0.5%.

Considering return (R_p) of a portfolio and variance (σ^2_p) of portfolio, the objective is to maximize the expected return (R), while diminishing incurred risk (σ), measured as standard deviation/variance in Markowitz (1952). Equation (1) is maximized subject to expected return [equation (2)], portfolio return variance [equation (3)] and constraints [equations (4) and (5)].

$$Max(\lambda E(R_p) - (1-\lambda)\sigma_p^2) \quad (1)$$

$$E(R_p) = \sum_i w_i E(R_i) \quad (2)$$

$$\sigma_p^2 = \sum_i \sum_j w_i w_j \sigma_i \sigma_j \rho_{ij} \quad (3)$$

where: $\rho_{ij} = 1$ for $i=j$.

Constraints:

$$\sum_i w_i = 1 \quad (4)$$

$$0 \le w_i \le 1 \quad (5)$$

The expected return of each asset is $E(R_i)$, each asset variance is σ_i, and each asset weight is w_i.

Equation (1) reflects the trade-off between return (R_p) and risk (σ_p) of portfolio. By solving the problem for different values of $\lambda \in (0, 1)$: the efficient line/frontier is then identified. If $\lambda=1$ the model will search for the portfolio with highest possible return regardless of the variance. If $\lambda=0$, the minimum variance portfolio (MVP) will be identified. Higher values of λ put more emphasis on portfolio's expected return and less on its risk (Markowitz, 1952). Equation (4) and (5) are the constraints on the weights that they must not exceed certain bounds.

Representation of solution

A current solution for HC-S is represented by a vector of numbers ($y_i,\ldots,y_n$). The element in position *i* represents the relative weight of the capital invested in stock *i*. The vector of numbers ($y_i,\ldots,y_n$) is normalized to make sure that the weights in all the assets add up to 1. The percentage/weight to be invested in stock *i* is x_i, as shown by equation (6).

$$x_i = y_i \Big/ \sum_{i=1}^n y_i \quad (6)$$

One advantage of using this representation is that the vector, y, may take any number without violation of budget constraint that the weights add up to 100%.

Definition of neighbourhood for HC-S

Figure 1 shows the hill climbing procedure for the proposed hill climbing algorithm HC-S. The procedure is for implementing the definition of neighbourhood. A current solution has two neighbours or two possible candidate solutions. Elements of vector y in the range of 0 to 100 are randomly generated. The number of elements of y is equal to the number of asset/stocks. The randomly picked position in y is denoted as pos. ThP is a small percentage, which we refer to as threshold percentage, by which elements of y will be varied to get the next neighbour. The neighbourhood definition is to pick just one position (pos) in the current solution, y, at random. After picking the random position in the current solution, one neighbour is obtained by adding ThP to that position and another is obtained by subtracting ThP on the same position. This gives two neighbours (two possible candidate solutions) to be compared with the current solution, at random. The first better candidate solution (neighbour) to be picked is taken to be the current solution out of the two possible candidate solutions. If no better solution is found, another position, pos, in y is picked at random and the procedure is repeated. The procedure is repeated for a pre-set number of iterations, or until local maximum is obtained. In the procedure (Figure 1) mean returns of all stocks in column vector are denoted as retasset, given assets' co-variances/deviations matrix are denoted as dev, and λ is the level of risk aversion.

Figure 1 shows the block diagram with the summary of the neighbourhood definition of HC-S.

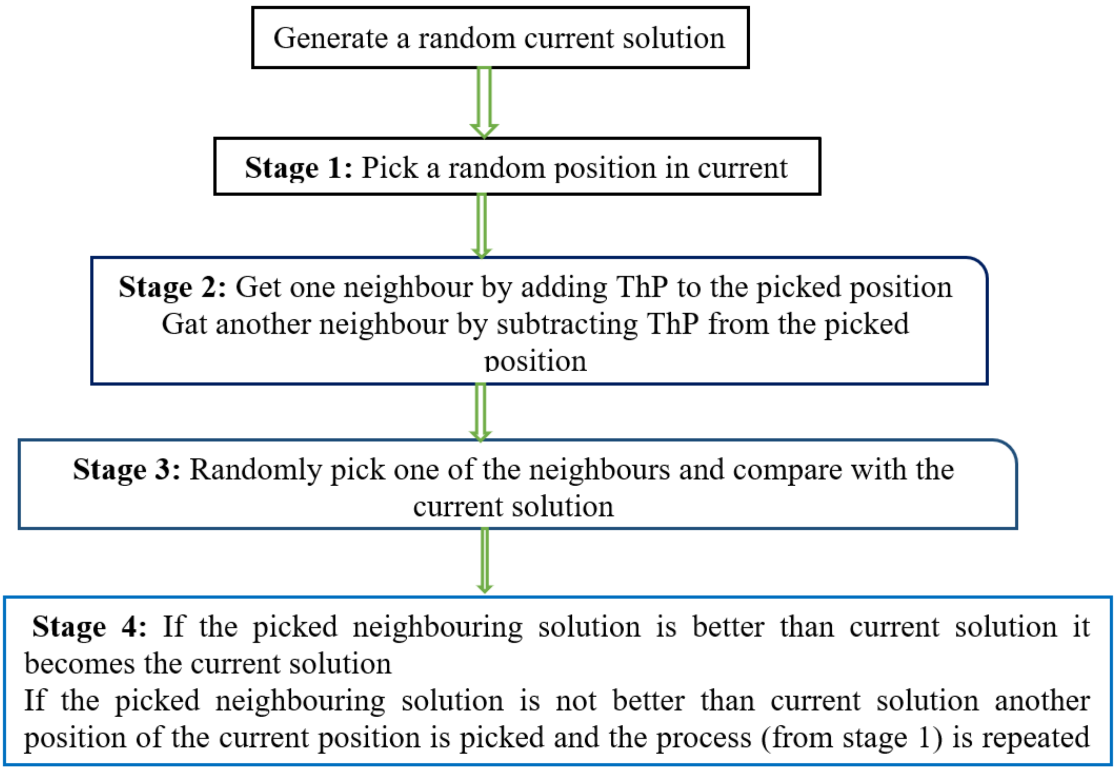

Figure 1: Block diagram showing summary of the neighbourhood definition of HC-S.

Below is the pseucode for procedure of HC-S (Function Move_to_neighbour) shows the Function to search for better neighbouring solution. It calculates the objective values of the two neighbouring solutions, compare with present solution and returns the better neighbouring solution.

Table 1: Pseudocode for hill climbing procedure of HC-S

Procedure HC-C (ThP, λ, retasset, dev)	
Randomly generate initial current solution y	
Begin	
Repeat	
pick random position, (pos), in current solution y	% Generate yplus from current solution%
yplus = y	%Generate yminus from currentsolution%
yminus = y	% Get a neighbour of current solution %
yplus(pos) = yplus(pos)*(1 + ThP)	% Get second neighbour of current solution %
yminus(pos) = yminus(pos)*(1 - ThP)	% Pick a better neighbour solution %
yb4=y	
y = **move_to_neighbour** (y, yplus, yminus, λ, retasset, dev)	% Stopping criterion; (no neighbour is better than current solution or pre-set maximum number of iterations reached)%
Until halting criterion is met	
End.	

Table 2: Pseudocode of a function to search for better neighbouring solution

Function Move_to_neighbour (y, yplus, yminus, λ, retasset, dev)	
Begin	
$x_i = y_i \Big/ \sum_{i=1}^{n} y_i$	% Find weights, x, of all the assets n in portfolio%
$xplus_i = yplus_i / \sum_{i=1}^{n} yplus_i$	% Find weights, xplus, of all assets n %
$xminus_i = yminus_i / \sum_{i=1}^{n} yminus_i$	% Find weights, xminus, of all assets n%
xvalue = **objectvalue** (x, λ, retasset, dev, fitvalue)	% Calculate objective value of portfolio x and denote as xvalue. %
xplusvalue = **objectvalue** (xplus, λ, retasset, dev, fitvalue)	% Calculate objective value of portfolio xplus and denote as xplusvalue%
xminusvalue = **objectvalue** (xminus, λ, retasset, dev, fitvalue)	%Calculate objective value of portfolio xminus and denote as xminusvalue. %
if xplusvalue>xvalue **then**y=yplus	% Return yplus if it is better than y. %
end if	
if xminusvalue>xvalue **then** y=yminus	% Return yminus if it is better than y. %
end if	
return y	
End.	

Table 3: Pseudocode for calculating objective function value

Function Objectvalue (x, λ, retasset, dev, fitvalue)	
Begin	%Calculate effective expected return of portfolio%
retpor t= scalar/dot product (retasset, x)	% Calculate effective risk/variance of portfolio %
risk = x*dev*x'	%Calculate objective/objective value according to equation (1) above. %
fitvalue = λ*retport – (1 - λ)*risk	
return fitvalue	
End	

The function to search for better neighbouring solution (Table 2) requires calculating of the objective values of the candidate solutions. Table 3 shows the function for calculating the objective/fitness value of the candidate solutions, from equation (1). It is used to measure the quality of a portfolio.

RESULTS AND DISCUSSIONS

Efficient frontier

For every level of return, there is one portfolio that has the lowest possible risk and for every level of risk there is a portfolio that offers the highest return. This combination when plotted on a graph of the curve/line is known as the efficient frontier (Markowitz, 1959). The portfolios of this combination of return, risk values, plotted on the efficient frontier make up the set of efficient portfolios (Markowitz, 1959). The standard Markowitz model, equations (1) to (5), was used to find an optimum portfolio of 230 assets. Figure 2 is a plot of return of a portfolio (ret of porti) versus risk of a portfolio (risk of porti).

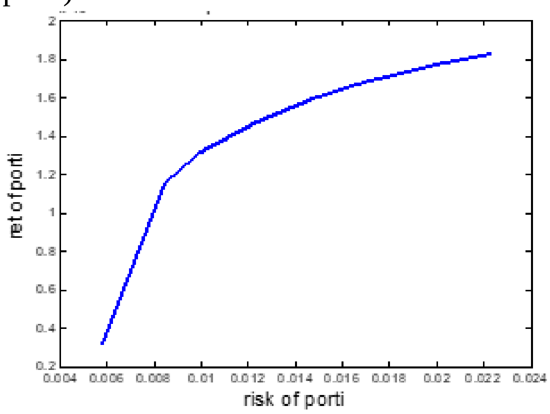

Figure 2: Efficient frontier of 230 assets portfolio using HC-S

The figure shows the efficient frontier obtained by tackling the Markowitz model using HC-S. It was applied on 230 assets portfolio. The 230 assets are from DAX stock exchange. The data used were daily returns over 1001 days.

Benchmarking HC-S

The algorithm is applied on a benchmark problem of solving standard Markowitz model as described in equations (1), (2), (3) under basic constraints (4) and (5). This problem has exact solution by standard methods. The standard method used for comparison is Quadratic Programming. Then the results from the algorithm proposed were compared with the results by Threshold Accepting. This is a well-established heuristic algorithm in portfolio optimization. The effectiveness, efficiency and reliability of the algorithm are further analysed. The assets and their return data used for applications in the algorithm are from DAX stock exchange. The data used were daily returns over 1001 days.

Benchmarking HC-S with Quadratic Programming method

Here HC-S was benchmarked on the Markowitz model and was tested on 10 assets portfolio. The results are compared with Quadratic Programming (QP) method, which is a standard method. Table 4 shows the experimental results obtained on benchmarking HC-S with QP. They are the percentage values in a table and corresponding bar charts of the weights of 10 assets portfolio. They were obtained by finding minimum variance portfolio (Markowitz model with $\lambda = 0$ in expression (1)) by quadratic programming method and by the hill-climbing algorithms HC-S. Quadratic programming (QP) produces exact solution so results by HC-S are compaired with QP results to see how accurate the method is. The values show the relative weights (of total bugdet) to be invested in each asset. The results (weights) by algorithm HC-S is from the best solution after 100 runs/iterations. Figure 3 shows the results of the algorithms HC-S in comparison to QP. The blue bars are that of Quadratic Programming (QP) and the red ones are of HC-S respectively.

Table 4: Experimental results on benchmarking HC-S with Quadratic Programming

algorithm	**Weight in each asset**									
QP	0.0053	0.0802	0.1150	0.3191	0.1622	0.0599	0.0419	0.0067	0.0356	0.1741
HC-S	0.0053	0.0801	0.1150	0.3193	0.1620	0.0601	0.0419	0.0067	0.0357	0.1739

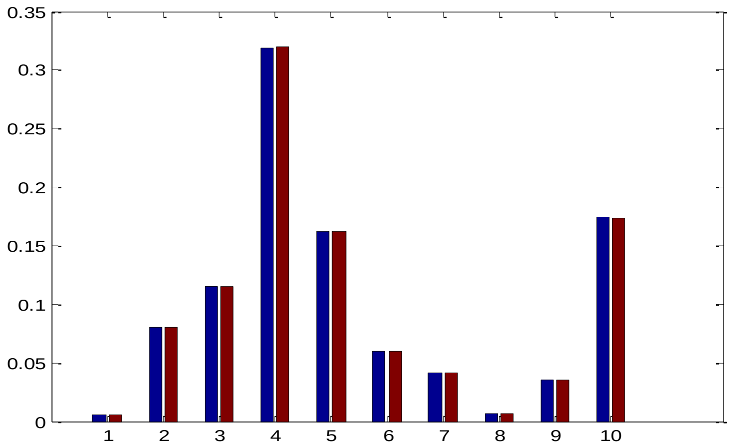

Figure 3: Comparison of HC-S (red-bars) with Quadratic Programming (blue-bars)

Table 4 and Figure 3 shows that solutions obtained by HC-S do not differ much from the exact solution by quadratic programming (QP). Variance/risk was calculated from the weights obtained by the methods QP and HC-S. The two methods attained the same low portfolio risk of 6.9751e-005. Attaining the same value of risk as QP depicts that the algorithm HC-S attains very accurate solutions. The similar height bars of HC-S compared to QP also depict that the algorithm HC-S give very accurate solutions.

Benchmarking the Algorithm using Threshold Accepting

HC-S was benchmarked on the Markowitz model, equation (1). They are tested on 100 assets portfolio. The results are compared with Threshold Accepting, which is a well-established Hill Climbing algorithm in portfolio selection and optimization. The assets and their return data used for applications in the algorithms are from DAX stock exchange. The data used were daily returns over 1001 days. The following is the algorithm that is evaluated.

- HC-S: Hill Climbing-Simple
- HC-S (9e+5): HC-S with 9e+5 iterations.

Table 5 shows the experimental results on the portfolio optimization on 100 stocks from DAX stock exchange; taken after 100 runs. The results show the values of objective function, number of functional evaluations required to reach final objective value, and average time in seconds for one run to converge to local maximum (final solution). The Best Final Objective value is the highest objective function value obtained in all 100 runs. Final objective values obtained in each run were recorded. Table 5 is the Mean, STD and Worst of Final objective values in all the 100 runs. The Mean and STD of Number of functional evaluations to reach final objective value, of the 100 runs, are also given.

Table 5: Experimental results on Portfolio optimization on 100 stocks, after 100 runs

Algorithm Iterations		**HC-S** (9e+5)	**Threshold Accepting** (9e+5)
Best Final Objective value		0.000596	0.000588
Final objective value	Mean STD Worst	0.000594 6.46e-6 0.000559	0.000563 3.46e-5 7.2563e-5
No. of functional evaluations to final objective value	Mean STD	2.7e+5 6800	3.0e+5 1770
Average time for 1 run (in sec.)		39.0	704.7

STD =Standard Deviation

Discussion

HC-S managed to attain higher best final objective value (0.000596) than Threshold Accepting (0.000588). The best final objective values are higher in HC-S showing that the method is more robust than Threshold Accepting as HC-S better escape local optima. To understand the significance of the difference in final objective value we look at the best final objective value of HC-S which is 0.000596. This translates to a return of 0.14% and a risk of 1.34% one day after investment, of the 100 stocks considered. The best final objective value of Threshold Accepting, 0.000588, translates to a return of 0.13% and a risk of 1.54% one day after investment. The following days could include compounded interest on the original capital. From the return and risk values, it is observed that you incur more risk but expect less return when you use the Threshold Accepting rather than HC-S to find an optimal portfolio.

The mean of final objective value of HC-S is higher (0.000594) than that of Threshold Accepting (0.000563). The worst final objective of HC-S is a lot better (0.000559) than that of Threshold Accepting (7.2563e-5). The STD of mean of final objective value of HC-S (6.46e-6) is 10 times less that of Threshold Accepting (3.46e-5). The number of functional evaluations for HC-S was 2.7e5 while that of Threshold Accepting was 3.0e5. HC-S was faster as it required less number of functional evaluations. The STD of the number of functional evaluations of HC-S (6800) is more than that of Threshold Accepting (1770).

Considering the time in seconds for one run to converge to best final objective value, Threshold Accepting (704.7), required more time than HC-S (39.0). This shows that it is far more expensive (time wise) to compute neighbourhood function of Threshold Accepting than that of HC-S.

A t-test was performed on final objective values and on the number of functional evaluations to final objective of the 100 runs. Both outcomes displayed a rejection of the null hypothesis at the 5% (default value) significance level. The t-test was performed using Mat-lab (R2010a).

Furthermore, to use Threshold Accepting, one has to first calculate and sort threshold sequences according to a certain problem. These are the sequences by which poor solutions will be accepted to avoid being trapped in a local optimum. The process makes Threshold Accepting quite cumbersome. HC-S was tested on the Markowitz model; in finding weights for 100 stocks in portfolio optimization, where a budget and return constraints are imposed. Results demonstrate that HC-S manages to find significantly better solutions than Threshold Accepting, an established algorithm for portfolio optimization. Table 6 summarises the benchmarking results of the algorithm.

Table 6: Benchmarking HC-S with T.A.

Algorithm	Effectiveness	Efficiency
T.A.	Well established algorithm in portfolio optimization	Efficient
HC-S	More effective in finding better solution than **T.A**	More efficient and quite faster than **T.A**

CONCLUSIONS AND RECOMMENDATION

A heuristic algorithm (HC-S) has been proposed, its effectiveness and efficiency for the portfolio optimization problem demonstrated. The algorithm produced attained promising results for portfolio optimization. HC-S was used to tackle the portfolio optimization problem, of the standard Markowitz model, where a budget constraint is imposed and no short-selling is permitted. HC-S. Benchmarking with Quadratic programming (QP) showed that HC-S attains accurate solutions. Also, HC-S has been demonstrated to be more effective and efficient than Threshold Accepting (TA), an established algorithm for portfolio optimization since HC-S find solutions with significantly higher objective value and require less computing time. It is recommended that, realistic, non-linear constraints like cardinality, maximum holding size, and minimum holding size, transaction costs, and regulations should be incorporated in the proposed hill climbing algorithm to solve the extended portfolio optimization problem.

REFERENCE

Aranha C. C., Iba, H. (2009) Using memetic algorithms to improve portfolio performance in static and dynamic trading scenarios. Proceedings of the 11th Annual Conference on Genetic and Evolutionary Computation, GECCO 2009, pp. 1427–1434. ACM, New York

Aranha, C.C., Iba, H (2008). A tree-based ga representation for the portfolio optimization problem. Proceedings of the 10th Annual Conference on Genetic and Evolutionary Computation, GECCO 2008, pp. 873–880. ACM, New York

Arriaga, J., Valenzuela-Rendón, M. (2012). Steepest Ascent Hill Climbing for Portfolio Selection. In:, *et al.* Applications of Evolutionary Computation. EvoApplications 2012. Lecture Notes in Computer Science, vol 7248. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-29178-4_15

Crama Y. and Schyns M. (2003). Simulated annealing for complex portfolio selection problems. European Journal of Operational Research, 150: 546–571.

Di Tollo G. and Roli A. (2008). Metaheuristics for the portfolio selection problem. International Journal of Operation Research, 5(1): 130-135.

Doering J. Kizys R., Juan A. A, Fitó A., Polat O., (2019) Metaheuristics for rich portfolio optimization and risk management: Current state and future trends. Operations Research Perspectives, Volume 6, 100121. ISSN 2214-7160, https://doi.org/10.1016/j.orp.2019.100121

Dueck G. and Scheuer T. (1990). Threshold accepting: A general purpose optimization algorithm. Journal of Computational Physics; 90(1):161–175. https://doi.org/10.1016/0021-9991(90)90201-B

Dueck G. and Winker P. (1992). New concepts and algorithms for portfolio choice. Applied stochastic models and data analysis, 8: 159-178. https://doi.org/10.1002/asm.3150080306

Gilli M. and Këllezi E. (2000). Heuristic Approaches for Portfolio Optimization, 6th Intl. Conf. on Computing and Economics, July, Barcelona, Spain.

Gilli M. and Schumann E. (2010). Portfolio Optimization with Threshold Accepting: A Practical Guide, Ch. 9. In Stephen E. Satchell (Ed), Optimizing Optimization*:* The Next Generation of Optimization Applications and Theory, Elsevier, 201-223. https://doi.org/10.1016/B978-0-12-374952-9.00009-9

Gilli M. and Schumann E. (2012). Heuristic Optimisation in Financial Modelling. Annals of Operations Research, 193(1): 129-158. https://doi.org/10.1007/s10479-011-0862-y

Gilli M. and Winker P. (2008). A Review of heuristic optimization methods in econometrics, Swiss Finance Institute, Research Paper No. 08-12. Available online at SSRN: http://ssrn.com/abstract=1140655, Accessed on 20th May 2021.

Kirkpatrick S., Gelatt Jr. C.D. and Vecchi M.P. (1983). Optimization by simulated annealing, Science, 220(4598): 671–680.

Kalayc C. B., Polat O. Akbay A.M, (2020). An efficient hybrid metaheuristic algorithm for cardinality constrained portfolio optimization, Swarm and Evolutionary Computation, Volume 54, 100662, ISSN 2210-6502, https://doi.org/10.1016/j.swevo.2020.100662.

Lwin, K., Qu, R. (2013). A hybrid algorithm for constrained portfolio selection problems. *A*ppl Intell 39**,** 251–266 https://doi.org/10.1007/s10489-012-0411-7

Maringer D. (2008). Heuristic Optimization for Portfolio management. Nov, IEEE Computational Intelligence Magazine**,** 3(4):31–34, doi:10.1109/MCI.2008.929847

Markowitz H. (1952). Portfolio selection. The Journal of Finance, 7(1): 77-91.

Markowitz H. (1959). Portfolio selection: Efficient diversification of investments. John Wiley and Sons, New York, USA.

Mercangöz B. A. and Eroglu E. (2021) The Genetic Algorithm: An Application on Portfolio Optimization In MehdiKhosrow-Pou (ed): Research Anthology on Multi-Industry Uses of Genetic Programming and Algorithms Pages: 21. IGI Global |

Milhomem D. A. and Dantas M. J. P. (2020) Analysis of new approaches used in portfolio optimization: a systematic literature review. Systematic Review Prod. 30 Scientific Electronic Library Online https://doi.org/10.1590/0103-6513.20190144

Sharpe W.F. (2000). Portfolio theory and capital markets. McGraw-Hill, USA. ISBN-13: 978-0071353205

Schaerf, A. (2002) Local Search Techniques for Constrained Portfolio Selection Problems. Computational Economics 20, 177–190. https://doi.org/10.1023/A:1020920706534

Suraj S. M. Manoj T. (2018). Multi-objective heuristic algorithms for practical portfolio optimization and rebalancing with transaction cost. Applied Soft Computing. Volume 67, 865-894

Silva Y. L. T.V Herthel A. B, Subramanian A. (2019). A multi-objective evolutionary algorithm for a class of mean-variance portfolio selection problems, Expert Systems with Applications, Volume 133, Pages 225-241, ISSN 0957-4174, https://doi.org/10.1016/j.eswa.2019.05.01.

Winker P. (2001). Optimization Heuristics in Econometrics- Application of Threshold Accepting. John Wiley and Sons, New York, USA.

Winker P. and Maringer D. (2007). The Threshold Accepting Optimization Algorithm in Economics and Statistics. In: Kontoghiorges E.J. and Gatu C. (Eds.), Optimisation, Econometric and Financial Analysis. Advances in Computational Management Science, Springer, 9: 107-125. Springer, Berlin, Heidelberg. https://doi.org/10.1007/3-540-36626-1_6