Novel Multiobjective Evolutionary Algorithm Approaches with application in the Constrained Portfolio Optimization

A dissertation submitted in partial fulfillment of the requirements
for the degree of

Doctor of Philosophy

in the Department of Informatics
in the School of Information and Communication Technologies
at the University of Piraeus

by

Konstantinos Liagkouras

Under the Supervision of
Associate Professor
Konstantinos Metaxiotis

Department of Informatics
School of Information and Communication Technologies
University of Piraeus
November 2015

Novel Multiobjective Evolutionary Algorithm Approaches with application in the Constrained Portfolio Optimization

A dissertation submitted in partial fulfillment of the requirements
for the degree of

Doctor of Philosophy

in the Department of Informatics
in the School of Information and Communication Technologies
at the University of Piraeus

by

Konstantinos Liagkouras

Advisory Committee: Konstantinos Metaxiotis, Associate Professor, Univ. of Piraeus
Dimitrios Despotis, Professor, Univ. of Piraeus
Dimitrios Apostolou Assistant Professor, Univ. of Piraeus

APPROVED BY

Date: …………………………………………………..

………………………….
Konstantinos Metaxiotis
Associate Professor
Univ. of Piraeus

…………………………..
Dimitrios Despotis
Professor
Univ. of Piraeus

………………………..
Dimitrios Apostolou
Assistant Professor
Univ. of Piraeus

………………………..
George Tsihrintzis
Professor
Univ. of Piraeus

………………………..
Siskos Ioannis
Professor
Univ. of Piraeus

………………………..
Dimitrios Askounis
Professor
NTUA

………………………..
Konstantinos Nikolopoulos
Professor
Bangor University

Abstract

Multiobjective optimization (MO) is the problem of simultaneously optimizing two or more conflicting objectives subject to certain constraints. Many real-world problems involve simultaneous optimization of several often conflicting objectives. The portfolio optimization problem belongs to this category of problems. According to Markowitz's Mean - Variance model (MV) an investor attempts to maximize portfolio expected return for a given amount of portfolio risk or minimize portfolio risk for a given level of expected return.

The portfolio optimization problem involves two conflicting objectives (i.e. expected return and portfolio risk) and thus belongs to the family of multiobjective problems. With the assistance of scalarization techniques a multiple objective problem can be converted into a single objective problem. However, the drawbacks to these conventional approaches lead to the development of alternative techniques that yield a set of Pareto optimal solutions rather than only a single solution.

The problem becomes much more complicated when we incorporate to the portfolio model some real world constraints. These additional constraints made the portfolio optimization problem difficult to be solved with exact methods. In the last decade several metaheuristic optimization techniques have been developed to address the challenges imposed by complex multiobjective optimization problems. Due to the intrinsic multiobjective nature of the portfolio optimization problem, multiobjective approaches, particularly multiobjective evolutionary algorithms (MOEAs) are suitable in handling the difficulties imposed by this type of problems.

Especially in the presence of multiple constraints the portfolio optimization problem becomes very complicate and efficient solution needs to be found. Furthermore, the existing multiobjective evolutionary algorithms (MOEAs) techniques cannot be used directly to solve the constrained portfolio optimization problem as a number of configuration issues related to the application of MOEAs for solving the constrained portfolio optimization problem must be addressed. The successful implementation of the constrained portfolio optimization problem by the MOEAs requires the development of novel algorithmic and technical approaches. In particular new multiobjective evolutionary approaches are needed to efficiently solve the constrained portfolio optimization problem.

In this thesis we address these issues by examining a number of configuration issues related to the application of MOEAs for solving the constrained portfolio optimization problem. Furthermore we introduce a new multiobjective evolutionary algorithm (MOEA) that incorporates a novel representation scheme and specially designed genetic operators for the solution of the constrained portfolio optimization problem. These issues have been addressed in this thesis and a set of efficient solutions is found for each of the examined test problems.

In this thesis we develop a methodological framework for conducting a comprehensive literature study based on the papers published in MOEAs for the Portfolio Management over a long time span across various disciplines. This framework is being used to gain an understanding of the current state of the MOEAs for the Portfolio Management research field. Based on the literature study, we identify potential areas of concern in regard to MOEAs for the Portfolio Management. Based on the examination of the state-of-the art we present the best practices from a technical and algorithmic point of view for dealing with the complexities of the constrained portfolio optimization problem. We introduce new genetic operators to enhance algorithms' performance. We propose a novel representation scheme for the solution of the constrained portfolio optimization problem. Finally, we introduce a novel MOEA for the solution of the constrained portfolio optimization problem. The experimental results applied to the constrained portfolio optimization problem, indicate that the proposed approach generates solutions that lie on the true efficient frontier (TEF) for all of the examined cases for a fraction of time required by exact approaches.

Keywords: **Multiobjective optimization, evolutionary algorithms, genetic operators, portfolio optimization, efficient frontier, cardinality constraint, encoding scheme.**

Acknowledgements

Most of the work in this thesis would not have been possible without the help, the support and the ideas suggested by collegues, professors and friends. My first great thank must be given to Associate Professor Konstantinos Metaxiotis, who supervised my activity providing me with the methodology and knowledge he thought to be necessary for my research and, on several occasions, brought me back on the proper route where appropriate. Without him, the outcome of my research would not be available at the moment.

I would also like to acknowledge the help provided by Professor Dimitris Despotis and Assistant Professor Dimitris Apostolou for their support and for the research discussions we had together during my studies as a PhD candidate at the University of Piraeus.

I want furthermore to thank my lab co-workers: Dimitris Sotiros, Grigoris Koronatos and Aggeliki Mikeli for the time we shared together.

Many thanks to my family, for unselfishly bearing with the time away from them to pursue this life goal over the last five years.

Konstantinos Liagkouras,

Piraeus, November 2015

Contents

Contents

Contents

Contents

Contents

Contents

Contents

Contents

List of Figures

List of Figures

List of Figures

List of Figures

List of Figures

List of Tables

List of Tables

List of Tables

List of Tables

List of Tables

List of Tables

Chapter 1

Introduction

Multiobjective optimization (MO) is the problem of simultaneously optimizing two or more conflicting objectives subject to certain constraints. Many real-world problems involve simultaneous optimization of several often conflicting objectives. The portfolio optimization problem is a characteristic example of this category of problems. According to Markowitz's [151] Mean - Variance theory (MV) an investor attempts to maximize portfolio expected return for a given amount of portfolio risk or minimize portfolio risk for a given level of expected return. The MV theory was criticized for unrealistic assumptions. Responding to the critics, researchers incorporated to the portfolio model some real world constraints like bounds on holdings, cardinality, minimum transaction lots and sector capitalization constraints.

The turn toward evolutionary approaches for handling the difficulties introduced by the constrained portfolio optimization problem can be easily justified by a number of reasons. Among them, distinguished place possesses the ability of evolutionary algorithms (EAs) to solve difficult multiobjective optimization problems (MOP) that are too complex to be solved using deterministic techniques [7], [33], [43], [95], [96], [109], [153], [177], [190], [194], [207], [214]. The introduction of realistic constraints into the portfolio optimization problem converts the problem from a classical quadratic optimization problem to a quadratic mixed-integer problem (QMIP) that is NP-hard [159].

Thanks to Multiobjective Evolutionary Algorithms (MOEAs) techniques the classical portfolio model can be extended to handle two or more conflicting objectives subject to various realistic constraints.

Because the various objectives functions in the portfolio selection problem are usually in conflict with each other, each time that we attempt to optimize further an objective other objectives suffer as a result. Therefore, the objective in MOEAs is to find the Pareto front of efficient solutions that provide a tradeoff between the various objectives.

1.1 Motivation and Objectives of this Research

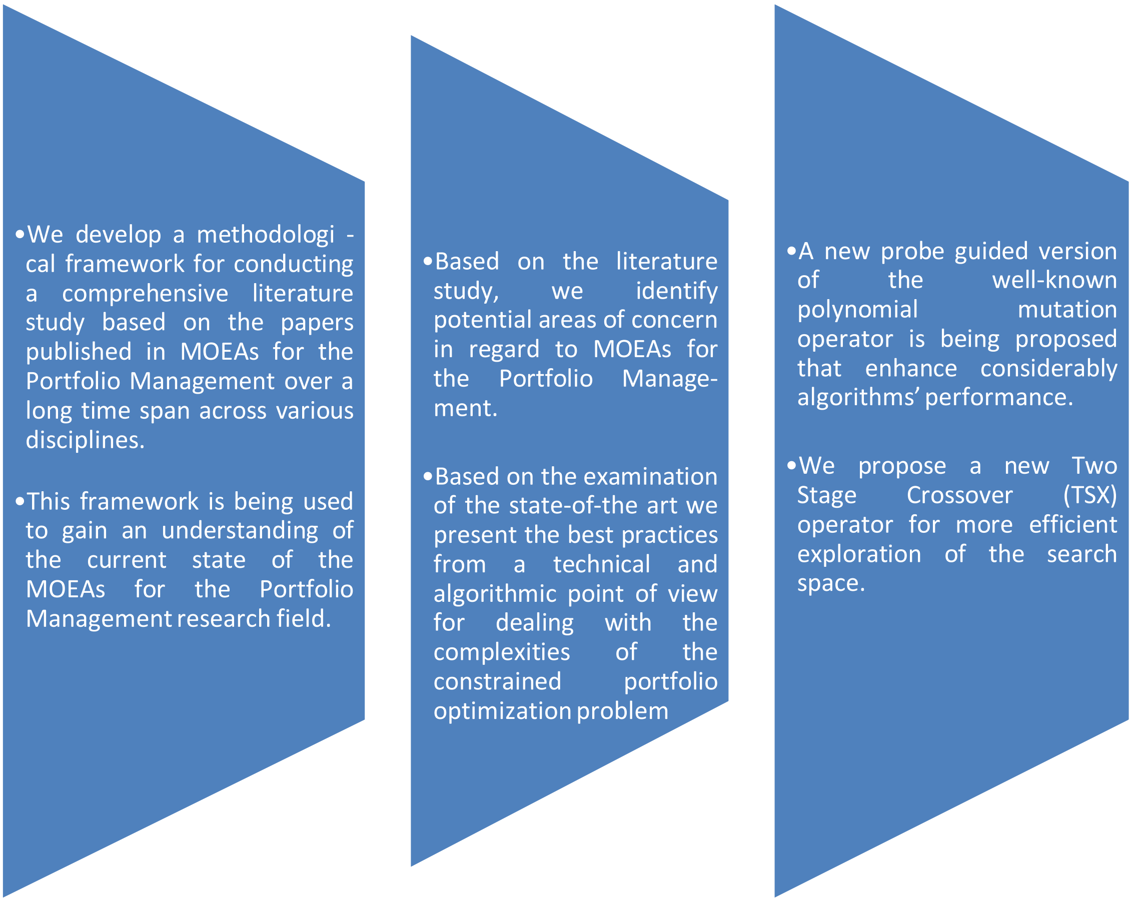
The portfolio optimization problem belongs to the family of multiobjective problems. Especially in the presence of multiple constraints the portfolio optimization problem becomes very complicate and efficient solution needs to be found. Furthermore, the existing multiobjective evolutionary algorithms (MOEAs) techniques cannot be used directly to solve the constrained portfolio optimization problem as a number of configuration issues related to the application of MOEAs for solving the constrained portfolio optimization problem must be addressed. The successful implementation of the constrained portfolio optimization problem by the MOEAs requires the development of novel algorithmic and technical approaches. In particular new multiobjective evolutionary approaches are needed to efficiently solve the constrained portfolio optimization problem.

The purpose of this PhD dissertation is fivefold, namely:

- To present the current state of research in portfolio optimization with the support of MOEAs.
- To address a number of configuration issues related to the application of MOEAs for solving the constrained portfolio optimization problem
- To introduce new genetic operators to enhance algorithms' performance.
- To introduce a novel representation scheme for the solution of the constrained portfolio optimization problem.
- To introduce a novel MOEA for the solution of the constrained portfolio optimization problem.

1.2 Contributions of this Research

The contributions of this research are summarized in the following:



- We develop a methodological framework for conducting a comprehensive literature study based on the papers published in MOEAs for the Portfolio Management over a long time span across various disciplines.
- This framework is being used to gain an understanding of the current state of the MOEAs for the Portfolio Management research field.

- Based on the literature study, we identify potential areas of concern in regard to MOEAs for the Portfolio Management.
- Based on the examination of the state-of-the art we present the best practices from a technical and algorithmic point of view for dealing with the complexities of the constrained portfolio optimization problem

- A new probe guided version of the well-known polynomial mutation operator is being proposed that enhance considerably algorithms' performance.
- We propose a new Two Stage Crossover (TSX) operator for more efficient exploration of the search space.

This research introduces a new multiobjective evolutionary algorithm (MOEA), called Efficiently Encoded Multiobjective Portfolio Optimization Solver (EEMPOS) that incorporates an efficient encoding scheme and specially designed genetic operators for the solution of the constrained portfolio optimization.

1.3 Organization of this Dissertation

The remainder of this dissertation is organized as follows:

Chapter 2 Briefly reviews the most well-known MOEAs alongside with their strengths and weaknesses.

Chapter 3 Briefly reviews the existing literature in the area of portfolio optimization with the support of MOEAs.

Chapter 4 Presents a methodological framework for conducting a comprehensive literature study based on the papers published in MOEAs for the Portfolio Management over a long time span. This framework is being used to gain an understanding of the current state of the MOEAs for the Portfolio Management research field.

Chapter 5 Based on the examination of the state-of-the art, presents the best practices from a technical and algorithmic point of view for dealing with the complexities of the constrained portfolio optimization problem.

Chapter 6 Proposes a new probe guided version of the polynomial mutation (PLM) operator. The experimental results reveal that the proposed Probe Guided Mutation (PGM) operator outperforms with confidence the performance of the classical PLM operator for all performance metrics when applied to the solution of the cardinality constrained portfolio optimization problem (CCPOP), but also to Zitzler-Deb-Theile (ZDT) [229] and Deb-Theile-Laumanns- Zitzler (DTLZ) [60] families of test functions.

Chapter 7 Proposes a new Two Stage Crossover (TSX) operator for more efficient exploration of the search space. The performance of the proposed TSX operator is assessed in comparison with the Simulated Binary Crossover (SBX) operator with the assistance of three well-known MOEAs, namely the NSGAII, the SPEA2 and the MOCELL, for the solution of the DTLZ1-7 set of test functions [60]. We also compare the proposed TSX operator with other popular reproduction operators like the Differential Evolution (DE) and the Particle Swarm Optimization (PSO). It is shown with the assistance of the DTLZ set of test functions that the TSX operator can substantially improve the results generated by three popular performance metrics for most of the cases.

Chapter 8 Introduces a new multiobjective evolutionary algorithm (MOEA), called Efficiently Encoded Multiobjective Portfolio Optimization Solver (EEMPOS) that incorporates an efficient encoding scheme and specially designed genetic operators for the solution of the constrained portfolio optimization problem. In order to evaluate the performance of the proposed EEMPOS, we calculate the True Efficient Frontier (TEF) by formulating the constrained portfolio optimization problem as a Mixed Integer Quadratic Program (MIQP) and we compare the relevant results with the approximate efficient frontiers that are derived by the proposed EEMPOS. The relevant results verify that the EEMPOS generates solutions that lie on the TEF for a fraction of the time required by the MIQP.

Chapter 9 This dissertation is concluded with a summary of the research and proposed directions for future study.

1.4 Publications based on this Thesis

The research undergone during the preparation of this thesis resulted in the following list of publications.

International journal papers

1. Metaxiotis, K. & Liagkouras, K. (2012) Multiobjective Evolutionary Algorithms for Portfolio Management: A comprehensive literature review. Expert Systems with Applications, Elsevier, 39 (14): 11685-1169.

2. Liagkouras K. & Metaxiotis K. (2014) A new Probe Guided Mutation Operator and its application for solving the Cardinality Constrained Portfolio Optimization Problem. Expert Systems with Applications, Elsevier, 41 (14), 6274-6290.

3. Liagkouras K. & Metaxiotis K. (2015) Efficient Portfolio Construction with the Use of Multiobjective Evolutionary Algorithms: Best Practices and Performance Metrics, International Journal of Information Technology & Decision Making, 14 (03), 535-564, World Scientific.

4. Liagkouras K. & Metaxiotis K. (2015) An experimental analysis of a new two-stage crossover operator for multiobjective optimization, Soft Computing, pp. 1-31, Springer DOI: 10.1007/s00500-015-1810-6.

5. Liagkouras K. & Metaxiotis K. (2015) An Experimental Analysis of a New Interval-Based Mutation Operator, International Journal of Computational Intelligence and Applications 14 (3) 1550018, World Scientific.

6. Liagkouras K. & Metaxiotis K. (2015) A new Probe Guided Mutation Operator for more Efficient Exploration of the search space: An Experimental Analysis, International Journal of Operational Research, vol. 25, No. 2, 2016, Inderscience.

7. Liagkouras K. & Metaxiotis K. (2015) Enhancing the performance of MOEAs: An experimental presentation of a new Fitness Guided Mutation Operator, Journal of Experimental & Theoretical Artificial Intelligence, Taylor & Francis, DOI: 10.1080/0952813X.2015.1132260.

8. Liagkouras K. & Metaxiotis K. (2015) Examining the Effect of different Configuration Issues of the Multiobjective Evolutionary Algorithms on the Efficient Frontier Formulation for the Constrained Portfolio Optimization Problem, Journal of the Operational Research Society, Palgrave, Accepted publication pending.

9. Liagkouras K. & Metaxiotis K. (2015) A new Efficiently Encoded Multiobjective Algorithm for the Solution of the Constrained Portfolio Optimization Problem, Under review, European Journal of Operational Research.

Book chapters

1. Liagkouras K. & Metaxiotis K. (2015) A new Two Stage Crossover Operator and its application for solving the Cardinality Constrained Portfolio Selection Problem, "Multiple Criteria Decision Making in Finance, Insurance and Investment", Springer, edited by Minwir Al-Shammari and Hatem Masri DOI 10.1007/978-3-319-21158-9_8.

2. Liagkouras K. & Metaxiotis K. (2015) A Probe Guided Crossover operator for more Efficient Exploration of the Search Space, Studies in Computational Intelligence, Vol. 627, George A. Tsihrintzis et al. (Eds): Intelligent Computing Systems, Springer.

International conference papers

1. Metaxiotis, K., Liagkouras, K. (2015) The solution of the 0-1 multi-objective knapsack problem with the assistance of multi-objective evolutionary algorithms based on decomposition: A comparative study, 2015 5th International Workshop on Computer Science and Engineering: Information Processing and Control Engineering, WCSE 2015-IPCE.

2. Liagkouras, K. & Metaxiotis, K. (2015) Advanced Technologies and Algorithms for Efficient Portfolio Selection, International Journal of Social, Behavioral, Educational, Economic and Management Engineering Vol:9, No:7, 2015, World Academy of Science, Engineering and Technology

3. Metaxiotis, K. & Liagkouras, K. (2014) An exploratory crossover operator for improving the performance of MOEAs, Advances in Applied and Pure Mathematics, pp. 158-162, ISBN: 978-1-61804-240-8.

4. Liagkouras, K. & Metaxiotis, K. (2014) Application of Customer Relationship Management Systems in Business: Challenges and Opportunities, International Journal of Social, Education, Economics and Management Engineering Vol:8, No:6, 2014, World Academy of Science, Engineering and Technology.

5. Metaxiotis, K. & Liagkouras, K. (2014) An Expert System Designed to Be Used with MOEAs for Efficient Portfolio Selection, International Journal of Computer, Information Science and Engineering Vol:8 No:2, 2014, World Academy of Science, Engineering and Technology.

6. Metaxiotis, K & Liagkouras, K., (2013) A fitness guided mutation operator for improved performance of MOEAs Electronics, Circuits, and Systems (ICECS), 2013 IEEE 20th International Conference on, pp. 751-754, DOI: 10.1109/ICECS.2013.6815523.

7. Liagkouras, K. & Metaxiotis, K. (2013) The Constrained Mean-Semivariance Portfolio Optimization Problem with the Support of a Novel Multiobjective Evolutionary Algorithm, Journal of Software Engineering and Applications, 2013, 6, 22-29 doi:10.4236/jsea.2013.67B005.

8. Liagkouras, K. & Metaxiotis, K. (2013) An Elitist Polynomial Mutation Operator for Improved Performance of MOEAs in Computer Networks, pp. 1-5, Computer Communications and Networks (ICCCN), 2013 22nd International Conference on, DOI: 10.1109/ICCCN.2013.6614105.

9. Liagkouras, K. & Metaxiotis, K. (2012) Modern Portfolio Management with the Support of Multiobjective Evolutionary Algorithms: An Analysis of Problem Formulation Issues, World Academy of Science, Engineering and Technology 69 2012, Paris, France 27 – 28 June 2012.

In total, eleven (11) papers published in international journals and books and nine (9) papers in international conferences with more than 35 citations in scopus.

Chapter 2

Multiobjective Evolutionary Algorithms (MOEAs): A brief review

The idea of using techniques based on the emulation of the mechanism of natural selection to solve problems can be traced as long back as the 1930s (Cannon [25]). Since 1930s, this research field did not see major developments for almost three decades. However, in 1960s three studies (Holland [115]; Schwefel [185]; Fogel [91]) set the foundations of what is nowadays denominated "Evolutionary Algorithms" (EAs).

EAs are population based stochastic optimization heuristics inspired by Darwin's Evolution Theory. An EA searches through a solution space in parallel by evaluating a set (population) of possible solutions (individuals). An EA starts with a random initial population. Then the 'fitness' of each individual is determined by evaluating the objective function. After the best individuals are selected, new individuals for the next generation are created. The new individuals are generated by altering the individuals through random mutation and by mixing the decision variables of multiple parents through crossover. Then the generational cycle repeats until a breaking criterion is fulfilled. EAs have been applied successfully to a wide range of problems such as engineering, biology, genetics, finance etc. According to the literature EAs have been proved very effective for single–optimization (Goldberg [100]; Schwefel [186]; Fogel [92]).

The last years there has been an increasing interest to explore the application of evolutionary algorithms for the solution of multiobjective optimization problems (MOPs).

Multiobjective optimization (MO) is the problem of maximizing / minimizing a set of conflicting objectives functions subject to a set of constraints. In Multiobjective optimization there is not a single solution that maximizes / minimizes each objective to its fullest. This happens because the various objectives functions in the problem are usually in conflict with each other. Therefore, the objective in MO is to find the Pareto front of efficient solutions that provide a tradeoff between the various objectives. EAs are naturally suited for extension into the multiobjective optimization problems domain, thanks to the population based search strategy and the simple selection strategy. The ability of EAs to deal simultaneously with a set of possible solutions (population) allows to find several members of the Pareto optimal set in a single run of the algorithm, instead of having to perform a series of separate runs as in the case of the traditional mathematical programming techniques. Moreover, EAs are less susceptible to the shape or continuity of the Pareto front (they can easily deal with discontinuous and concave Pareto fronts), whereas these two issues are known problems with mathematical programming techniques. During multiobjective optimization two goals are to be reached. On the one hand the solutions should be as close to the global Pareto-optimal front as possible and on the other hand the solutions should also cover the whole Pareto-front. The first goal is often achieved through elitism by replacing random individuals with individuals on the Pareto front. The second goal can be achieved by punishing individuals that are too close together (Fitness Sharing).

According to Coello Coello et al. [40], [41], [42], the traditional evolutionary algorithms cannot efficiently deal with multi-objective optimization problems for two (2) reasons:

1. Due to stochastic noise, evolutionary algorithms tend to converge to a single solution if run for a sufficiently large number of iterations. Thus, it is necessary to block the selection mechanism so that different solutions (non-dominated) are preserved in the population of an evolutionary algorithm.
2. All non-dominated solutions should be sampled at the same rate during the selection stage, since all non-dominated solutions are equally good among themselves.

Over the past years researchers developed several approaches for the solution of multi-objective optimization problems with the use of EAs. The first implementation of a multi-objective evolutionary algorithm (MOEA) dates back to the mid-1980s (Schaffer [179], [180]). Since then, a considerable amount of research has been done in this area, now known as evolutionary multiobjective optimization. A brief review of the most well-known MOEAs is presented below:

2.1 Vector Evaluation Genetic Algorithm (VEGA)

Schaffer [179], [180] and Schaffer & Grefenstette [181] introduce the Vector Evaluation Genetic Algorithm (VEGA) in the mid 1980s, which was the first implementation of a multi-objective evolutionary algorithm (MOEA). In reality VEGA was not nothing more than a simple genetic algorithm with a modified selection mechanism. A weakness of VEGA approach according to Schaffer is that the generated solutions are non-dominated in a local sense, because their non-dominance is limited to the current population. Schaffer highlighted another problem that in genetics is known as "speciation" which is the evolution of "species" within the population which excel in one dimension of performance, without considering at the other dimensions. Thus VEGA approach tend to reject individuals with acceptable performance, perhaps above average, but not outstanding for any of the objective functions. Schaffer called this tendency of VEGA to reject individuals with acceptable performance for any objective function as "middling", and it is the most serious shortcoming of VEGA method.

2.2 Niched Pareto Genetic Algorithm (NPGA)

NPGA developed by Horn et al. [116]. NPGA extends the traditional GA to multiple objectives through the use of Pareto domination ranking and fitness sharing (or niching). The main advantages of this method is its adaptability to a wide range of multiobjective optimization problems and its ability to search non-linear and discontinuous search spaces without relying on the need for continuous first and second derivatives.

2.3 Niched Pareto Genetic Algorithm II (NPGA-II)

Erickson et al. [83] proposed the NPGA-II. This algorithm relies on the traditional Pareto ranking approach, but it keeps its tournament selection scheme. Ties are solved through fitness sharing as in its predecessor. However, the niche count of the NPGA-II is computed using individuals from the next partially filled generation using a technique called "continuously updated fitness sharing".

2.4 Nondominated Sorting Genetic Algorithm (NSGA)

Nondominated Sorting Genetic Algorithm (NSGA) proposed by Srinivas and Deb [196] is based on Goldberg [100]. Goldberg suggested a nondominated sorting procedure to overcome the weakness of VEGA to bias towards some Pareto-optimal solutions. Schaffer himself the creator of VEGA had realized VEGA's bias towards some Pareto-optimal solutions. NSGA eliminates the bias in VEGA and thereby distributes the population over the entire Pareto-optimal regions.

2.5 Nondominated Sorting Genetic Algorithm II (NSGA-II)

Deb et al. [57] propose the NSGA-II, a fast non-dominated sorting based multi-objective evolutionary algorithm with $O(MN^2)$ computational complexity. NSGA-II implements a selection operator that creates a mating pool by combining the parent and offspring populations and selecting the best, with respect to fitness and spread, N solutions.

2.6 Multi-Objective Genetic Algorithm (MOGA)

Fonseca and Fleming [93] proposed a multiobjective genetic algorithm (MOGA) in which the rank of a certain individual corresponds to the number of individuals in the current population by which it is dominated. Based on this scheme, all the nondominated individuals are assigned rank 1, whereas dominated ones are penalized according to the population density of the corresponding region of the tradeoff surface.

2.7 Strength Pareto Evolutionary Algorithm (SPEA)

Zitzler and Thiele [233] introduced this algorithm. SPEA integrates different MOEAs. SPEA uses an archive containing nondominated solutions previously found (the so-called external nondominated set). At each generation, nondominated individuals are copied to the external nondominated set. For each individual in this external set a strength value is computed. In SPEA the fitness of each member of the current population is computed according to the strengths of all external nondominated solutions that dominate it. The fitness assignment process of SPEA considers both closeness to the true Pareto front and even distribution of solutions at the same time. Thus, instead of using niches based on distance, Pareto dominance is used to ensure that the solutions are properly distributed along the Pareto front.

2.8 Strength Pareto Evolutionary Algorithm II (SPEA-II)

SPEA-II was introduced by Zitzler, Laumanns and Thiele [231]. The authors tried to develop a MOEA that eliminates the weaknesses of its predecessor (SPEA) and to incorporate most recent developments. The main differences of SPEA2 in comparison to SPEA are: 1. An improved fitness assignment scheme is used, which takes for each individual into account how many individuals it dominates and it is dominated by, 2. A nearest neighbor estimation technique is incorporated which allows a more precise guidance of the search process, 3. A new archive truncation method guarantees the preservation of boundary solutions.

2.9 Pareto Archived Evolution Strategy (PAES)

Knowles and Corne [125] introduced the PAES. The authors argue that PAES may represent the simplest possible non-trivial algorithm capable of generating diverse solutions in the Pareto optimal set. The algorithm in its simplest form, is a (1 + 1) evolution strategy (i.e. a single parent that generates a single offspring), employing local search but using a reference archive of previously found solutions in order to identify the approximate dominance ranking of the current and candidate solution vectors. PAES (1 + 1) is intended as a good baseline approach, against which more involved methods may be compared, and may also serve well in some real-world application when local search seems superior to or competitive with population-based methods. Moreover, they introduced $(1 + \lambda)$ and $(\mu + \lambda)$ variants of PAES as extensions to the basic algorithm.

2.10 Pareto Envelope - based Selection Algorithm (PESA)

Corne et al. [45] introduce a new multiobjective evolutionary algorithm called PESA, in which selection and diversity maintenance are controlled via a simple hyper - grid based scheme. PESA integrates the selection and diversity maintenance. They compare PESA with two other MOEAs (SPEA and PAES) on some multiobjective test problems proposed by Deb K. [51]. According to their study the PESA outperforms the other two MOEAs (SPEA and PAES) overall on these problems.

2.11 Pareto Envelope - based Selection Algorithm II (PESA-II)

Corne et al. [44] introduce PESA-II algorithm. PESA-II employs a new selection method. Specifically in the selection technique used by PESA-II, instead of assigning a selective fitness to an individual, selective fitness is assigned to the hyperboxes in objective space which are currently occupied by at least one individual in the current approximation to the Pareto frontier. A hyperbox is thereby selected, and the resulting selected individual is randomly chosen from this hyperbox. According to the authors this selection method is shown to be more sensitive to ensuring a good spread of development along the Pareto frontier than individual – based selection. The method is implemented in PESA-II (a version of PESA which uses region-based selection) and performance is tested by using Deb’s test suite [51]. According to their study, the PESA-II using the new selection technique is found to give significantly superior results to the other methods compared namely PAES, PESA and SPEA.

2.12 Archive-Based Hybrid Scatter Search *(AbYSS)*

Nebro et al. [166] proposed the use of a new algorithm called AbYSS to solve multiobjective optimization problems. The authors adapt the well-known scatter search template for single-objective optimization to the multiobjective domain. The result is a hybrid metaheuristic algorithm, which follows the scatter search structure but uses mutation and crossover operators from evolutionary algorithms. AbYSS incorporates typical concepts from the multiobjective field, such as Pareto dominance, density estimation, and an external archive to store the non-dominated solutions. AbYSS is evaluated with a standard benchmark including both unconstrained and constrained problems, and it is compared with NSGA-II and SPEA2. According to the authors, the results obtained indicate that AbYSS outperforms the other two algorithms as regards the diversity of the solutions, and it obtains very competitive results according to the convergence to the true Pareto fronts and the hypervolume metric.

2.13 Generalized Differential Evolution 3 (GDE3)

Kukkonen and Lampinen [132] developed a version of generalized differential evolution, called GDE3. GDE3 is an extension of differential evolution (DE) for global optimization with an arbitrary number of objectives and constraints. In the case of a problem with a single objective and without constraints GDE3 falls back to the original DE. According to the authors, GDE3 improves earlier GDE versions in the case of multi-objective problems by giving a better distributed solution. Performance of GDE3 is demonstrated with a set of test problems and the results are compared with other methods.

2.14 Indicator-Based Evolutionary Algorithm (IBEA)

Zitzler and Kunzli [230] examine how preference information of the decision maker can in general be integrated into multiobjective search. The main idea is to first define the optimization goal in terms of a binary performance measure (indicator) and then to directly use this measure in the selection process. IBEA can be combined with arbitrary indicators and in contrast to existing algorithms, can be adapted to the preferences of the user. Moreover, does not require any additional diversity preservation mechanism such as fitness sharing to be used. According to the authors IBEA outperforms NSGA-II and SPEA2, on several continuous and discrete benchmark problems with respect to different performance measures.

2.15 Cellular Genetic Algorithm for Multiobjective Optimization (MOCell)

Nebro et al. [164] introduced a new cellular genetic algorithm for solving multiobjective continuous optimization problems. MOCell uses an external archive to store non-dominated solutions and a feedback mechanism in which solutions from this archive randomly replaces existing individuals in the population after each iteration. The result is a simple and elitist algorithm called MOCell. The proposed algorithm has been evaluated with both constrained and unconstrained problems and compared against NSGA-II and SPEA2. The relevant experimental results indicate that MOCell obtains competitive results in terms of convergence, and outperforms the other two compared algorithms concerning the diversity of solutions along the Pareto front.

2.16 The Multiobjective Evolutionary Algorithm based on Decomposition (MOEA/D)

Zhang and Li [222] propose a multiobjective evolutionary algorithm based on decomposition (MOEA/D). It decomposes a multiobjective optimization problem into a number of scalar optimization subproblems and optimizes them simultaneously. Each subproblem is optimized by only using information from its several neighboring subproblems, which makes MOEA/D have lower computational complexity at each generation than MOGLS and nondominated sorting genetic algorithm II (NSGA-II). Experimental results have demonstrated that MOEA/D with simple decomposition methods outperforms or performs similarly to MOGLS and NSGA-II on multiobjective 0–1 knapsack problems and continuous multiobjective optimization problems. According to the authors the MOEA/D using objective normalization can deal with disparately-scaled objectives, and MOEA/D with an advanced decomposition method can generate a set of very evenly distributed solutions for 3-objective test instances.

2.17 The Multiobjective Evolutionary Algorithm based on Decomposition with Dynamical Resource Allocation (MOEA/DDRA)

In MOEA/D as introduced by [222], [223] all the sub problems are treated equally and receive about the same amount of computational effort. However, a more recent study by Zhang et al. [223] assigns different levels of computational effort in each sub problem based on the different level of difficulty in obtaining the solution. In particular, the new version of MOEA/D with a dynamical resource allocation (MOEA/D-DRA) computes a utility parameter π^i for each of the sub problems *i*, allowing computational effort to be distributed based on their utilities.

2.18 Speed-constrained Multiobjective particle swarm optimization (SMPSO).

Nebro et al. [165] presented a new multiobjective particle swarm optimization algorithm (PSO), called Speed-constrained Multiobjective PSO (SMPSO). The proposed algorithm uses of a strategy to limit the velocity of the particles by allowing to produce new effective particle positions in those cases in which the velocity becomes too high. The SMPSO uses polynomial mutation as a turbulence factor and an external archive to store the non-dominated solutions found during the search. The proposed algorithm is compared with respect to five multi-objective metaheuristics representative of the state-of-the-art in the area. The experimental results indicate that SMPSO obtains remarkable results in terms of the resulting approximation sets and the convergence speed.

2.19 S Metric Selection evolutionary multiobjective optimization algorithm (EMOA)

Beume et al. [18] proposed a steady-state EMOA that features a selection operator based on the hypervolume measure combined with the concept of non-dominated sorting. The proposed algorithm aims to maximize the dominated hypervolume within the optimisation process. According to the authors, the algorithm's population evolves to a well-distributed set of solutions, thereby focussing on interesting regions of the Pareto front. The performance of the proposed algorithm is compared to state-of-the-art methods on two- and three-objective benchmark suites as well as on aeronautical real-world applications.

2.20 SMS-EMOA with the Prospect Indicator (SMSP-EMOA)

Phan et al. [172] proposed an indicator-based EMOA, called SMSP-EMOA, as an extension of the SMS-EMOA. SMSP-EMOA uses the prospect indicator in its parent selection and the hypervolume indicator in its environmental selection. The prospect indicator measures the potential of each individual to reproduce offspring that dominate itself and spread out in the objective space. Thus, the proposed algorithm allows the parent selection operator to maintain sufficient selection pressure, even in high dimensional MOPs, thereby improving convergence velocity toward the Pareto-optimal front, and to diversify individuals, even in high dimensional MOPs, thereby spreading out individuals in the objective space. Experimental results show that SMSP-EMOA's parent selection operator complement its environmental selection operator. Experimental results indicate that SMSP-EMOA outperforms SMS-EMOA and other well-known traditional EMOAs in optimality and convergence velocity without sacrificing the diversity of individuals.

Chapter 3

Modern Portfolio Selection with the support of MOEAs: A Literature Review

3.1 Introduction

MOEAs can be useful in the solution of complex problems for which no efficient deterministic algorithm exists. In finance there are several NP-hard problems for which the use of a heuristic is clearly justified. Below, we provide a brief review of the most well known papers that deal with the implementation of MOEAs for the solution of the portfolio optimization problem.

3.2 Multiobjective Evolutionary Algorithms with Application in Portfolio Optimization: A Literature Review

Arnone et al. [9] were the first to use MOEAs for optimizing investment portfolios. The authors adopt the Markowitz model (i.e. maximize return while minimizing risk) however they do not adopt the variance of the distribution of portfolio returns as their measure of risk. Instead, they use lower partial moments, which refer to the down-side part of the distribution of returns. The use of downside risk makes the problem more difficult, because the shape of the objective surface is generally non-convex. Therefore, quadratic programming can no longer be used to find exact solutions. The authors adopt GA with a weighted linear aggregating function [152], [218] to solve this problem. The weights are called trade-off coefficients.

Shoaf and Foster [188], [189] use a GA with linear combination of weights for portfolio selection based on the Markowitz model. They minimize portfolio variance and maximize the expected return of the portfolio. The authors highlight an important issue: the encoding. The portfolio selection problem is an allocation problem. Thus, a direct representation (i.e. using decision variables as usually done with GAs for representing the weights of each stock) does not work well. The reason is that this type of representation will frequently produce infeasible solutions in which the values allocated do not sum to 1.0 which is a constraint imposed on the problem. They propose an alternative representation that solve this problem, but with the side-effect of higher sensitivity to the mutation and crossover rates.

Vedarajan, et al. [210] also adopt the two objectives from Markowitz's model: maximize the expected return of the portfolio and minimize risk. The authors use a GA with a linear aggregating function that combines the two objectives into a single scalar value, and in which the weights are varied in order to generate different nondominated solutions.

Chang et al. [28] consider the problem of finding the efficient frontier associated with the standard mean – variance portfolio optimization model. They extend the standard model by imposing cardinality constraints. They use three heuristic algorithms based upon genetic algorithms, tabu search and simulated annealing for finding the cardinality constrained efficient frontier and they present the computational results.

Lin et al. [145] consider a variation of the mean-variance model with fixed transaction costs and minimum transaction lots. The portfolio selection problem is modeled as a non-smooth nonlinear integer programming problem with multiple objective functions. They develop a genetic algorithm based on NSGA-II and Geconop for the solution of the proposed model and provide a numerical example to test the efficiency of the algorithm.

Schaerf A. [178] examines the mean – variance model but applies additional constraints on the cardinality of the portfolio and on the quantity of individual shares. He uses local search techniques, mainly tabu search. He also compares the previous work on portfolio selection that makes use of the local search approach and proposes new algorithms that combine different neighborhood relations.

Crama and Schyns [46] consider the classical mean – variance model enriched with additional realistic constraints. Their model is a mixed integer quadratic programming problem. They apply heuristic techniques (Simulated Annealing) as the exact optimization algorithms run into difficulty with their enriched mean-variance model.

Maringer and Kellerer [150] suggest a hybrid local search algorithm which combines principles of Simulated Annealing and evolutionary strategies. They applied the algorithm to the problem of portfolio selection when there are constraints on the number of different assets in the portfolio and non-negativity of the assets' weights, and they found the algorithm efficient and reliable.

Ehrgott et al [78] propose a model for portfolio optimization extending the Markowitz mean-variance model. Based on cooperation with Standard and Poor's they use five specific objectives related to risk and return and allow consideration of individual preferences through the construction of decision-maker specific utility function. Numerical results using customized local search, simulated annealing, tabu search and genetic algorithm heuristics show that problems of practically relevant size can be solved quickly.

Doerner et al [70] support that one of the most important management issues lies in determining the "best" portfolio out of a given set of investment proposals. They propose a two-phase procedure that first identifies the solution space of all efficient portfolios and then allows an interactive exploration of that space. However, determining the solution space is a challenging issue, because of the NP-hard nature of the problem. They propose meta-heuristics (Pareto Ant Colony Optimization) as a good approximate solution space.

Streichert et al. [201] investigate the impact of different crossover operators for a real-valued Evolutionary Algorithm on the constrained portfolio selection problem based on the Markowitz mean-variance model. They also introduce an extension of a real-valued genotype, which increases the performance of the Evolutionary Algorithm significantly, independent of the crossover operator used.

Subbu et al. [203] propose a hybrid multiobjective optimization approach that combines evolutionary algorithms with linear programming for investment portfolio optimization. The authors maximize the portfolio expected return, while minimizing both the surplus variance and the portfolio value at risk. They also consider duration and convexity mismatch constraints, as well as linear portfolio investment constraints. The authors use the Pareto Sorting Evolutionary Algorithm (PSEA), which uses a small population size and an archive that retains the nondominated solutions found along the search.

Armananzas and Lozano [8] apply three well-known optimization techniques: greedy search, simulated annealing and ant colony optimization for the solution of the portfolio optimization problem. They test the behaviors of the different algorithms by formulating Pareto fronts for five stock indexes.

Hochreiter [114] propose the use of heuristic algorithms for the financial decision optimization problems due to the complexity and non-convexity of financial engineering problems. The author examines the risk - return portfolio optimization problem and solve it with an evolutionary computation approach. Finally, the author conduct a comparison and provides numerical results of the Evolutionary Algorithm for three (Standard Deviation, Value at Risk and Conditional Value at Risk) risk measures.

Skolpadungket et al. [193] apply various techniques of multiobjective genetic algorithms to solve portfolio optimization with some realistic constraints, such as cardinality constraints and round-lot constraints. The authors experiment with VEGA, Fuzzy VEGA, MOGA, SPEA-II, NSGA-II. The results show that using fuzzy logic to combine optimization objectives of VEGA improves performances measured by Generation Distance (GD) defined by average distances of the last generation of population to the nearest members of the true Pareto front but its solutions tend to cluster around a few points. MOGA and SPEA-II use some diversification algorithms and they perform better in terms of finding diverse solutions around Pareto front.

Chiam et al. [34] consider an order-based representation for evolutionary multiobjective portfolio optimization, which can be extended to handle various realistic constraints like floor and ceiling constraint and cardinality constraint. The authors provide a comprehensive experimental study in evolutionary multiobjective portfolio optimization, which includes the evaluation of the algorithmic performance based on a set of benchmark problems with proper performance metrics and statistical tests.

Cesarone et al [27] present a Limited Asset Markowitz (LAM) model as they call it, with the introduction of quantity and cardinality constraints. The model is a Mixed Integer Quadratic Programming problem. The authors test their model using real-world capital market indices and they report that the LAM model outperforms the classical Markowitz portfolio.

Branke et al. [24] propose to integrate an active set algorithm optimized for portfolio selection into a multi-objective evolutionary algorithm (MOEA). According to the authors, the idea is to let the MOEA come up with some convex subsets of the set of all feasible portfolios, solve a critical line algorithm for each subset, and then merge the partial solutions to form the solution of the original non-convex problem. They show that the proposed envelope-based MOEA outperforms existing MOEAs.

Anagnostopoulos and Mamanis [4] investigate the ability of Multiobjective Evolutionary Algorithms namely the Non-dominated Sorting Genetic Algorithm II (NSGA-II), Pareto Envelope-based Selection Algorithm (PESA) and Strength Pareto Evolutionary Algorithm II (SPEA-II), for solving complex portfolio optimization problems. The authors evaluated the aforementioned MOEAs techniques with results provided by exact methods. They conclude that the PESA performs the best in terms of the closeness to the Pareto optimal front, while NSGA-II and SPEA-II have the best average performance in terms of hypervolume indicator. According to the authors another advantage of the methods is their flexibility to adapt in any addition of new constraint and / or replacement of the risk function.

Chang et al [30] introduce a heuristic approach to portfolio optimization problems in different risk measures by employing genetic algorithm (GA) and compare its performance to mean-variance model in cardinality constrained efficient frontier. They show that complex portfolio optimization problems can be solved by GA if mean-variance, semivariance, mean absolute deviation and variance with skewness are used as the measures of risk. They provide empirical result in order to prove the robustness of their heuristic method.

Soleimani et al. [195] consider a portfolio selection model which is based on Markowitz's portfolio selection problem including three constraints: minimum transaction lots, cardinality constraints and market capitalization. They solve this mixed-integer nonlinear programming problem (NP-Hard) using a Genetic Algorithm

Anagnostopoulos and Mamanis [5] consider the portfolio selection as a tri-objective optimization problem so as to find tradeoffs between risk, return and the number of securities in the portfolio. The authors apply quantity and class constraints into the model in order to limit the proportion of the portfolio invested in assets with common characteristics and to avoid very small holdings. The resulting model is a mixed-integer multiobjective optimization problem and the authors apply multiobjective optimization techniques, namely the Non-dominated Sorting Genetic Algorithm II (NSGA-II), Pareto Envelope-based Selection Algorithm (PESA), and Strength Pareto Evolutionary Algorithm II (SPEA-II), for its solution.

Thomaidis N. [206] considers the problem of structuring a portfolio that outperforms a benchmark index, assuming restrictions on the total number of tradable assets. The author experiments with non-standard formulations of active portfolio management, outside the mean-variance framework, incorporating approximate (fuzzy) investment targets and portfolio constraints. He applies three nature-inspired optimization procedures namely simulated annealing, genetic algorithms and particle swarm optimization. Finally, optimal portfolios derived from these methods are benchmarked against the Dow Jones Industrial Average index.

Deng and Lin [63] consider the Particle Swarm Optimization (PSO) for solving the cardinality constrained portfolio optimization problem. The authors propose an improved PSO that increases exploration in the initial search steps and improves convergence speed in the final search steps. They test the algorithm using five stock indexes. The computational test results indicate that the proposed PSO outperforms basic PSO algorithm, genetic algorithm (GA), simulated annealing (SA), and tabu search (TS) in most cases.

Deng and Lin [64] in another paper apply Ant Colony Optimization (ACO) for solving the cardinality constrained Markowitz mean-variance portfolio model, which is a nonlinear mixed quadratic programming problem. The authors obtain numerical solutions for five stock indexes. The test results indicate that the ACO is much more robust and effective than Particle Swarm Optimization (PSO), especially for low risk investment portfolios.

Anagnostopoulos and Mamanis [6] compare the effectiveness of five multiobjective evolutionary algorithms (MOEAs) together with a steady state evolutionary algorithm on the mean-variance cardinality constrained portfolio optimization problem . The MOEAs considered are the Niched Pareto Genetic Algorithm II (NPGA-II), Nondominated Sorting Genetic Algorithm II (NSGA-II), Pareto envelope-based selection algorithm (PESA), Strength Pareto Evolutionary Algorithm II (SPEA-II), and e-Multiobjective Evolutionary Algorithm (e-MOEA).

Woodside et al. [216] consider the application of genetic algoritihm, tabu search and simulated annealing metaheuristic approaches to finding the cardinality constrained efficient frontier that arises in portfolio optimization. The authors consider the mean-variance model as extended to include the discrete restrictions of buy-in thresholds and cardinality constraints.

Chen et al [32] apply a Genetic Relation Algorithm (GRA) with an operator, called guided mutation for solving large scale portfolio optimization problems. According to the authors, in order to pick up the most efficient portfolio, GRA considers the correlation coefficient between stock brands. Guided mutation generates offspring according to the average value of correlation coefficient in each individual.

Zhu et al. [226] suggest a meta-heuristic approach to portfolio optimization problem using Particle Swarm Optimization (PSO) technique. The authors test the model on various restricted and unrestricted risky investment portfolios and a comparative study with Genetic Algorithms is implemented.

Golmakani and Fazel [102] propose a heuristic method for solving an extended mean – variance portfolio selection model. Their model includes four sets of constraints namely: bounds on holding, cardinality, minimum transaction lots and sector capitalization constraints. Their model is a quadratic mixed-integer programming model and they propose a heuristic based on Particle Swarm Optimization (PSO) method. The authors compare their model with the Genetic Algorithm (GA) and the computational results show that the proposed PSO effectively outperforms GA especially in large-scale problems.

Deb et al. [59] examine a Bi-objective portfolio optimization for minimizing risk and maximizing expected return. The authors modify a bi-objective evolutionary algorithm (NSGA-II) in order to develop a customized hybrid NSGA-II procedure for handling situations that are non-conventional for classical quadratic programming approaches. Finally, the authors demonstrate how the evolutionary algorithms, for large-scale problems succeed to find fronts that can be difficult for other methods to obtain.

M. Woodside-Oriakhi et al. [217] consider the problem of rebalancing an existing financial portfolio, where transaction costs have to be paid if we change the amount held of any asset. These transaction costs can be fixed or variable (related to the amount traded). The authors indicate the importance of the investment horizon when rebalancing such a portfolio and illustrate the nature of the efficient frontier that results when in the presence of transaction costs. They model the problem as a mixed-integer quadratic programme with an explicit constraint on the amount that can be paid in transaction cost. Their model incorporates the interplay between optimal portfolio allocation, transaction costs and investment horizon. Experimental results are presented for the solution of publicly available test problems involving up to 1317 assets.

Andriosopoulos and Nomikos [7] make use of two evolutionary algorithms – the differential evolution algorithm and the genetic algorithm for addressing the index-tracking problem for passive investment. The performance of the suggested investment strategy is tested under three different scenarios: buy-and-hold, quarterly and monthly rebalancing, accounting for transaction costs where necessary.

Chen and Wang [33] proposed a hybrid stock trading system based on Genetic Network Programming (GNP) and Mean Conditional Value-at-Risk Model (GNP–CVaR). According to the authors proposed approach, combines the advantages of evolutionary algorithms and statistical model and provides a useful tool to construct portfolios and generate effective stock trading strategies for investors with different risk-attitudes. Simulation results on five stock indices show that model based on GNP and maximum Sharpe Ratio portfolio performs the best in bull market, and that based on GNP and the global minimum risk portfolio performs the best in bear market. The portfolios constructed by Markowitz's mean–variance model perform the same as mean-CVaR model [124]. According to the authors, the proposed system significantly improves the function and efficiency of original GNP, which can help investors make profitable decisions.

Corazza et al. [43] consider a nonlinear mixed-integer portfolio selection model which takes into account several constraints used in fund management practice. The problem is NP-hard and exact algorithms for its minimization, which are both effective and efficient, are still sought at present. Thus, to approximately solve this model the authors make use of the heuristics Particle Swarm Optimization (PSO). Since PSO was originally conceived for unconstrained global optimization problems, the authors apply it to a novel reformulation of our mixed-integer model, where a standard exact penalty function is introduced.

Fu et al. [95] examine the application of the Genetic Algorithms (GA) and the Hierarchical GA in portfolio management. The authors examine different algorithms and the usage of different numbers of technical indicators to determine the optimized parameters setting of different technical indicators and portfolio weighting in different economic situations.

Gupta et al. [105] proposed a multiobjective credibilistic model with fuzzy chance constraints of the portfolio selection problem. The key financial criteria used are short-term return, long-term return, risk and liquidity. The model generates portfolios which are optimal to the extent of achieving the highest credibility values for the objective functions. The problem is solved using a hybrid intelligent algorithm that integrates fuzzy simulation with a real-coded genetic algorithm. The authors provide numerical examples to demonstrate effectiveness of the solution approach and efficiency of the model.

Li and Xu [137] address the multi-objective portfolio selection model with fuzzy random returns for investors by studying three criteria: return, risk and liquidity. In addition, securities historical data, experts' opinions and judgements and investors' different attitudes are considered in the portfolio selection process, such that the investor's individual preference is reflected by an optimistic–pessimistic parameter λ. To avoid the difficulty of evaluating a large set of efficient solutions and to ensure the selection of the best solution, a compromise approach-based genetic algorithm has been designed to solve the proposed model. Finally, the authors provide a numerical example to illustrate the proposed algorithm.

Lwin et al. [147] studied the extended Markowitz's mean-variance portfolio optimization model. They considered the cardinality, quantity, pre-assignment and round lot constraints in the extended model. These four real-world constraints limit the number of assets in a portfolio, restrict the minimum and maximum proportions of assets held in the portfolio, require some specific assets to be included in the portfolio and require to invest the assets in units of a certain size respectively. The authors propose an efficient learning-guided hybrid multiobjective evolutionary algorithm to solve the constrained portfolio optimization problem in the extended mean-variance framework. A learning-guided solution generation strategy is incorporated into the multiobjective optimization process to promote the efficient convergence by guiding the evolutionary search towards the promising regions of the search space. The proposed algorithm is compared against four existing state-of-the-art multi-objective evolutionary algorithms, namely Non-dominated Sorting Genetic Algorithm (NSGA-II), Strength Pareto Evolutionary Algorithm (SPEA-2), Pareto Envelope-based Selection Algorithm(PESA-II) and Pareto Archived Evolution Strategy (PAES). Computational results are reported for publicly available OR-library datasets from seven market indices involving up to 1318 assets. Experimental results on the constrained portfolio optimization problem demonstrate that the proposed algorithm significantly outperforms the four well-known multi-objective evolutionary algorithms with respect to the quality of obtained efficient frontier in the conducted experiments.

Mishra et al. [158] consider the portfolio assets selection problem as a multiobjective optimization one, considering the budget, floor, ceiling and cardinality as constraints. A novel multiobjective optimization algorithm, namely the non-dominated sorting multiobjective particle swarm optimization (NS-MOPSO), has been proposed and employed to solve the portfolio selection problem. The performance of the proposed algorithm is compared with four multiobjective evolution algorithms (MOEAs), based on non-dominated sorting, and one MOEA algorithm based on decomposition (MOEA/D). The performance results obtained from the study are also compared with those of single objective evolutionary algorithms, such as the genetic algorithm (GA), tabu search (TS), simulated annealing (SA) and particle swarm optimization (PSO). The comparisons of the performance include three error measures, four performance metrics, the Pareto front and computational time. A nonparametric statistical analysis, using the Sign test and Wilcoxon signed rank test, is also performed, to demonstrate the superiority of the NS-MOPSO algorithm. On examining the performance metrics, it is observed that the proposed NS-MOPSO approach is capable of identifying good Pareto solutions, maintaining adequate diversity. The proposed algorithm is also applied to different cardinality constraint conditions, for six different market indices, such as the Hang-Seng in Hong Kong, DAX 100 in Germany, FTSE 100 in UK, S&P 100 in USA, Nikkei 225 in Japan, and BSE-500 in India.

Suksonghong et al. [204] experiments the use of multi-objective genetic algorithms (MOGAs), namely, the nondominated sorting genetic algorithm II (NSGA-II), strength Pareto evolutionary algorithm II (SPEA-II) and newly proposed compressed objective genetic algorithm II (COGA-II) for solving the portfolio optimization problem for a power generation company (GenCo) faced with different trading choices. The authors introduce an additional objective to enhance the diversification benefit alongside with the three original objectives of the mean–variance–skewness

(MVS) portfolio framework. Experimental results show that the inclusion of the fourth objective within the MOGAs produces Pareto fronts that also cover those based on the traditional MVS framework, thereby offering better trade-off solutions while promoting investment diversification benefits for power generation companies.

Babaei et al. [12] attempt to shed some light on the dependence structure among the financial returns along with the fat-tailed distribution associated with them for the Portfolio optimization problem. The authors try to find a remedy for this shortcoming by exploiting stable distributions as the marginal distributions together with the dependence structure based on copula function.

The portfolio optimization problem is formulated as a multi-objective mixed integer programming. Value-at-Risk (VaR) is specified as the risk measure due to its intuitive appeal and importance in financial regulations. In order to enhance the model's applicability, the authors examine cardinality and quantity constraints in the model. Imposing such practical constraints has resulted in a non-continuous feasible region. The authors propose two variants of multi-objective particle swarm optimization (MOPSO) algorithms to tackle this issue. Finally, a comparative study among the proposed MOPSOs, NSGAII and SPEA2 algorithms is made to demonstrate which algorithm is outperformed. The empirical results reveal that one of the proposed MOPSOs is superior over the other salient algorithms in terms of performance metrics.

Chapter 4

A methodological framework for conducting a comprehensive literature study based on the papers published in MOEAs for the Portfolio Management

In this section we present a methodological framework for analyzing the available literature on the Multi-objective Evolutionary Algorithms for Portfolio Selection. Specifically 91 papers have been analyzed according to a number of different parameters such as which authors, journals or countries' institutions have contributed to the research in the field and which are the most popular objectives, constraints and risk measures applied in formulation of portfolio management models with the support of MOEAs. The purpose of such analysis is to provide an insight into the current state of research in Multi-Objective Evolutionary Algorithms and to identify potential areas of concern with regard to MOEAs for the Portfolio Management.

4.1 The Methodological Approach

In the first stage of our study we had to determine which type of publications to include in the review e.g. journal publications, book series, conference papers, books. We decided to include journal publications and book series. The main reasons for concentrating on journal and book series papers are: 1. Journal and book series papers are easily accessed through online databases, 2. Books are usually focused on explaining a research field, their purpose is to make a field of study approachable to a wide audience. However, our study is focused on presenting the latest advances in MOEA for the Portfolio Management which in most of the cases is beyond the purpose of a book. 3. Conference papers are not easily accessible. The journals and book series that included in our study, accessed through two online bibliographical databases: ScienceDirect – Elsevier and Springer as the Table 4.1 indicates.

Table 4.1 Bibliographical databases

Bibliographical databases	
ScienceDirect – Elsevier	ScienceDirect is an electronic full text database of scientific journals. It includes more than 2,500 journals and more than nine million full-text articles.
Springer	Springer is a scientific publisher with about 2,000 journals, focusing mainly in science, technology, medicine, business, transport and architecture.

Furthermore, we had to decide which period to include in the literature study. We find only two publications in Multi-objective Evolutionary Algorithms for the Portfolio Management prior to 1994 (Dueck and Winker [75], Arnone et al. [9]). For that reason, we decided that it was reasonable to include publications from 1994 and onwards. Thus, the period we decided to include in the literature review is from 1994 to 2011.

Figure 4.1 displays the relationship between MOEAs papers (field independent) and MOEAs papers focusing on Portfolio Management, for the period of time from 1994 to 2011.

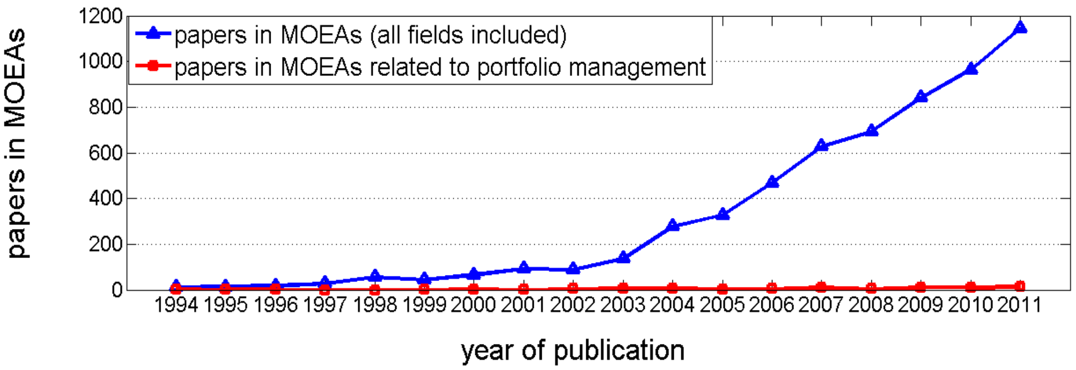


Figure 4.1 Papers published in MOEAs (all fields of study included) compared with paper in MOEAs related to portfolio management

Figure 4.1, indicates clearly that the undergoing research in the MOEAs focusing on Portfolio Management is in its early stages. On the other hand, we notice an impressive growth in the number of published papers related to MOEAs in general, from 2003 and onwards.

After choosing the type of publication (journals and book series papers) and the period of time (from 1994 to 2011), the next important question we had to answer was to decide which papers to include and which keywords we want to use in the search and where the keywords should appear namely: the title, abstract or keywords of the paper. As we mentioned earlier we chose two electronic bibliographic databases and we searched for the keywords "MOEAs" and "Multi-objective Evolutionary Algorithms" within titles, abstracts or keywords. In order to exclude editorial comments or book review the length of papers had to be at least four pages. From this search we ended up with a total pool of 6,620 journal and book series papers. From this pool of 6,620 papers to our database 692 papers were excluded as it was obvious from their titles that the specific papers did not deal with MOEAs or they were prefaces of books or contents of journals volumes or editorial notes. Finally publications excluded because only mentioned MOEAs or Multi- objective Evolutionary Algorithms as an example of one technique among other techniques without focusing on MOEAs. Thus, we ended up with a final pool of 5,928 papers. From this pool of 5,928 papers in MOEAs, 91 of them focus on MOEAs for Portfolio Management.

4.2 Findings about the MOEAs for the Portfolio Management

As we mentioned earlier, the purpose of our study is to provide an insight into the current state of research in Multi-Objective Evolutionary Algorithms and to identify potential areas of concern with regard to MOEAs for the Portfolio Management. In the following pages we will answer questions such as which journals, authors and which countries' institutions have contributed the most in the development of the field and which are the most popular objectives, constraints and risk measures applied in the formulation of portfolio management models with the support of MOEAs

We start by presenting the contribution of each individual journal to the study of the field.

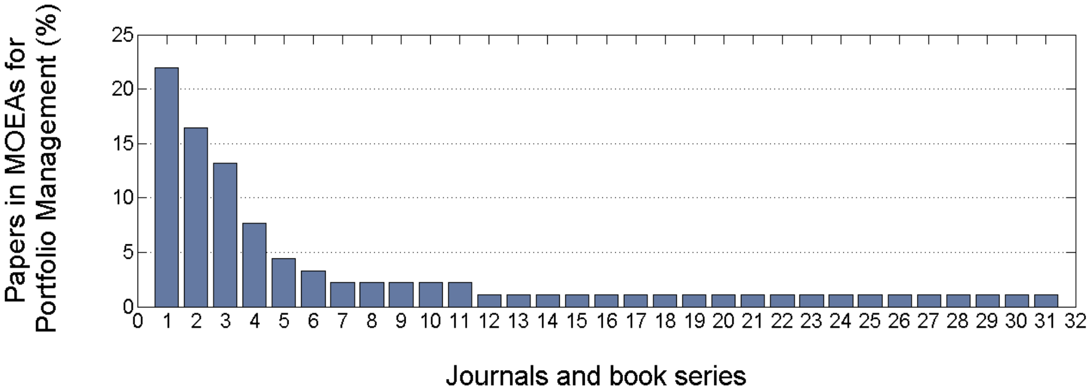


Figure 4.2 Papers in MOEAs focusing on Portfolio Management per Journal

From Figure 4.2 and Table 4.2 it is evident that the first three journals: *1. Lecture Notes in Computer Science (it is actually a book series), 2. Expert Systems with Applications* and *3. European Journal of Operational Research*, concentrate the majority of the publications in MOEAs focusing on Portfolio Management.

Table 4.2 Contribution of various Journals to the study of MOEAs for the Portfolio Management

Contribution of Journals to the study of MOEAs for Portfolio Management		
	Papers	Percentage
1. Lecture Notes in Computer Science	20	21.97%
2. Expert Systems with Applications	15	16.47%
3. European Journal of Operational Research	12	13.18%
4. Computers & Operations Research	7	7.68%
5. IEEE Transactions on Evolutionary Computation	4	4.40%
6. Computational Statistics & Data Analysis	3	3.30%
7. Applied Soft Computing	2	2.20%
8. Information Sciences	2	2.20%
9. Annals of Operations Research	2	2.20%
10. OR Spectrum	2	2.20%
11. International Transactions in Operational Research	2	2.20%
12. Applied Mathematics and Computation	1	1.10%
13. Electric Power Systems Research	1	1.10%
14. Mathematical Programming	1	1.10%
15. Computational Economics	1	1.10%
16. Computational Management Science	1	1.10%
17. Computational Optimization and Applications	1	1.10%
18. International Journal of Automation and Computing	1	1.10%
19. Quantitative Finance	1	1.10%
20. Nonlinear Analysis: Real World Applications	1	1.10%
21. International Journal of Operations Research	1	1.10%
22. Journal of Risk	1	1.10%
23. Mathematical Finance	1	1.10%
24. Optimization Methods and Software	1	1.10%
25. Advanced Modeling and Optimization	1	1.10%
26. ORSA Journal on Computing	1	1.10%
27. Evolutionary Computation	1	1.10%
28. Complex Systems	1	1.10%
29. Automation in Construction	1	1.10%
30. Journal of Mathematical Economics	1	1.10%
31. Journal of Economic Behavior & Organization	1	1.10%

Below, we will attempt to identify which authors have contributed the most to the field. According to Table 4.3, a total of 199 different authors, either as single authors or as co-authors have contributed to the total pool of papers. The 79.90% have contributed to a single paper and only 14.57% have contributed to two papers. The most contributing author was *Deb, K.* with 6 publications [24], [54], [57], [59], [161], [196].

Table 4.3 Papers in MOEAs focusing on Portfolio Management per author

Publications		
	Number of authors	**Percentage**
1	159	79.90%
2	29	14.57%
3	9	4.53%
4	1	0.50%
5	0	0.00%
6	1	0.50%

Note: 199 different authors

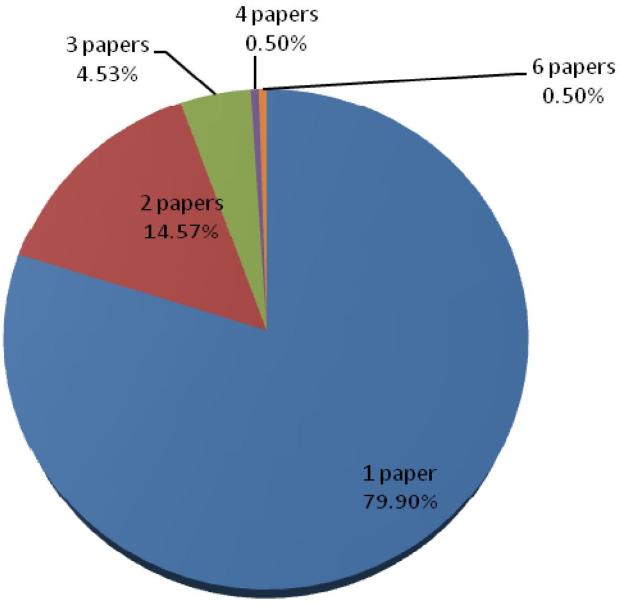


Figure 4.3 Papers in MOEAs focusing on Portfolio Management per author

Table 4.4 displays the most prolific authors in MOEAs focusing on the problem of Portfolio Management.

Table 4.4 The most prolific authors in MOEAs focusing on the problem of Portfolio Management.

Author	Number of publications in MOEAs for Portfolio Management
Deb, K.	6
Steuer, R.E.	4
Al Mamun, A.	3
Anagnostopoulos, K. P.	3
Beasley, J.E.	3
Chang, T.J.	3
Chiam, S. C.	3
Qi, Y.	3
Hirschberger, M.	3
Mamanis G.	3
Tan, K. C.	3
Agarwal, S.	2
Brabazon, A.	2
Chen, Y.	2
Deng, G.F.	2
Di Tollo, G.	2
Diosan, L.	2
Doerner, K.F.	2
Ehrgott, M.	2
Drezewski, R.	2
Fieldsend, J.E.	2
Gilli, M.	2
Golmakani, H. R.	2
Gutjahr, W.J.	2
Hartl, R.F.	2
Lin, W.T.	2
Lucas, C.	2
Meade, N.	2
Molina, J.	2
O’Neill, M.	2
Oltean, M.	2
Schaerf, A.	2
Schlottmann, F.	2
Schmeck, H.	2
Seese, D.	2
Siwik, L.	2
Stein, M.	2
Strauss, C.	2
Stummer, C.	2
Tewari, R.	2

Another issue of concern in our analysis is the nationality of the institutions that contributed to the study of the MOEAs for Portfolio Management. In order to obtain the necessary data we kept a record of the nationality of authors and co-authors' institutions at the time the articles were published. Table 4.5 presents the relevant results.

Table 4.5 Contribution of various countries institutions to the study of MOEAs for Portfolio Management

Contribution of various countries institutions to the study of MOEAs for Portfolio Management		
	Institutions	**Percentage**
USA	32	12.60%
UK	30	11.81%
Germany	30	11.81%
China	17	6.69%
India	17	6.69%
Taiwan	12	4.72%
Spain	11	4.33%
Austria	11	4.33%
Italy	10	3.94%
Singapore	9	3.54%
Iran	8	3.15%
Ireland	8	3.15%
Greece	7	2.76%
Switzerland	6	2.36%
Poland	6	2.36%
Portugal	5	1.97%
South Korea	4	1.58%
Romania	4	1.58%
Japan	4	1.58%
France	4	1.58%
Canada	3	1.18%
Saudi Arabia	2	0.79%
Netherland	2	0.79%
Norway	2	0.79%
Denmark	2	0.79%
Belgium	2	0.79%
Australia	1	0.39%
New Zealand	1	0.39%
Turkey	1	0.39%
Colombia	1	0.39%
Monaco	1	0.39%
Brazil	1	0.39%

The Table 4.5 is dominated by US, UK and German institutions that count together for about 36% of the total publications in MOEAs focusing on Portfolio Management. Chinese and Indian institutions are next with about 6.7% of the total publications produced to each one of these two countries. Significant contribution to the study of the field display institutions of smaller countries like Austria, Greece, Ireland and Switzerland.

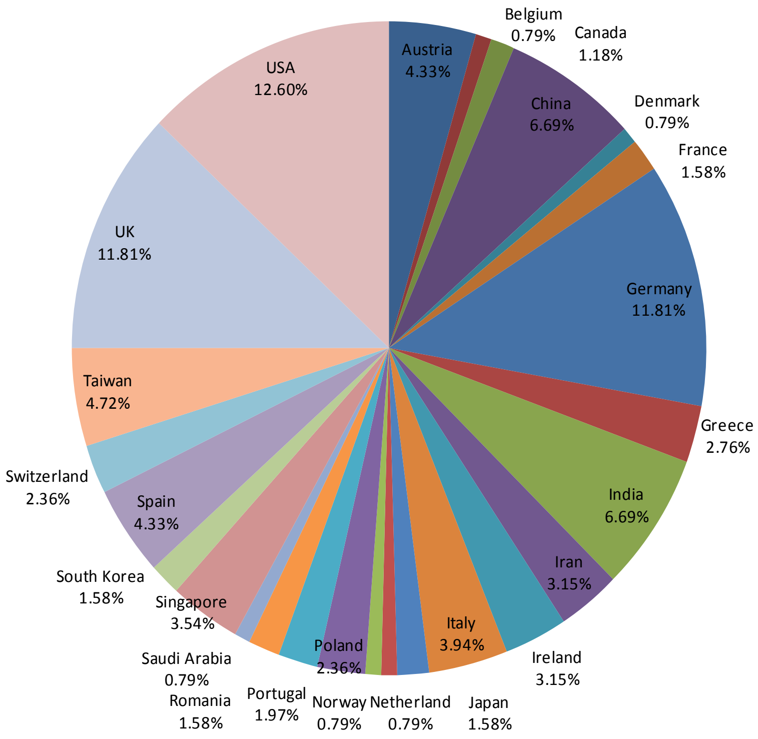


Figure 4.4 Contribution of various countries institutions to the study of MOEAs for Portfolio Management

Figure 4.5 displays the number of objectives used by the scholars in MOEAs for the Portfolio Management.

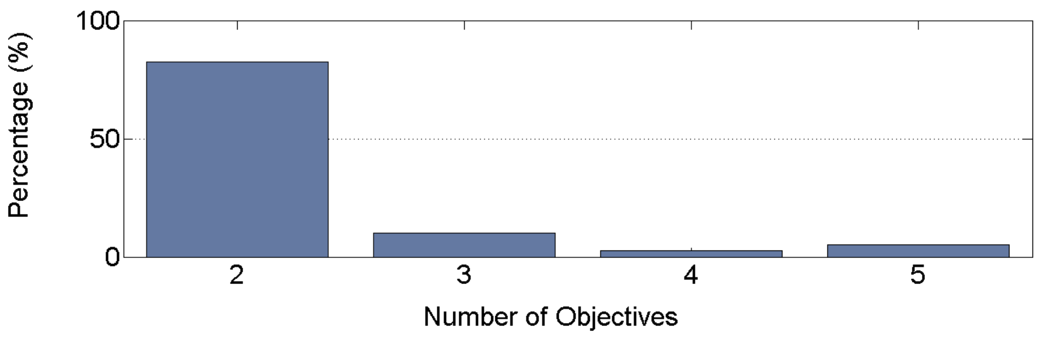


Figure 4.5 Number of objectives in the formulation of MOEAs for Portfolio Management models

From Figure 4.5 it is evident that the majority of the scholars select to use only two objectives in their Multiobjective Evolutionary Algorithms for the Portfolio Management Models.

Table 4.6 Number of objectives in the formulation of MOEAs for Portfolio Management models

Number of Objectives	
	Percentage
Two Objectives	82.50%
Three Objectives	10.00%
Four Objectives	2.50%
Five Objectives	5.00%

Table 4.7 displays which are the most popular *objectives* in MOEAs for Portfolio Management models among the authors.

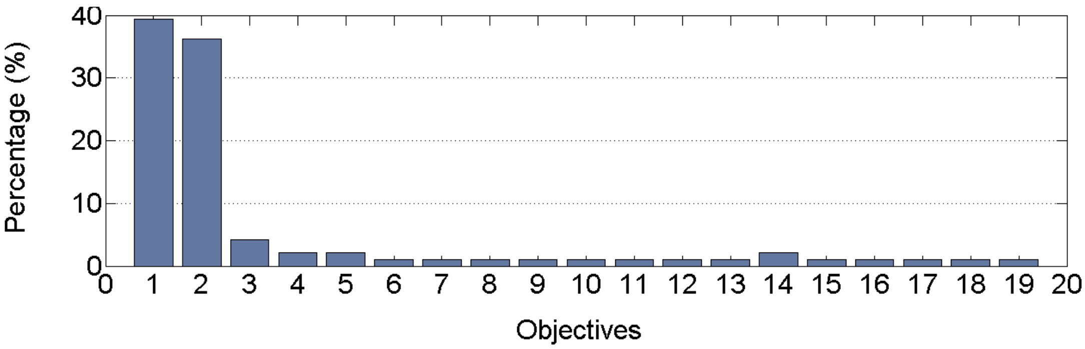


Figure 4.6 The most commonly used Objectives in MOEAs for Portfolio Management models

Table 4.7 The most popular Objectives in MOEAs for Portfolio Management models

The most popular Objectives in the formulation of MOEAs for Portfolio Management Models	
	Percentage
1. Mean	39.38%
2. Variance	36.18%
3. VaR	4.27%
4. Annual Dividends	2.13%
5. Expected Shortfall	2.13%
6. 12-months performance	1.06%
7. Probability that the portfolio delivers positive returns relatively to a benchmark	1.06%
8. 3-year performance	1.06%
9. Standard and Poor's star ranking	1.06%
10. Cardinality Constraints	1.06%
11. Social responsibility	1.06%
12. Research & Development	1.06%
13. 12-month volatility	1.06%
14. Skewness	2.13%
15. Sharpe Ratio	1.06%
16. CVaR	1.06%
17. Lower Partial Moments (LPM)	1.06%
18. Semi-variance	1.06%
19. Mean Absolute Deviation (MAD)	1.06%

As we can see from Table 4.7, the most popular *objectives* among authors in the study of MOEAs for Portfolio Management are the *Mean* (Portfolio's Expected Return) and *Variance* [13]. It is surprising that sixty years after the seminal work of Markowitz [79] that the majority of the authors still consider these two objectives introduced by Markowitz as the two dominants objectives in the study of the Portfolio Selection problem [17], [28], [30], [35], [46], [47], [57], [59], [63], [64], [65], [67], [85], [88], [102], [114], [119], [136], [144], [150], [169], [173], [178], [187], [195], [197], [216]. Others objectives that seem to attract the attention of scholars are: *VaR* [4], [99], *Annual Dividends* [78], [198], *Expected Shortfall* [4] , [99] and *Skewness*.

The next question we had to answer is which are the most commonly used *Risk Measures* [13] in the study of MOEAs for Portfolio Management models. Table 4.8 displays our findings about the most popular Risk Measures.

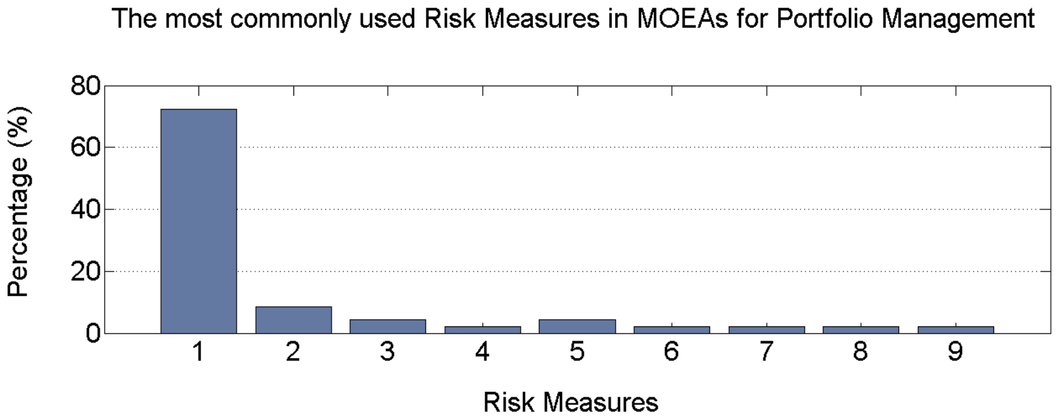


Figure 4.7 The most commonly used Risk Measures in MOEAs for Portfolio Management

Table 4.8 The most popular Risk Measures in MOEAs for Portfolio Management models

The most popular Risk Measures in the formulation of MOEAs for Portfolio Management	
	Percentage
1. Variance	72.4%
2. VaR	8.52%
3. Expected Shortfall	4.26%
4. 12-month volatility	2.13%
5. Skewness	4.26%
6. CVaR	2.13%
7. Lower Partial Moments (LPM)	2.13%
8. Semi-variance	2.13%
9. Mean Absolute Deviation (MAD)	2.13%

We notice that the most popular risk measure is the *Variance* [6], [17], [24], [28], [30], [46], [65], [67], [88], [114], [119], [178], [199], followed by the *VaR* [4], [65], [114], the *Expected Shortfall* [4], [114] and the *Skewness* [30]. It causes surprise the wide use of Variance as a risk measure although its well known shortcomings such as its lack of coherence [11] and penalization of both returns above and below expected return. A possible explanation is the easiness of variance's calculation.

Another issue of concern is how many constraints are used by the authors in the formulation of MOEAs for Portfolio Management Models. According to Figure 4.8 the majority of the scholars prefer to apply *two constraints* in the formulation of their models [5], [6], [17], [24], [26], [47], [59], [63], [64], [67], [78], [85], [136], [144], [150], [178], [203], [216].

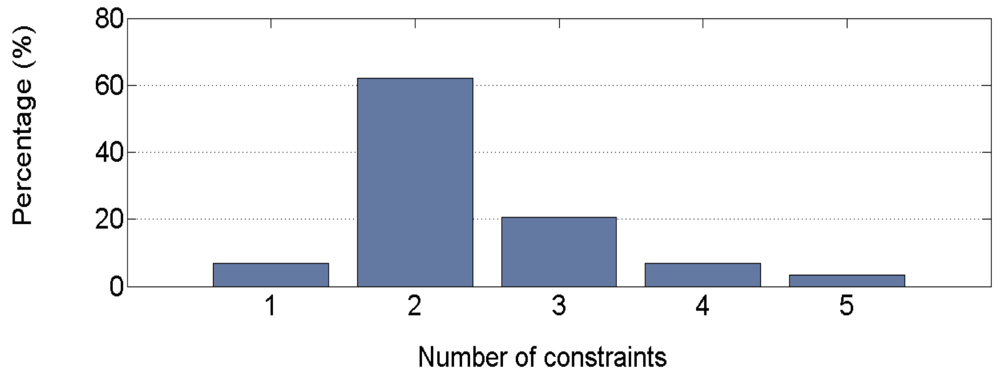


Figure 4.8 Number of constraints used in MOEAs for Portfolio Management models

Table 4.9 Number of constraints used in MOEAs for Portfolio Management models

Number of Constraints	
	Percentage
One constraint	6.90%
Two constraints	62.07%
Three constraints	20.69%
Four constraints	6.90%
Five constraints	3.44%

Knowing the number of constraints applied to the problem the next question that come naturally to the reader's mind is which are these constraints? Table 4.10 provides an answer to that question by displaying the most popular constraints among the authors of papers in MOEAs for Portfolio Management models.

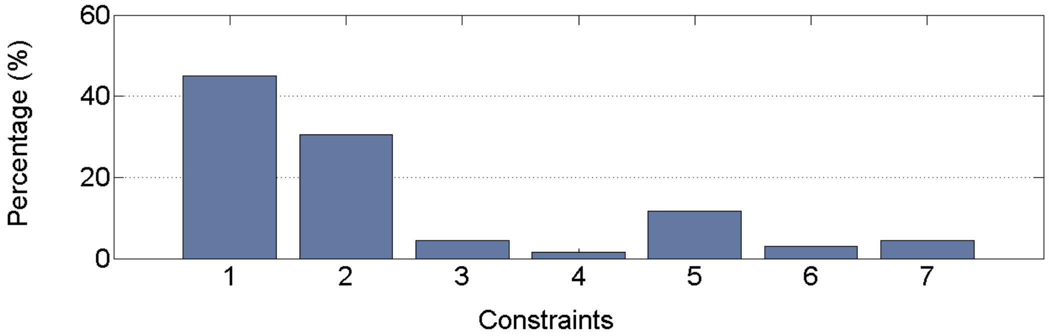


Figure 4.9 The most commonly used constraints in MOEAs for Portfolio Management models

Cardinality constraints [4], [6], [17], [24], [28], [30], [35], [46], [47], [65], [67], [78], [85], [88], [99], [114], [119], [136], [150], [178], [187], [195], [201] and *Lower and Upper bounds* [4], [5], [6], [17], [35], [46], [47], [65], [67], [78], [85], [99], [178] seem to dominate as the constraints of preference among the authors. Table 4.10 reveals the relevant figures.

Table 4.10 The most commonly used constraints in MOEAs for Portfolio Management models

The most popular Constraints in the formulation of MOEAs for Portfolio Management models	
	Percentage
1. Cardinality constraints	44.93%
2. Lower and Upper bounds	30.43%
3. Transaction costs	4.35%
4. The 5-10-40 constraint	1.45%
5. Transaction Roundlots	11.59%
6. Nonnegativity constraints	2.90%
7. Sector capitalization constraints	4.35%

The next question we had to answer is which disciplines have contributed the most to the development of MOEAs focusing on the Portfolio Management. In order to answer this question, first we had to determine the disciplines. For that purpose we used the discipline that the journal itself refers to on its web page under the title: “aim and goal” or “aim and scope”. Table 4.11 displays the emerged disciplines.

Table 4.11 Contribution of various disciplines to the development of MOEAs focusing on the Portfolio Management

Discipline	
	Definition
Computer Science	Journals focusing on technical aspects of Information Technology
Artificial Intelligence	Journals focusing on studying and designing intelligent systems
Operations Research	Journals focusing on the effective use of technology by organizations
Finance- Economics	Journals focusing on the management of money, banking, investment, credit and Journals dealing with production, distribution and consumption of goods and services
Mathematics - Statistics	Journals dealing with the logic of quantity, shape, arrangement and Journals focusing on the collection, organization and interpretation of data.
Other	Journals focusing on aspects not included in the above-mentioned five disciplines

Table 4.12 displays the classification of journals, publishing articles on the MOEAs for the Portfolio Management, in various *Disciplines*. The two dominant disciplines are those of *Operations Research* and *Computer Science*.

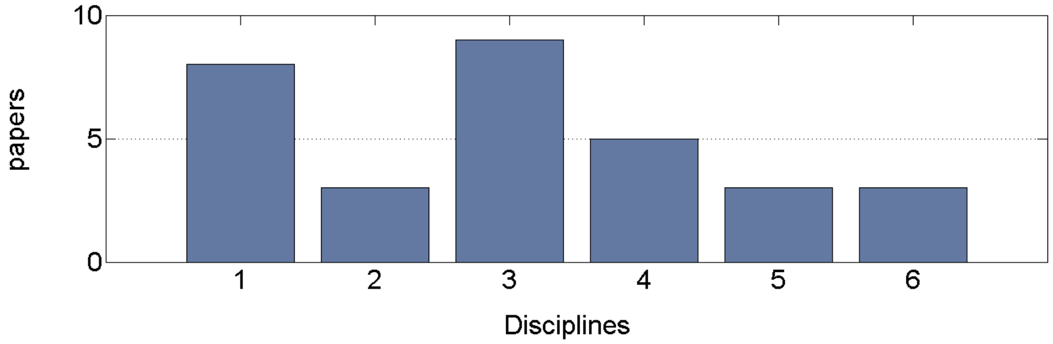


Figure 4.10 Disciplines of journals publishing papers in MOEAs for the Portfolio Management

Table 4.12 Disciplines of journals publishing papers in MOEAs for the Portfolio Management

Disciplines of Journals publishing papers in MOEAs for Portfolio Management	
	Percentage
1. Computer Science	25.80%
2. Artificial Intelligence	9.68%
3. Operations Research	29.03%
4. Finance – Economics	16.13%
5. Mathematics – Statistics	9.68%
6. Other	9.68%

A final issue that we will address is the classification of the papers according to the methodological approach followed by the authors. In order to be able to carry out this classification we went through the titles and abstracts of the papers. When the results from this preliminary study were inconclusive we went through the body of the article. Going through the papers we distinguished the following Methodological Categories that appear in the Table 4.13.

Table 4.13 Classification of the papers according to the methodological approach

Methodological Category	
	Definition
Empirical – Experiment	Papers focusing on gaining knowledge by means of quantitative or qualitative analysis of data and papers focusing on verifying, falsifying or establishing the accuracy of a hypothesis in order to explain a phenomenon or predict the results of an action.
Theoretical	Papers focusing on providing a synthesis of research on a topic for a specified period of time.
Model Design	Papers focusing on developing a new algorithm or mathematical model in order to explain a phenomenon or predict the results of an action.
Combined	Papers using a combination of the above – mentioned categories fall into this category.
Not mentioned	Papers that do not mention any methods either explicitly or implicitly.

The results of the methodological classification of the papers in MOEAs focusing on Portfolio Management appear in Table 4.14 and 4.15. Table 4.14 displays the various categories emerged after the classification and the corresponding numbers of papers.

The *Combined* methodological classification is the dominant approach adopted by the majority of the authors counting for the 81.31% of all cases.

Table 4.14 Methodological classification of the papers in MOEAs focusing on Portfolio Management

Methodological Categories		
	Papers	**Percentage**
Combined	74	81.31%
Model Design	9	9.89%
Review	2	2.20%
Theoretical	2	2.20%
Empirical – Experiment	4	4.40%

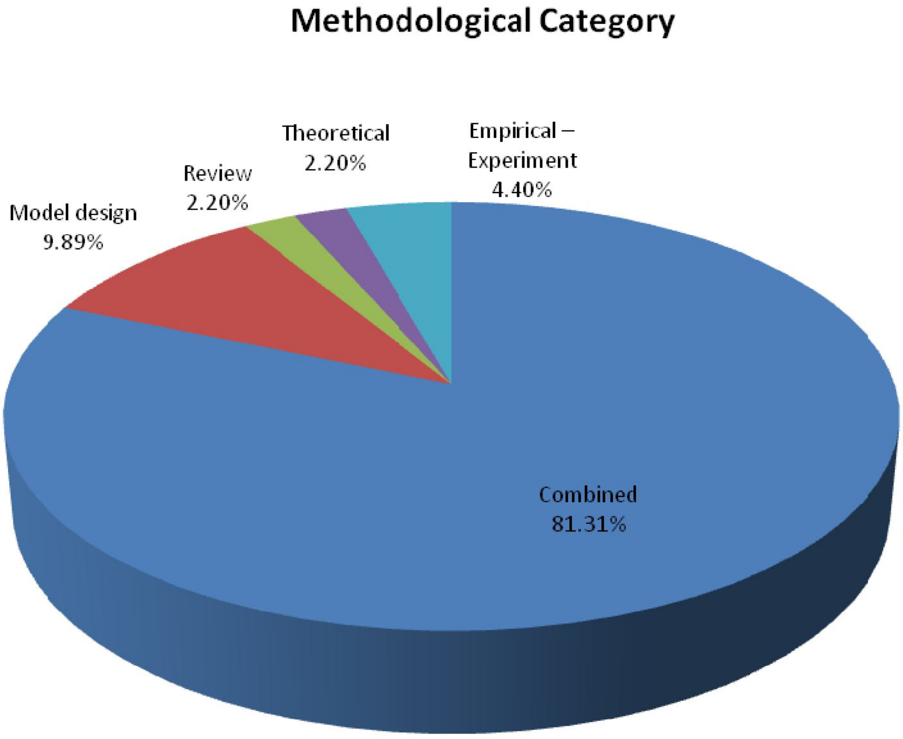


Figure 4.11 Methodological classification of the papers in MOEAs focusing on Portfolio Management

Table 4.15 indicates how the *Combined* methodological classification is even further analyzed into the various subcategories. As expected a mix of *Model Design* and *Empirical or Experimental* research is the way to go for most of the authors [24], [26], [27], [28], [29], [31], [36], [37], [38], [39], [44], [45], [49], [51], [54], [57], [69], [70], [71], [73], [74], [75], [79], [83], [100], [104], [106], [110], [113], [114], [115],

[118], [120], [135], [146], [148], [154], [161], [169], [171], [173], [183], [201], [210], [215], [226].

Table 4.15 Analysis of the Combined methodological classification into the various subcategories

Combined methodological approach is analyzed		
	Papers	**Percentage**
Model design and Empirical - Experiment	71	95.95%
Review and Empirical - Experiment	2	2.70%
Theoretical and Empirical - Experiment	1	1.35%

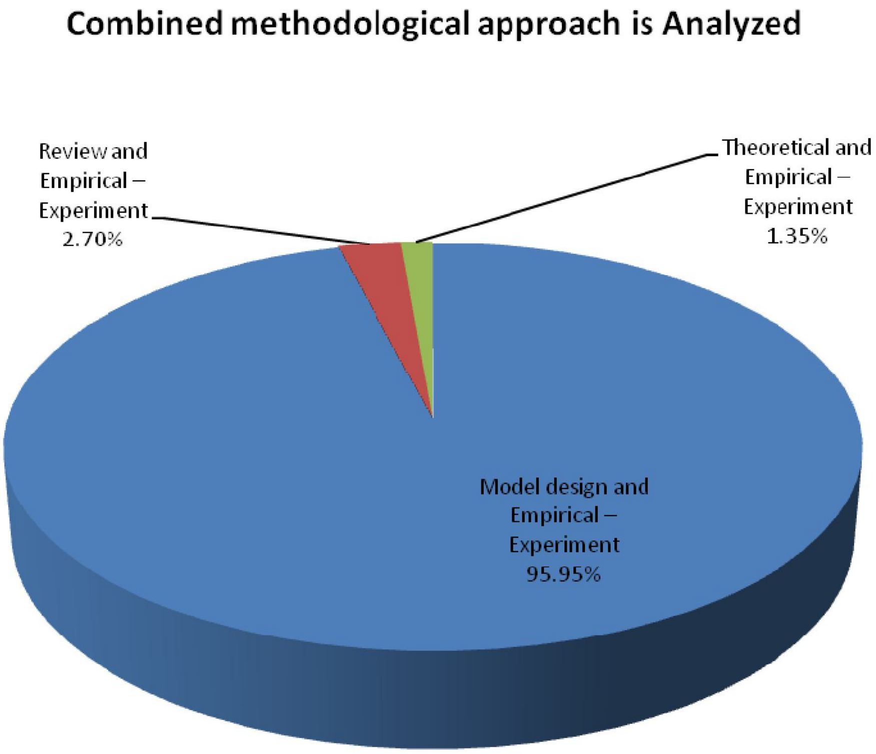


Figure 4.12 Analysis of the Combined methodological classification into the various subcategories

4.3 Conclusions

The purpose of this chapter is to provide an insight into the current state of research in Multi-objective Evolutionary Algorithms focusing on the Portfolio Management. To serve this purpose we analyzed 91 papers according to journal, author and nationality of authors' institution. Moreover, we addressed issues related to the problem's formulation such as which are the most popular objectives, constraints and risk measures that are applied in the portfolio management models with the support of MOEAs. On top of that we presented the various disciplines and the research methodologies of the available literature in the field.

We started by presenting the methodological framework for the analysis of the available literature on the Multi-objective Evolutionary Algorithms focusing on the Portfolio Management. Then, we presented graphically the sharp increase in the number of papers published in MOEAs field independent, to name some of them: engineering, science, finance, biology, medicine, genetics, robotics, physics and chemistry over the period 1994 – 2011. For the same period of time we presented the number of papers published in MOEAs focusing on the Portfolio Management and we noticed a relatively small increase compared with the impressive increase in the volume of papers published in MOEAs as a total. This indicates clearly that the undergoing research for the implementation of MOEAs to the Portfolio Selection problem is in its early stages and there is a lot of room for further research.

Furthermore, our analysis revealed that almost 80% of the authors in portfolio management with the support of MOEAs have contributed a single paper and only 14.57% have contributed in two papers. Moreover, we found out that US, UK and German institutions have contributed the most to the study of the MOEAs focusing on Portfolio Management. Chinese and Indian institutions are next with about 6.7% of the total publications produced to each one of these two countries. Significant contribution to the study of the field display institutions of smaller countries like Austria, Greece, Ireland and Switzerland.

The analysis revealed that 82.50% of the MOEAs for Portfolio Management models make use of only two *objectives* and that the *portfolio's expected return* and *variance* are the most popular among them, followed by *VaR*, *Annual Dividends*, *Expected Shortfall* and *Skewness*. Moreover, we found that the most widely used risk measure in the relevant literature is the *Variance*, followed by *VaR*, *Expected*

Shortfall and *Skewness*. Another issue of concern in our study was how many constraints are used by the authors in the formulation of MOEAs for Portfolio Management models. We found that the majority of the scholars used *two constraints*. Additionally, we found that the most widely used constraints are the *Cardinality constraints* followed by the *Lower and Upper bounds*, *Transaction Roundlots* and *Sector or Class capitalization constraints*. The next question we had to answer was which disciplines have contributed the most to the development of MOEAs focusing on the Portfolio Management. We found out that the two dominant disciplines are those of *Operations Research* and *Computer Science*. A final issue of concern in our analysis was to identify the methodological classification of choice for the authors of papers in MOEAs for the Portfolio Management. About 81.31% of the authors chose the *Combined* methodological approach, and a huge majority, about 95.95% of the authors who chose the *Combined* methodological approach selected a combination of *Model design* and *Empirical–Experiment* methodological approach.

4.4 Future Research

The use of Multiobjective Evolutionary Algorithms (MOEAs) for the portfolio selection is still in its early stages. The analysis of the problem's formulation revealed areas of concern with regard to MOEAs for the Portfolio Management. Specifically, it seems that the majority of the authors focus on the computational aspects of the problem and overlook its financial aspects. This become evident from the wide use of variance as the quantification risk measure of preference among the authors, although its well known undesirable properties such as the lack of coherence [10] and penalization of both returns above and below expected return. Thus, a potential path of future research would be the incorporation in the problem formulation of risk quantification measures with better mathematical properties. Another issue of concern that arises from our analysis is the number of constraints used by the authors in the problem formulation. We found out that the majority of the studies adopt only two constraints in the problem formulation. However, it is well known that the real world portfolio managers deal with multiple constraints. That way the models loose in interpretive power and their usefulness restricted only for academic purposes. The incorporation of additional real world constraints is a challenge that remains to be answered by the authors of papers in MOEAs for the portfolio management. Above we identified several areas of concern with regard to the formulation of the MOEAs for the portfolio management problem. In order to be able to address these issues further research is required towards the creation of new algorithmic frameworks. Currently the Pareto–dominance based framework is the most widely accepted. However, some studies (Purshouse [176]) indicate that traditional Pareto ranking schemes do not behave well in the presence of many objectives (where, "many" they mean more than 3-4). Moreover further research is required towards the incorporation of preferences in the portfolio selection problem. The decision making process involved in portfolio selection is very complex and difficult to automate. Incorporation of preferences in use with MOEAs remains a major challenge. Finally the creation of interactive MOEAs with the decision maker for the portfolio selection problem consists another potential area for future research. Above, we identified some possible paths for future research in Multiobjective Evolutionary Algorithms (MOEAs) for the Portfolio Selection. It is evident that the field presents significant research opportunities both for academics and practitioners interested in those innovative technologies.

Chapter 5

Best Practices and Performance Metrics in the construction of Efficient Portfolios with the Use of Multiobjective Evolutionary Algorithms

5.1 Introduction

The usefulness of the Multiobjective Evolutionary Algorithms (MOEAs) for the solution of the constrained portfolio optimization problem is indisputable and can be testified by a considerable number of studies in the field [87]. Numerous techniques have been devised for handling the difficulties arisen by the complexities of the constrained portfolio optimization problem. Most of the authors in the field attempt to provide evidence about the efficacy of the proposed algorithms and techniques with the assistance of experimental comparative analysis. However, the vast majority of the studies examines only some aspects of the constrained portfolio optimization problem and does not seem to capture the entire picture of the field. Our aspiration is to cover this gap in the relevant literature, by examining the state-of-the-art in order to provide the best practices in the field with respect to the constrained portfolio optimization problem with the support of MOEAs from a technical and algorithmic point of view.

The remainder of the chapter is organized as follows. Section 5.2 provides an introduction to the constrained portfolio optimization problem with the support of MOEAs. Section 5.3 examines a wide range of real world constraints imposed to the solution of the constrained portfolio optimization problem and presents their corresponding algorithmic and technical treatment. Section 5.4 presents the best practices in the field with regard to a number of configuration issues related to the application of MOEAs for solving the constrained portfolio optimization problem. Section 5.5 present the most popular performance metrics along with their corresponding strengths and weaknesses for the evaluation of the derived solutions.

Based on our findings, section 5.6 identifies some potential paths for future research. Finally, section 5.7, concludes the chapter.

5.2 The Portfolio Optimization problem with the support of MOEAs

The application of MOEAs techniques for solving the portfolio optimization problem has witnesses a considerable increase over the past years [19], [86], [123], [137], [143], [147], [160], [204], [209], [213], [224]. The turn toward evolutionary approaches for handling the difficulties introduced by the constrained portfolio optimization problem can be easily justified by a number of reasons. Among them, distinguished place possesses the ability of evolutionary algorithms (EAs) to solve difficult multiobjective optimization problems (MOP) that are too complex to be solved using deterministic techniques [7], [10], [33], [43], [95], [96], [97], [109], [177], [190], [194], [214]. The introduction of realistic constraints into the portfolio optimization problem convert the problem from a classical quadratic optimization problem to a quadratic mixed-integer problem (QMIP) that is NP-hard [159].

Despite, the obvious benefits from the application of evolutionary techniques for the solution of the portfolio optimization problem, there are a number of configuration issues that cause controversy in the field. In particular, there is not always unanimity about the best practices in the field. The purpose of this study is to cover this gap in the relevant literature and provide the best practices in the field by examining the state-of-the-art. For that purpose 220 journal and conference papers in the field have been analyzed and useful information about a number of configuration issues has been extracted.

Below, we provide in the form of graphs and tables the extracted intelligence by examining the state-of-the-art in the field. We start with Table 5.1 that presents which are the most popular MOEAs when it comes to the application of evolutionary techniques to the solution of the portfolio optimization problem.

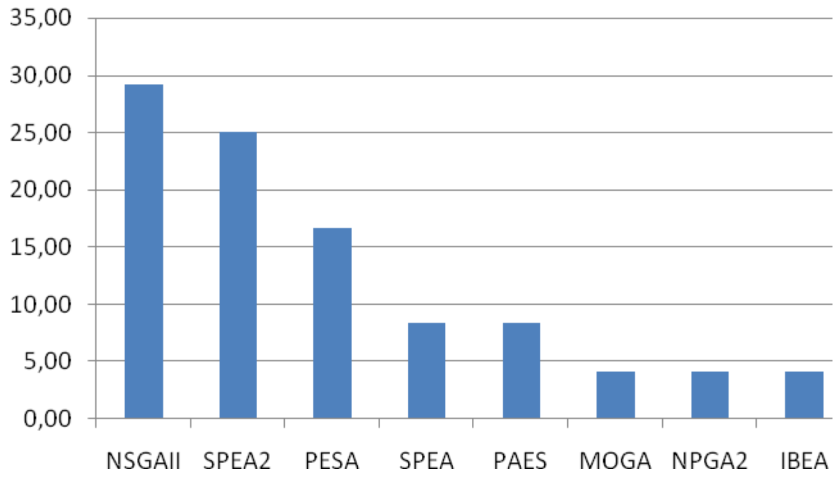


Figure 5.1 The most commonly used MOEAs for solving the portfolio optimization problem (%)

Table 5.1: The most commonly used MOEAs for solving the portfolio optimization problem (%)

MOEA type	NSGAII	SPEA2	PESA	SPEA	PAES	MOGA	NPGA2	IBEA
Percentage (%)	29.17	25.00	16.67	8.33	8.33	4.17	4.17	4.17

As shown from Table 5.1, NSGAII and SPEA2 are two most popular MOEAs with 29.17% and 25% respectively, when it comes to the application of evolutionary algorithm techniques for multiobjective optimization to the solution of the portfolio optimization problem. The purpose of the present study in not to provide a analytical account of each individual MOEA. For this purpose, the interested reader can advise the corrresponding papers: NSGAII [57], SPEA2 [231], PESA [45], SPEA [233], PAES [125], MOGA [93], NPGA2 [83], IBEA [230].

Please, notice that the aforementioned MOEAs cannot be used directly in their standard format for the solution of the constrainted portfolio optimization problem (CPOP), as special treatment is required for handling the various objectives and constraints imposed to the problem. In the following pages we will present the corresponding algorithmic and technical treatment for the efficient incorporation of the constraints into the MOEAs.

The CPOP in its standard format is a bi-objective problem (return and risk) and can be formulated as follows. Let Ω be the search space. Consider 2 objective functions f_1, f_2 where $f_i: \Omega \rightarrow R^m$ and

Optimize $f(w) = (f_1(w), f_2(w))$ (2.1)

Maximize portfolio return $f_1(w) = \sum_{i=1}^{m} w_i \overline{r_i}$ (2.2)

Minimize portfolio risk $f_2(w) = \sum_{i=1}^{m} \sum_{j=1}^{m} w_i w_j \sigma_i \sigma_j \rho_{ij}$ (2.3)

where:

- Decision variables $w = (w_1, .., w_m)$ subject to $w \subset \Omega$ and m equal to the number of stocks.
- Rate of return of assets: $r_1, r_2 .., r_m$.
- ρ_{ij} is the correlation between asset i and j and $-1 \leq \rho_{ij} \leq 1$.
- σ_i, σ_j represent the standard deviation of stocks returns i and j

From the formal presentation of the portfolio optimization problem it becomes clear that it is a bi-objective problem where the first objective corresponds to the *return of assets* and the second objective corresponds to the portfolio's risk which is a function of stocks' *standard deviation* and *correlation* alike. The higher the portfolio's *return* the better and the lower the portfolio's *risk* the better.

5.3 Algorithmic and technical treatment of the constraints imposed to the Portfolio Optimization problem in the context of MOEAs

A number of constraints can be imposed to the aforementioned bi-objective problem [139]. By examining the available literature in the field we found out that the following real world constraints have been considered by the studies in the field.

1. *Budget or summation constraint*
2. *Floor and ceiling constraint*
3. *Cardinality constraint*
4. *Class constraints*
5. *Roundlots constraints*
6. *Transaction costs*
7. *Pre-assignment constraint*
8. *Trading constraints*
9. *Turnover constraint*
10. *The 5-10-40 constraint*

According to a study by Metaxiotis K. and Liagkouras K. [155] the 62.07% of the studies in the field make use of only two constraints. Also, according to the same study the 20.69% of the studies in the field make use of three constraints and only the 6.90% of the studies in the field make use of four constraints.

The successful implementation of the aforementioned constraints by the MOEAs requires special algorithmic and technical configurations. The scholars in the field realized the need of introducing a number of *repair mechanisms* for the efficient implementation of the CPOP by the MOEAs. Table 5.2 summarizes the most commonly used repair mechanisms by the studies in the field.

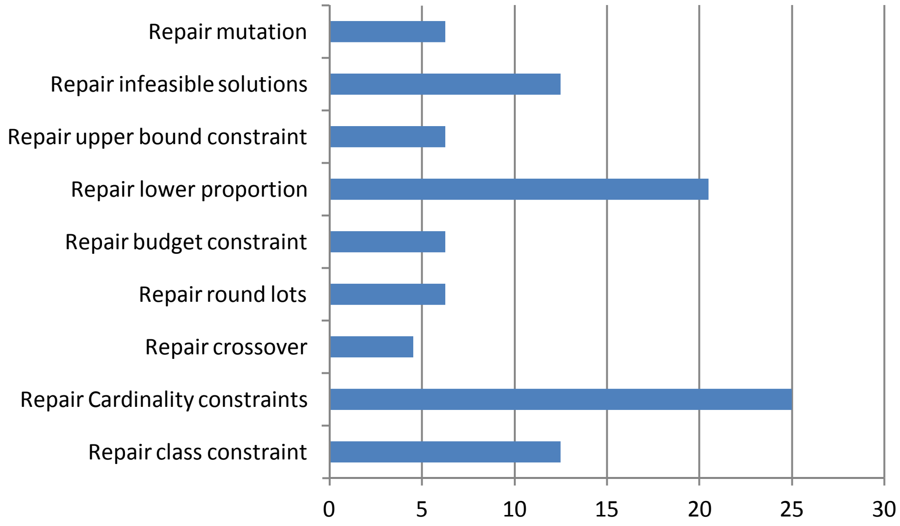


Figure 5.2. The most commonly used repair mechanisms by the studies in the field (%)

Table 5.2: The most commonly used repair mechanisms by the studies in the field (%).

Repair mechanism	Percentage(%)
Repair mutation	6.25
Repair infeasible solutions	12.5
Repair upper bound constraint	6.25
Repair lower proportion	20.5
Repair budget constraint	6.25
Repair round lots	6.25
Repair crossover	4.5
Repair Cardinality constraints	25
Repair class constraint	12.5

Below, by examining the state-of-the-art we provide rigorous algorithmic and technical treatment for the correct implementation of the various constraints imposed to the CPOP.

Solution representation

A solution to the portfolio optimization problem can be represented by an array of m variables $w_1, w_2, \ldots, w_m$ where w_i represents the fraction of the amount invested in asset i. Thus the $\mathbf{w} = (w_1 , w_2 , \ldots, w_m)$ could be a hypothetical solution vector of the CPOP.

5.3.1 Budget constraint

Budget constraint or summation constraint $\sum_{i=1}^{m} w_i = 1$ requires all portfolios to have non-negative weights ($0 \leq w_i \leq 1 \;\; i = 1,2,..,m$) that sum to 1. We should highlight that the Budget constraint is also included in the Modern Portfolio Theory (MPT) as presented by Markowitz [151]. Non-negative weights means that short sales are not allowed.

In case that we had to take into account only the budget constraint it would be enough to generate random decision variables in the range $w_i \in \{0, 1\}$. Then we could easily proceed with the normalization as follows: $w_i^{'} = \frac{w_i}{\sum_{j=1}^{m} w_j}$, where w_i represents the actual weight of asset *i* and $w_i^{'}$ the normalized weight.

5.3.2 Floor and ceiling constraint

Floor and ceiling constraint $a_i \leq w_i \leq b_i, \forall = 1,2,..., m$, where α_i is the minimum weighting that can be held of asset *i* (*i* = *1, ..., m*), b_i = the maximum weighting that can be held of asset *i* (*i* = *1, ...,m*) and $0 \leq a_i \leq b_i \leq 1 \;\; , \forall = 1,2,..., m$. *Ceiling constraints* are used in order to avoid excessive exposure to a specific asset and in many cases are imposed by law. Floor constraints on the other hand are used to avoid the cost of administrating very small portions of assets.

Schaerf [178] introduced an improved version of the normalization mechanism capable of satisfying the floor and ceiling constraints too, as shown by the following formula: $w_i^{'} = \varepsilon_i + \frac{w_i - \varepsilon_i}{\sum_{j=1}^{m} w_j - \varepsilon_i}$, where ε_i is the floor constraint.

Chiam et al. [34] proposed a different methodology for handling the floor and ceiling constraints. Specifically, the weight vector is adjusted to the floor and ceiling constraint as follows: $w_i^{'} = \; a_i + (b_i - a_i) \times w_i$, where a_i and b_i denotes respectively the minimum and maximum weighting that can be held of asset *i*. According to [34], assets will be added to the portfolio until the total weight of the portfolio exceeds one. At this point three (3) different cases have been identified:

1. After removing the last added asset, the remaining weight is between the floor and ceiling limits. In that case the weight of the last added asset is adjusted to be equivelant to the remainder needed to attain the portfolio total weight of one.
2. After removing the last added asset, the remaining weight is less than the floor limit. In this case there are two (2) different scenarios:

   2.a After removing the second last added asset, the remaining weight is between the floor and ceiling limits. In that case the weight of the second last added asset is adjusted to be equivelant to the remainder needed to attain the portfolio total weight of one. Obviously, to the last asset is assigned a weight of zero.

   2.b After removing the second last added asset, the remaining weight is outside the floor and ceiling limits. In that case, all the weight vectors will be readjusted by either increasing or decreasing them by a predefined percentage.

According to Mishra et al. [158] the normalization of the weights is performed as follows: $w_i^{'} = \frac{w_i z_i}{\sum_{i=1}^{m} w_i z_i}$.

In case that we take into account the floor and ceiling constraints the formula is modified as follows: $w_i^{'} = a_i z_i + \frac{w_i z_i}{\sum_{i=1}^{m} w_i z_i}(b_i z_i - \sum_{i=1}^{m} a_i z_i)$, where $z_i = \{0, 1\}$ and $a_i \leq w_i \leq b_i, \forall\, i = 1,2, \dots, m$.

However, If we take into consideration only the floor constraint and there is no restriction on the upper limit the formula is modified as follows:
$w_i^{'} = a_i z_i + \frac{w_i z_i}{\sum_{i=1}^{m} w_i z_i}(1 - \sum_{i=1}^{m} a_i z_i),\ i = 1, \dots, m.$

Finally, if we take into consideration only the ceiling constraint and there is no restriction on the lower limit the formula is modified as follows: $w_i^{'} = b_i z_i - \frac{w_i z_i}{\sum_{i=1}^{m} w_i z_i}(b_i z_i),\ i = 1, \dots, m$

5.3.3 Cardinality constraint

Cardinality constraint $C_{\min} \leq \sum_{i=1}^{m} q_i \leq C_{\max}$, where $C_{\min}$ is the minimum number of assets that a portfolio can hold, C_{max} is the maximum number of assets that a portfolio can hold, $q_i = 1$, for $w_i > 0$ and $q_i = 0$, for $w_i = 0$. In the relevant literature and in business practice the cardinality constraint is expressed in various forms. For instance, in certain cases the number of assets in the portfolio is set to a given value and in some other cases is bounded between a lower and upper value. The cardinality constraint is used to facilitate the portfolio management and to reduce its management costs [69].

Streichert et al. [200] proposed a repair mechanism capable of handling cardinality constraints. A real-valued genotype is applied to represent the solution vector. In case that cardinality constraints are considered in the model, only the K largest values of w_i are kept before applying the normalization. To the rest of the decision variables w_i is assigned a zero value.

A number of papers examine the cardinality constraint (CC) but in a simplified form where the inequality restriction is replaced by an equality restriction instead [8], [202], [159]. However Chiam et al. [34] proposed a repair mechanism for the cardinality constraint (CC) in its generic format as it is expressed with the assistance of inequality restrictions by the following relation $C_{\min} \leq \sum_{i=1}^{m} q_i \leq C_{\max}$, where C_{min} is the minimum number of assets that a portfolio can hold, C_{max} is the maximum number of assets that a portfolio can hold, $q_i = 1$, for $w_i > 0$ and $q_i = 0$, for $w_i = 0$. The repair operator for the CC is described in Fig.3

```
IF number of asset > maximum cardinal
        Increase all weights by k%
else IF number of asset < minimum cardinal
        Decrease all weights by k%
end IF
```

Figure 5.3. Pseudo code for the repair operator

The CC repair mechanism as described in Fig. 5.3 increases the weights in the solution vector when its associated portfolio size exceeds the maximum cardinality and respectively decreases the weight in the solution vector when its associated portfolio size is smaller than the minimum cardinality. According to the authors [34] this repair mechanism is able to adjust the portfolio sizes of infeasible vector solutions back to the feasible range.

Budget, floor and ceiling constraints and cardinality constraints are the most commonly used constraints by the studies in the field. These constraints are considered hard constraints [201] that means that they must not violated. On top of the aforementioned constraints there is a number of additional real world constraints like the class or sector constraints, roundlots constraints, transaction costs constraints, pre-assignment constraints, trading constraints, turnover constraints and the 5-10-40 constraint that are less often addressed by the studies in the field [155]. Below we provide a presentation of these constraints:

5.3.4 Class constraints

Class constraints [28] limit the exposure of the portfolio to assets with common characteristic. Let Γ_m , *m = 1, ... , M, be M sets of assets that are mutually exclusive i.e.* $\Gamma_i \cap \Gamma_j = \emptyset, \forall\, i \neq j$. Class constraints limit the proportion of the portfolio that can be invested in assets in each class. Let $L_m$be the lower proportion limit and U_m be the upper proportion limit for class *m* then the class constraints are: $L_m \leq \sum_{i \in \Gamma_m} w_i \leq U_m$, *m = 1, ... , M.*

Anagnostopoulos K. and Mamanis G. [5] proposed a method for handling the class constraint. Specifically, they introduced a vector of real values that represents the proportion of the initial budget that should be invested in each class.

$$\mathrm{C} = (c_1\ , c_2, \dots, c_T), 0 \leq c_t \leq 1, \quad t = 1, \dots, T$$

The class weights *c* are associated with each group of assets *t* and they are normalized to find the proportion invested in assets belonging to the particular class by using the formula:

$$rcp(t) = L_t + \frac{c_t}{\sum_{j=1}^{T} c_j}\left(1 - \sum_{j=1}^{T} L_j\right), \quad t = 1, \dots, T$$

Where *rcp(t)* is the real class proportion invested in class *t*. Next, the real proportion of each class is shared in the corresponding assets of the particular class. Also, L_t is the lower proportion limit and U_t be the upper proportion limit for class *t*. The proportion associated with each asset in the portfolio is estimated as follows:

$$x_i = l_i\delta_i + \frac{w_i\delta_i}{\sum_{j\in G_{class(i)}} w_j\delta_j}\left(rcp(class(i)) - \sum_{j\in G_{class(i)}} l_j\,\delta_j\right), i = 1, \dots, n \qquad (5.1)$$

where *class(i)* returns the group that the asset *i* belongs.

According to the authors [5] the aforementioned methodology for handling the class constraint occasionally can lead to infeasible portfolios. In particular, in certain cases the portfolio does not contain an asset from a specific class, which must have a positive proportion weight in the portfolio. In this case the *rcp* of the particular class does not share and the resulting solution violates budget as well as one class constraint. They propose a repair mechanism for the infeasible solution, where a random asset is inserted from a specific class into the portfolio, simply by changing a zero value to 1 in the *Δ* string.

There is another case where the class constraint can lead to infeasible portfolios. Specifically, in certain cases the parenthesis in Eq.(5.1) is negative. This can occur when the *rcp* for a particular class is not enough to be shared in the corresponding assets, so that to satisfy lower quantity constraints for these assets. The proposed repair mechanism deletes surplus assets that belong to the particular class, by changing the value from 1 to 0 in the *Δ* string starting from the one with the smallest weight.

The aforementioned repair mechanism is able to satisfy the lower class bound. For the upper class bound is applied the following repair mechanism proposed by Chang et al. [28]. In particular, if a real class proportion *rcp(t)* violates its upper limit U_t, then the real class proportion for the particular class *t* takes the U_t value and the remaining class values are normalized as shown above. The same process is followed for the real asset weights.

5.3.5 Roundlots

Roundlots [201] give the smallest volumes c_i that can be purchased for each asset, $w_i = y_i \cdot c_i$, $i = 1, \dots, N$ and $y_i \in \mathbb{Z}$. In certain markers the securities are negotiated as multiples of minimum lots. The Roundlots constraint is used in order to represent the fact that each stock in certain markets is negotiated as multiple of a minimum tradable lot.

Streichert et al. [200] proposed a repair mechanism capable of handling cardinality, floor constraint and minimum roundlots. According to [200], the algorithm rounds w_i to the next roundlot level: $w_i^{'} = w_i - (w_i \; mod \; c_i)$. The remainder of the rounding process $\sum_i (w_i \; mod \; c_i)$ is spend in quantities of c_i on those $w_i^{'}$ which had the biggest values for $w_i \; mod \; c_i$ until all of the remainder is spent.

Chiam et al. [35] provides a practical way of handling the roundlot constraint. According to [35] the roundlot constraint is expressed by the following relationship: $w_i = \frac{c_i y_i}{C}$, where C is the total capital budget, c_i is the purchasing price for the minimum lot of asset i and $y_i \in \mathbb{Z}$ denotes the number of lots purchased for asset i. The inclusion of roundlot constraint into the CPOP may require relaxation of the budget constraint, as the total capital is very likely not to be exact multiples of the minimum lot prices for the various assets [35].

According to [35] every asset that is added into the portfolio will first be adjusted based on the floor and ceiling constraints, as shown above. Then, they can be rounded down to the largest weight available according to the: $w_i^{''} = w_i^{'} - w_i^{'} \cdot mod(\frac{c_i}{C})$. The remainder of the budget will be allocated to the assets in the existing portfolio provided that the ceiling constraint is not violated and also to assets outside the portfolio with the condition that the floor constraint will be satisfied.

5.3.6 Transaction costs

Transaction cost can be classified into two types: fixed and variable costs. Fixed costs are paid on all transactions irrespective of the volume of the transaction. While the variable costs are dependent on the transaction volume.

According to Woodside-Oriakhi et al. [217] the transaction cost can be represented with the assistance of equation $\sum_{i=1}^{m} (v_i^b y_i^b + v_i^s y_i^s + f_i^b a_i^b + f_i^s a_i^s) \leq D$ which limits the total transaction cost incurred.

where:

v_i^b is the variable transaction cost (≥ 0) for each unit of asset *i* that is bought.

v_i^s is the variable transaction cost (≥ 0) for each unit of asset *i* that is sold.

y_i^b the number of units (≥ 0) of asset *i* bought.

y_i^s the number of units (≥ 0) of asset *i* sold.

f_i^b is the fixed transaction cost (≥ 0) paid if we carry out any buying of asset i.

f_i^s is the fixed transaction cost (≥ 0) paid if we carry out any selling of asset *i*.

a_i^b 1 if we buy any of asset i, 0 otherwise.

a_i^s 1 if we sell any of asset i, 0 otherwise.

The $v_i^b y_i^b + v_i^s y_i^s$ is the variable cost associated with trading asset i and $f_i^b a_i^b + f_i^s a_i^s$ is the fixed cost associated with trading asset i. Please note that the variable cost correspond to a *V*-shaped function. The transaction costs constraint is usually considered when we want to incorporate into the CPOP the re-optimization of the portfolio of stocks successively over time. The purpose of the transaction costs constraint is to limit the cost associated with buying and selling stocks during the re-optimization process.

5.3.7 Pre-assignment constraint

In certain cases, usually after investor's request some specific assets should be included in the portfolio in predetermined proportions. This constraint can be imposed by setting predetermined value for the minimum and maximum weighting, α_i and b_i values for those particular assets.

According to Di Gaspero et al. [68] given a set of assets S = $\{s_1, \ldots, s_n\}$ and a set of binary decision variables *Z* such that $z_i = 1$ if and only if asset i is in the solution (i.e. $w_i > 0$), the pre-assignment constraint can be expressed by a binary vector *P* such that $p_i = 1$ if and only if s_i is pre-assigned in the portfolio. Where $z_i \geq p_i$, $i = 1, 2, \ldots, n$ Please notice that if the number of pre-assigned assets $\sum_{i=1}^{n} p_i$ is greater than the minimum cardinality (C_{min}), then C_{min} is implicitly set to it.

5.3.8 Trading constraints

In the multi-period formulation of the CPOP the Trading constraints can be useful for imposing minimum limits on buying and selling of very small quantities of assets as fixed costs can be disproportional high compared to the value of the transaction [69]. According to Chiam S.C. et al. [35] the trading constraints can be expressed as follows:

$$w_i = 0 \; or \; w_i \geq w_i^0 + \underline{B_i}$$

$$w_i = 0 \; or \; w_i \leq w_i^0 - \underline{S_i}$$

Where w_i^0 represents the holding proportion of asset i at zero period and $\underline{B_i}$ and $\underline{S_i}$ denotes respectively the minimum purchase and sale of asset *i* at period 1.

5.3.9 Turnover constraint

According to Scherer et al. [182] the turnover constraints are trading limits designed to protect from executing a market order at a worse rate than originally set in the order. Moreover, according to Chiam S.C. et al. [35] the main difference between the trading constraints and the turnover constraints is that the first impose minimum limits on buying and selling assets while the second impose upper bounds for assets variation between trading periods. According to Chiam S.C. et al. [35] the turnover constraints can be expressed as follows:

$$max(w_i - w_i^0, 0) \leq \overline{B_i}$$

$$max(w_i^0 - w_i, 0) \leq \overline{S_i}$$

Where w_i^0 represents the holding proportion of asset i at zero period and $\overline{B_i}$ and $\overline{S_i}$ denotes respectively the maximum purchase and sale of asset *i* at period 1.

5.3.10 The 5-10-40 constraint

The 5-10-40 constraint is based on the German investment law. This law impose that a fund may invest no more than 5% of its value in securities of the same issuer. However, this limit can be increased to 10% provided that the total value of any holdings between 5% and 10% does not exceed 40% of the fund.

Branke et al. [24] proposed a repair mechanism for the 5-10-40 constraint that is analysed in seven (7) steps as follows:

1. For all i with $z_i = 0$, set $w_i = 0$.

2. The vector $\boldsymbol{w}$ is normalized $w_i^{'} = \frac{w_i}{\sum_{i=1}^{m} w_i}$

3. The surplus amount exceeding the 10% threshold is calculated, $\Omega = \sum_{w_i > 0.1}(w_i - 0.1)$. All $w_i > 0.1$ are set to 0.1.

4. The surplus S has to be redistributed to the weights below 10%. Each weight below 10% is increased by the amount $(0.1 - w_i) \cdot \frac{\Omega}{\Psi}$, where $\Psi = \sum_{0 < w_i < 0.1}(0.1 - w_i)$. Since $\Psi < \Omega$, no weight will have a value above 10% after the first step.

5. In the group with more than 5% we only accept the assets with the largest weights such that the sum is still less than or equal to 40%. All others are capped to 5% and the excess weight is distributed to the other assets analogously to the above step.

6. In case that there is not enough room for all the excess weight to be redistributed among the assets with less than 5%, the remaining is used to fill up the assets between 5% and 10% up to 10% in order of decreasing weight.

7. The previous step may lead to a violation of the 5-10-40 constraint. In this case, assets are removed from the 5-10% group in order of increasing weights, and the weight in excess of 5% is distributed to the other assets in the 5-10% group as shown to the previous step. The 5-10-40 rule implicitly signifies that any feasible portfolio for the CPOP with 5-10-40 constraint will contain at least 16 different assets, 4 times 10% plus 12 times 5%.

5.4 Analysis of various MOEAs' configuration issues for efficient solution of the constrained portfolio optimization problem

As we have already highlighted the MOEAs cannot be used directly in their standard format for the solution of the constrainted portfolio optimization problem (CPOP), as special treatment is required with regard to a number of issues like the encoding type, the genetic operators and the objectives' handling. Below, we provide an analytical account of the best practices in the field as they emerge by examining the state-of-the-art.

5.4.1 Encoding types

The choice of the most suitable technique for the solution representation is of upmost importance as has serious impact on the performance and functionality of the entire MOEA. Table 3 presents the types of encoding for the representation of the solution vector and their corresponding popularity as it is indicated by the studies in the field.

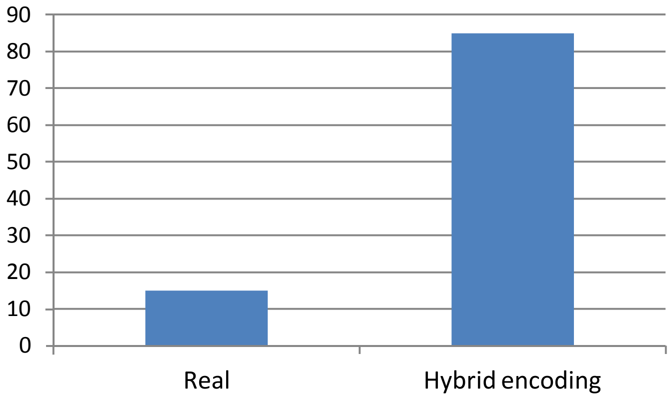


Figure 5.4. The most commonly used types of encoding for representing the solution vectors by the studies in the field (%)

Table 5.3: The most commonly used types of encoding for representing the solution vectors by the studies in the field (%)

Encoding Type	Percentage (%)
Real	15
Hybrid binary-real valued encoding	85

Among them, the hybrid representation proposed by Streichert et al. [200] is the most popular type of encoding for solution vector representation, as 85% of the studies in the field make use of this encoding type [24], [28], [35], [36], [78], [121], [140], [158], [200]. In the hybrid representation two vectors are used for defining a portfolio: A binary vector that specifies whether a particular asset participates in the portfolio and a real-valued vector used to compute the proportions of the budget invested in the assets.

$$\Delta = \{z_1, \dots, z_n\}, \quad z_i = \{0, 1\}, \quad i = 1, \dots, n$$
$$W = \{w_1, \dots, w_n\}, \quad 0 \leq w_i \leq 1, \quad i = 1, \dots, n$$

Moreover through the hybrid encoding a number of constraints like for example the cardinality constraint can be satisfied. For instance, if the number of 1's in Δ overcomes the maximum cardinality, then those assets that have the minimum weight in $W$are deleted, by changing its value from 1 to 0 in Δ.

According to Streichert et al. [200] the usage of hybrid encoding for the solution representation facilitates the process of adding or removing an asset to the portfolio by simply mutating the bit-string Δ. Finally, the hybrid encoding is altered by mutating and crossing each genotype W and Δ separately from each other.

The 15% of the studies in the field use real encoding for the solution vector representation. In the real encoding solution representation, the binary vector (Δ) is replaced by a mechanism that ensures if an asset is selected to be included in portfolio is assigned a value of 1 otherwise 0.

Analytically, according to Gupta et al. [105] the real coded genetic algorithm (RCGA) is applied as follows: If the ith asset has been selected to be included in the portfolio then assign $y_i = 1$ and randomly generate $w_i \in [l_i y_i,\ u_i y_i]$, i.e. $w_i \in [l_i,\ u_i]$; otherwise assign $y_i = 0$ and thus $w_i = 0$.

5.4.2 Mutation operators

The mutation operator is being used by the MOEAs to maintain the diversity in the population and thereby avoid stagnation at a local optimum [15]. Table 5.4 presents various types of mutation operators that have been applied for solving the CPOP and their corresponding popularity as it is indicated by the studies in the field. As shown in Table 4 the most popular mutation operators when it comes to the solution of the CPOP are the Bit-flip mutation with 46.67% and the Gaussian random mutation with 33.33% respectively.

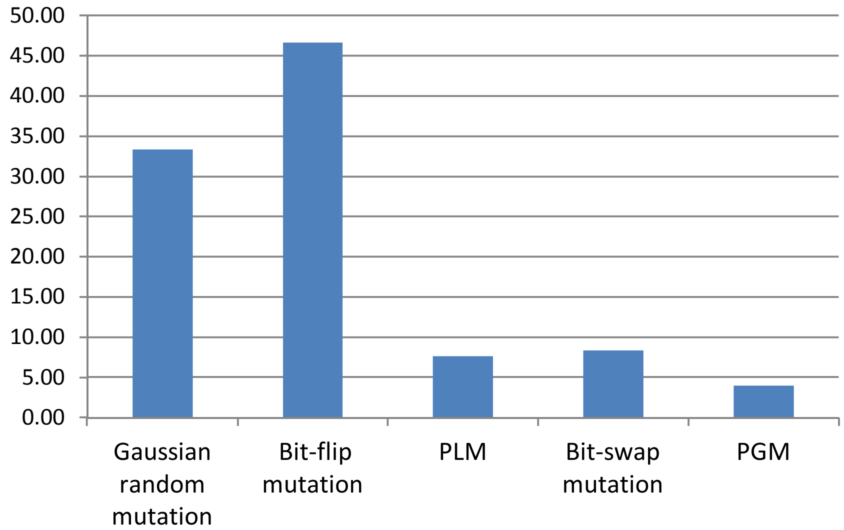


Figure 5.5. The most commonly used mutation operators by the studies in the field.

Table 5.4: The most commonly used mutation operators by the studies in the field.

Mutation Operator	Percentage (%)
Gaussian random mutation	33.33
Bit flip mutation	46.67
PLM	7.67
Bit-swap mutation	8.33
PGM	4.00

There is an interrelation between the encoding type and the selection of the appropriate mutation operator. For instance, we found out that a considerable percentage of the studies that use Hybrid encoding for the solution representation, apply Gaussian random mutation [5], [6], [24], [36] for the real-valued vectors and

Bit-flip mutation [5], [6], [24], [36], [76], [200], [201] for the binary-valued vectors. Below, we provide a short description of the mutation operators as appeared in Table 4.

We start with the Gaussian Mutation operator (M_G). Let $x \in [a,\ b]$ be a real variable. Then, the Gaussian mutation operator changes x to: $M_G(x) := \min(\max(N(x,\sigma),a),b)$.
Where σ may depend on the length of the interval $l := b - a$. The M_G is applied with probability p_m to each variable. The value of σ also depends on the number of the current generation. In particular, σ decreases with time, as at the beginning a stronger mutation supports the sampling of the search space and smaller displacements towards the end aid in fine tuning extreme values. Gaussian mutation considered suitable for local search.

The flip-bit mutation is being used for binary-valued vectors and reverses the value of a randomly selected gene as shown below:

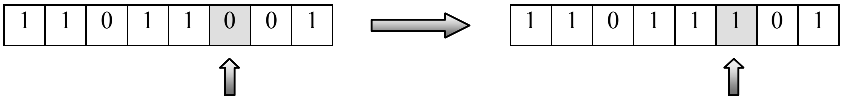


Figure 5.6. The Flip-bit mutation

The bit-swap mutation operator [35] is mainly used for order-based representation and allows the swapping between the assets and weights of two randomly selected alleles in a single solution vector as shown below:

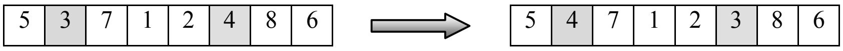


Figure 5.7. The Bit-swap mutation

The position of randomly chosen genes (assets in this case) is swapped. According to Chiam et al. [35] in bit-swap mutation in order to prevent neutral variation, it should be taken care that at least one of the selected assets should be within the portfolio (i.e. $w_i > 0$).

In polynomial mutation (PLM) [61] each decision variable x_i, can take values in the interval: $x_i^{(L)} \leq x_i \leq x_i^{(U)}, i = 1, 2, \dots, n$. Where $x_i^{(L)}$ and $x_i^{(U)}$ stand respectively for the lower and upper bounds for the decision variable i. Moreover, each decision variable has a probability P_m to be perturbed. For each decision variable, a random value *rand* is drawn. If $rand \leq P_m$ then, a mutated variable obtains its new value. If random value is $r \leq 0.5$ it samples to the region between X_{Low} and X_i, otherwise if $r > 0.5$ it samples to the region between X_i and X_{Upper}. The algorithm also calculates the δ_q value to be used in getting the variable its new value. According to Deb & Tiwari [61] one of the main advantages of polynomial mutation is that it allows us to sample the entire search space of the decision variable even though the value to be mutated is close to one of the boundaries (X_{Low} - X_{Upper}).

Liagkouras and Metaxiotis [140] presented the probe guided mutation (PGM) operator and its application for solving the CPOP. The proposed mutation operator incorporates a fitness functions evaluation mechanism that allows the evaluation of the corresponding fitness for the left hand side region and the right hand side region of the parent solution. The selection between the two alternative child solutions is done with the assistance of the Pareto optimality conditions. According to the authors, thanks to the PGM operator the algorithm is able to move fast towards the higher fitness regions of the search landscape and discover near optimal solutions.

5.4.3 Recombination operators

The recombination operator is responsible for the efficient exploration of the available search space. Table 5.5 presents a number of recombination operators and their corresponding popularity for solving the CPOP as it is indicated by the relevant studies in the field. As shown in Table 5, the most popular crossover operators when it comes to the solution of the CPOP are the Uniform crossover with 40.91% and the Single point crossover with 22.73% respectively.

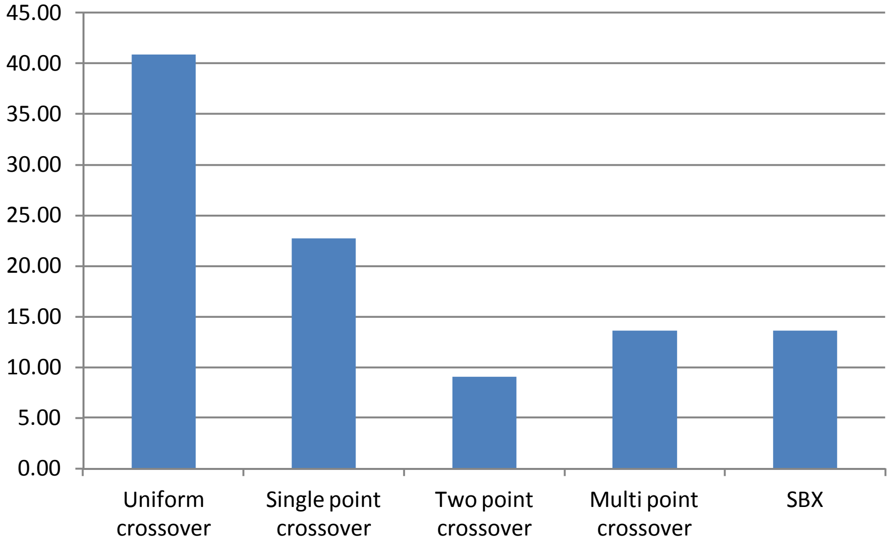


Figure 5.8. The most commonly used crossover operators by the studies in the field.

Table 5.5: The most commonly used crossover operators by the studies in the field.

Crossover Operator	Percentage (%)
Uniform crossover	40.91
Single point crossover	22.73
Two point crossover	9.09
Multi point crossover	13.64
SBX	13.64

In Uniform crossover [4], [5], [28], [30], [34], [45], [158], [169] two parents generate a single child. According to Chang et al. [28] if an asset is present in both parents it is present in the child. If an asset is present in only one parent it has probability 0.5 of being present in the child. Figure 5.9 provides an example of the uniform crossover.

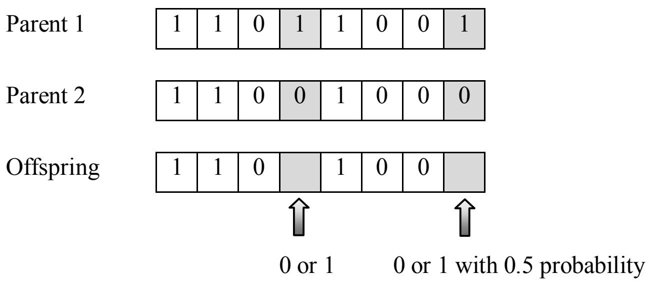


Figure 5.9. The Uniform crossover

In Single-point crossover [36], [184] randomly is selected one crossover point and then is copied everything before this point from *Parent 1* and then everything after the crossover point is copied from *Parent 2*. Figure 5.10 provides an example of Single-point crossover.

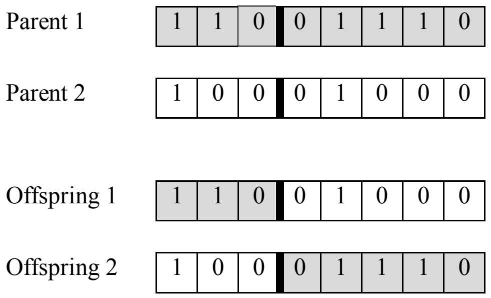


Figure 5.10. Single-point crossover

The Two-point crossover [76] is similar to single-point crossover except that two cut-points are randomly generated instead of one. Figure 5.11 provides an example of Two-point crossover.

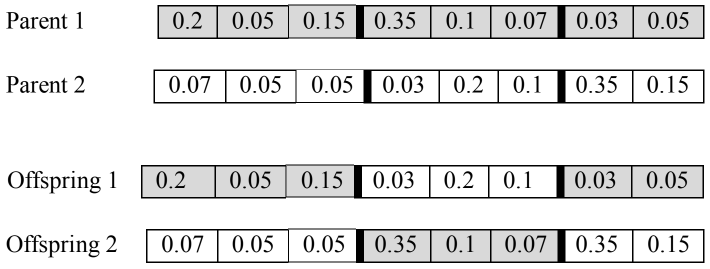


Figure 5.11. Two-point crossover

As expected after compliting the crossover process a repair mechanism is applied to the offspring solutions to satisfy the budget constraint as follows $w_i^{'} = \frac{w_i}{\sum_{j=1}^{m} w_j}$. Similarly to the Two-point crossover is applied the Multi-point crossover with only difference that instead of two crossover points in this case we have three or more crossover points.

The simulated binary crossover (SBX) operator [54] uses a probability distribution around two parents to create two children solutions. Each decision variable x_i, can take values in the interval: $x_i^{(L)} \leq x_i \leq x_i^{(U)}, i = 1, 2, \dots, n$. Where $x_i^{(L)}$ and $x_i^{(U)}$ stand respectively for the lower and upper bounds for the decision variable i. In SBX, two parent solutions $y^{(1)}$ and $y^{(2)}$ generate two children solutions $c^{(1)}$ and $c^{(2)}$ as follows:

$$c^{(1)} = 0.5\left[\left(y^{(1)} + y^{(2)}\right) - \beta_q |y^{(2)} - y^{(1)}|\right]$$
$$c^{(2)} = 0.5\left[\left(y^{(1)} + y^{(2)}\right) + \beta_q |y^{(2)} - y^{(1)}|\right]$$

Create a random number u between 0 and 1. $u \longrightarrow [0, 1]$;

where $\beta_q = \begin{cases} (au)^{1/(\eta_c+1)} & if\ u \leq \frac{1}{a}, \\ \left(\frac{1}{2-au}\right)^{1/(\eta_c+1)} & otherwise \end{cases}$

$\alpha = 2 - \beta^{-(\eta_c+1)}$ and

$$\beta = 1 + \frac{2}{y^{(2)} - y^{(1)}} min\left[(y^{(1)} - y^{(l)}), (y^{(u)} - y^{(2)})\right]$$

5.4.4 Selection operators

The selection operator is responsible of choosing the fittest chromosomes to evolve to the next generation. As shown from table 6 Tournament selection is by far the most popular selection operator as 95% of the studies in the field make use of this selection operator.

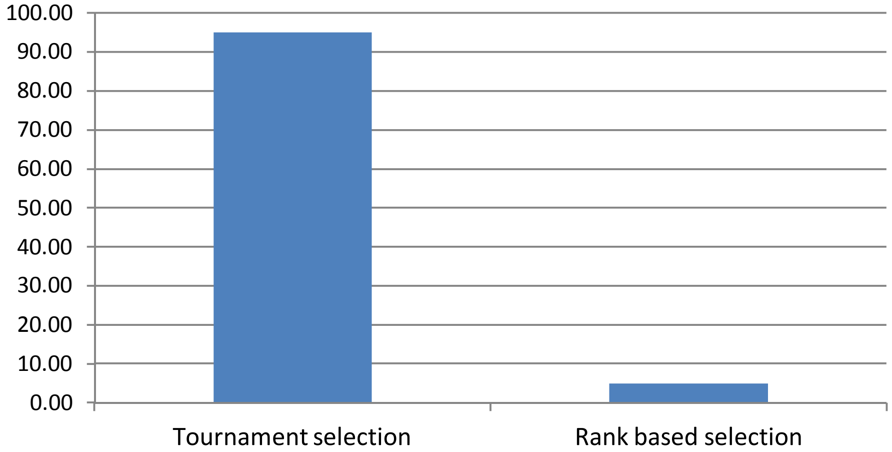


Figure 5.12. The most commonly used selection operators by the studies in the field (%).

Table 5.6: The most commonly used selection operators by the studies in the field (%).

Selection Operator	Percentage (%)
Tournament selection	95.00
Rank based selection	5.00

In tournament selection, *k* individuals are selected randomly from the population and compete against each other. The individual with the highest fitness is selected to be included to the next generation's population. The number of individuals competing in each tournament is referred to as tournament size. Binary (or pairwise) tournament selection (i.e. *k=2*) is the most popular tournament size among the studies in the field. We found out that about 85% of the total studies in the field make use of the binary tournament selection [4], [5], [24], [34], [36], [37], [140], [169]. As the tournament size increases (*k >2*) the selection pressure increases too. The selection pressure can be defined as the ratio between the population size and the number of individuals selected as parents of the next generation [157]. Thus, the selection pressure can be calculated as follows: $SP = 1 - \frac{|N_{parents}|}{|N_{popul}|}$, where $N_{parents}$ stands for the number of different selected parents and N_{popul} represents the population size. As selection pressure increases, the probability of making the wrong decision increases exponentially and the lack of diversity in the population may lead to premature convergence in local optima solutions [108].

Deb K. [52] suggested the use of binary tournament selection for efficient constraint handling. In particular he proposed the following criteria:

1. Any feasible solution is preferred to any infeasible solution.
2. Among two feasible solutions, the one having better objective function value is preferred.
3. Among two infeasible solutions, the one having smaller violation is preferred.

In another study, Chiam et al. [37] propose the binary tournament selection to be based on the Pareto ranking for the selection of the fittest and in the event of a tie the niche count to be employed.

The rank based selection is used by the 5% of the studies in the field. In the rank based selection the individuals are sorted from best to worst according to their fitness values. Thus, based on its rank, each individual i has a probability $p(i)$ of being selected. The $p(i)$ depends on a ranking function that can be of different types like for instance linear, exponential, power or geometric.

According to Goldberg and Deb [101] the tournament selection is superior to rank based selection because the tournament selection is faster than sorting the population to assign rank based probabilities. Although, both selection operators generate comparable results in terms of rate of convergence towards the optimum solution, the time complexity associated with the rank based selection deters many researchers in the field from employing this selection operator in their studies.

5.5 Performance Metrics

In this section the focus is concentrated on the evaluation of the relevant results from the MOEAs' implementation to the solution of the CPOP. In particular, the evaluation of the relevant results is not straightforward due to the multiobjective nature of the portfolio optimization problem. However, according to Zitzler et al. [229] there are a number of indicators that assist us to evaluate the quality of the derived solutions:

a. The smaller the distance of the resulting non-dominated set from the Pareto optimal front the better.
b. The more uniform the distribution of the solutions found across the non-dominated front the better.
c. The greater the extension of the obtained non-dominated front the better.

Based on the aforementioned principles, a number of performance metrics for multiobjective problems have been proposed. Below, we provide a concise presentation of the most popular performance metrics [232] as these emerged by the studies in the field.

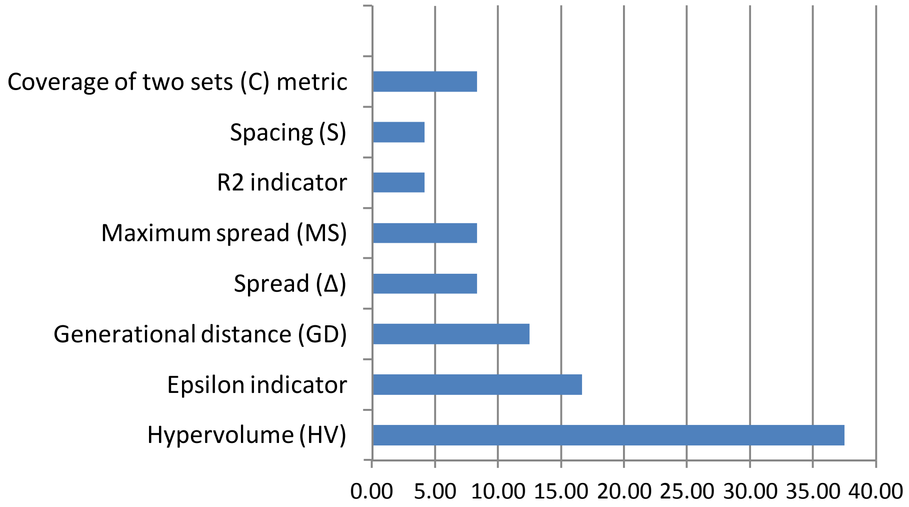


Figure 5.13. The most commonly used performance metrics by the studies in the field (%).

Table 5.7: The most commonly used performance metrics by the studies in the field (%).

Performance Metrics	**Percentage (%)**
Hypervolume (HV)	37.50
Epsilon indicator	16.67
Generational distance (GD)	12.50
Spread (Δ)	8.33
Maximum spread (MS)	8.33
R2 indicator	4.17
Spacing (S)	4.17
Coverage of two sets (C) metric	8.34

5.5.1 *Hypervolume*

As shown in Table 7 the most popular performance metric in the field is the Hypervolume (HV) [73], [76], [140], [200], [201]. Hypervolume, also known as Lebesgue measure [134], *S* metric [227] or 'hyperarea metric' [211] is an indicator of both the convergence and diversity of an approximation set. Thus, given a set *S* containing *m* points in *n* objectives, the hypervolume of *S* is the size of the portion of objective space that is dominated by at least one point in *S*.

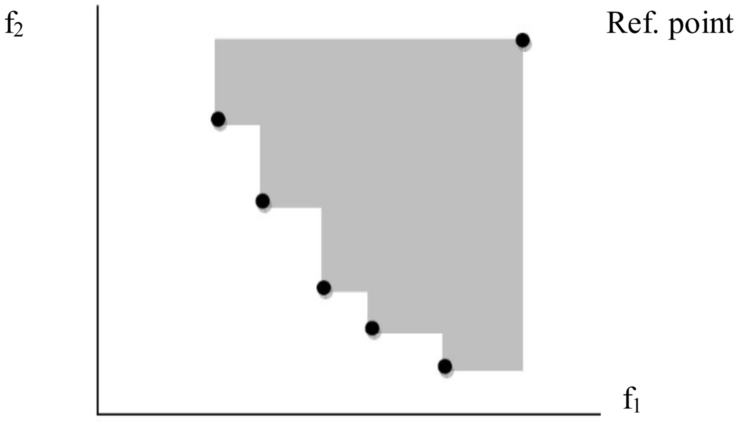


Figure 5.14 Hypervolume of a bi-objective minimization problem

The hypervolume of *S* is calculated relative to a reference point which is worse than (or equal to) every point in *S* in every objective. The greater the hypervolume of a solution the better considered the solution. One of the main advantages of hypervolume [233] is that it is able to capture in a single number both the closeness of the solutions to the optimal set and, to some extent, the spread of the solutions across objective space. According to a number of studies Hypervolume, also has nicer mathematical properties than many other metrics. In particular, Zitzler et al. [234]

state that hypervolume is the only unary metric of which they are aware that is capable of detecting that a set of solutions X is not worse than another set $X^{'}$.

5.5.2 *Epsilon indicator*

Epsilon indicator is the second most popular performance metric as it is indicated by the studies in the field [4], [5], [6], [140]. Zitzler et al. [234] introduced the epsilon indicator. There are two versions of epsilon indicator the multiplicative (I_{ε}) and the additive $(I_{\varepsilon+})$. The basic usefulness of epsilon indicator of an approximation set A is that it provides the minimum factor ε or respectively the minimum term ε by which each point in the real front R can be multiplied or respectively added, such that the resulting transformed approximation set is dominated by A. The epsilon indicator is a good measure of diversity, since it focuses on the worst case distance and reveals whether or not the approximation set has gaps in its trade-off solution set. The smaller the epsilon indicator, the better is the approximation set. The Pareto-optimal solutions set has $I_{\varepsilon} = 1$ and $I_{\varepsilon+} = 0$.

The epsilon indicator I_{ε} is defined for two Pareto sets A and B as $I_{\varepsilon}(A,B) = inf_{\varepsilon \in \mathbb{R}}\{\forall\, z^2 \in B \;\exists\; z^1 \in A\colon z^1 \;\preccurlyeq_{\varepsilon}\; z^2\}$. Where in the case of a minimization problem the multiplicative ε-dominance relation is defined as: $z^1 \preccurlyeq_{\varepsilon} z^2 \Leftrightarrow \;\forall\, i \in 1, \dots, n\colon z_i^1 \;\preccurlyeq\; \varepsilon \cdot z_i^2$. The additive epsilon indicator $I_{\varepsilon+}$ is defined by using the additive ε-dominance: $z^1 \;\preccurlyeq_{\varepsilon}\; z^2 \Leftrightarrow \forall\, i \in 1, \dots, n\colon z_i^1 \;\preccurlyeq\; \varepsilon + z_i^2$.

Many studies in the field use both HV and Epsilon indicator for the evaluation of the relevant results. According to Knowles et al. [126] when two algorithms generate conflicting preferences between these two metrics they are incomparable.

5.5.3 *Generational distance*

Generational distance (*GD*) [212] is the third most popular performance metric as it is indicated by the studies in the field. The *GD* is used to measure the proximity of the approximate efficient frontier (EF_{known}) found from the actual efficient frontier and is given by the following relationship:

$$GD = \sqrt{\left(\frac{1}{m}\sum_{i=1}^{m} d_i^2\right)}$$

Where m is the number of solutions found, d_i is the Euclidean distance in objective space, between the member *i* in EF_{known} and its nearest member of the actual efficient frontier (EF_{true}). A low value of *GD* indicates that the EF_{known} is very close to the true efficient frontier. A value of *GD* = 0 indicates that all the elements generated are in the Pareto optimal set.

5.5.4 *Spread metric*

Spread metric (*Δ*) [57] examines how evenly the solutions are distributed among the approximation sets in objective space. First, it calculates the Euclidean distance between the consecutive solutions in the obtained non-dominated set of solutions. Then it calculates the average of these distances. After that, from the obtained set of non-dominated solutions the extreme solutions are calculated. Finally, using the following metric it calculates the nonuniformity in the distribution.

$$\Delta = \frac{d_f + d_l + \sum_{i=1}^{N-1} |d_i - \bar{d}|}{d_f + d_l + (N-1)\bar{d}}$$

Where d_f and d_l are the Euclidean distances between the extreme solutions and the boundary solutions of the obtained nondominated set. The parameter $\bar{d}$ is the average of all distances d_i, $i = 1, 2, \ldots, (N - 1)$, where *N* is the number of solutions on the best nondominated front. A low value of *Δ* metric indicates wide and uniform spread out of the solutions across the Pareto front. *Δ* takes the value of zero for the optimum spread out of the solutions across the Efficient frontier.

5.5.5 *Maximum spread*

Maximum spread (*MS*) metric [35], [229] measures how well the true efficient frontier (*EF_{true}*) is covered by *EF_{known}* through the hyper-boxes formed by the extreme function values observed in both fronts. Deb [53] suggested that *MS* metric can be misleading if the objective functions have different ranges. For that reason a normalized version of the *MS* metric has been proposed:

$$MS = \sqrt{\frac{1}{L}\sum_{l=1}^{L}\left[\frac{\left(\max_{1\le i\le m} f_l^i - \min_{1\le i\le m} f_l^i\right)}{(F_l^{max} - F_l^{min})}\right]^2}$$

Where m is the size of the obtained approximate efficient frontier (EF_{known}), f_l^i is the function value of the l-th objective of solution i, F_l^{max} and F_l^{min} are the maximum and minimum of the l-th objective in the true efficient frontier (EF_{true}). The greater the value of *MS* metric is, the more the area of the EF_{true} is covered by EF_{known}.

5.5.6 *R2 metric*

R2 metric [107] uses the $C(A,B,u)$ function to decide which of two approximations is better on utility function u, without measuring by how much. Where A and B are two approximation sets, U is a set of utility functions, u: $R^K \to R$ which maps each point in the objective space into a measure of utility. Let u^* be the function defined as: $u^*(A) = max_{Z\in A}\{u(z)\}$ and

$$C(A,B,u) = \begin{cases} 1\ if\ u^*(A) > u^*(B) \\ \frac{1}{2}\ if\ u^*(A) = u^*(B) \\ 0\ if\ u^*(A) < u^*(B) \end{cases}$$

and p is an intensity function expressing the probability density of each utility function $u \in U$. *R2* takes into account the expected values of the utility. In particular, *R2* calculates the expected difference in the utility of an approximation set A with another one B. The *R2* metric is given by the following relationship:

$$R2(A,B,U,p) = E(u^*(A)) - E(u^*(B)) = \int_{u\in U} u^*(A)p(u)du - \int_{u\in U} u^*(B)p(u)du$$

5.5.7 *The spacing metric*

The spacing metric (*S*) [227] measures how evenly the solutions are distributed in the approximate efficient frontier (*EF*$_{known}$) and it is given by the following relationship:

$$S = \frac{\sqrt{\frac{1}{m}\sum_{i=1}^{m}(d_i - \bar{d})^2}}{\bar{d}}$$

Where d_i is the Euclidean distance in objective space, between the solution *i* and its nearest solution in the approximate efficient frontier. A value of zero for this metric indicates that the solutions in the approximate efficient frontier are equidistantly spaced.

5.5.8 *Coverage of two sets metric*

Coverage of two sets (*C*) metric [227] compares the quality of two non-dominated sets. Let *A* and *B* be two different sets of non-dominated solutions, then the *C* metric maps the ordered pair (*A, B*) into the interval [0, 1]:

$$C(A,B) = \frac{|\{b \in B/\ \exists a \in A: a \succcurlyeq b\}|}{|B|}$$

Where *α* and *b* are candidate solutions of sets *A* and *B* respectively. If *C(A, B)* = 1, all the candidate solutions in *B* are dominated by at least one solution in *A*. Likewise, if C(*A, B*) = 0, no candidate solutions in *B* is dominated by any solution in *A*. Finally, C(*A, B*) is not necessarily equal to *1 - C(B, A)*.

5.6 Recommendations for Future Research

The examination of the best practices in the field of the solution of the constrained portfolio optimization problem with the support of MOEAs resulted in some interesting findings.

For instance it appears to exist unanimity among the researchers about the best practices in the field for certain aspects of the problem's configuration. A striking example of the above statement is the adoption of the tournament selection operator by the 95% of the studies in the field. Moreover, 85% of the studies in the field make use of the hybrid representation scheme proposed by Streichert et al. [200].

However, the situation is more perplexed when we examine the best practices in the field with regard to some other important aspects of the problem's configuration. For example, it does not seem to exist a mutation operator that dominates the field.

Even more perplexed is the situation about the most appropriate crossover operator, as 40.91% of the studies in the field make use of the uniform crossover, while the 22.73% of the studies in the field make use of the single point crossover, 13.64% make use of the multi point crossover, also another 13.64% make use of the SBX and 9.09% make use of the two point crossover. We believe that whenever the situation is unclear, like the one we just described above, comparative experimental research can be of great value in the process of identifying the best practices in the field. Moreover we believe the development and adoption of more efficient genetic operators is very important for the performance of the entire algorithm.

More experimental work is also required on the interrelation of the various constraints imposed to the problem as this area of research is in its early stages. In particular, it should be studied how the extracted solutions are affected in the presence of multiple constraints. For serving better that purpose, different combinations of constraints should be examined. The resulting violations should be highlighted and whenever possible the outline of potential repair mechanisms for efficient constraints' handling should be proposed. Moreover the development and adoption of more efficient representation schemes is very important for the performance of the entire algorithm.

Another area of concern is the adoption of commonly acceptable performance metrics for the evaluation of the experimental results. A major issue that is related with the various performance metrics is the different focus of each particular metric. It is not unusual an algorithm to perform well in a particular metric and badly in another. All around Performance metrics such as the HV and Epsilon indicator are favoured by the researchers in the field as it is revealed by the relevant figures (37.50% and 16.67% respectively) as they are able to catch different aspects of the performance of a multiobjective algorithm. Moreover, due to their generic nature, they can facilitate the comparison of the experimental results between the various studies.

We believe that is important to be intensified the ongoing research towards the development of generic performance metrics that are able to capture the performance of an algorithm as a whole instead of focusing on certain aspects of the algorithms' performance.

Finally, we also believe that should be examined the incorporation of preferences in the solution of the constrained portfolio optimization problem with the support of MOEAs. Successful methodologies from the domain of the multiple criteria decision making (MCDM) such as [115], [116], [114], [113] with the appropriate adaptation for the solution of the portfolio optimization problem can be of great value toward the incorporation of preferences.

5.7 Conclusions

This chapter through the systematic study of the state-of-the-art presents the best practices in the field of the constrained portfolio optimization problem with the support of multiobjective evolutionary algorithms (MOEAs) from an algorithmic and technical point of view. In particular, we start with a concise presentation of the constrained portfolio optimization problem (CPOP) with the support of MOEAs. We proceed by identifying the most popular MOEAs for solving the CPOP as they emerge by the relevant studies in the field.

The MOEAs is their standard format cannot be used for solving the constrained portfolio optimization problem. Special algorithmic and technical treatment is required for the efficient incorporation of the constraints into the MOEAs. By examining the state-of-the-art we provide the best practices for efficient constraints handling.

Moreover we examine a number of configuration issues related to the application of MOEAs for solving the constrained portfolio optimization problem. In particular we analyse a number of technical issues like the selection of the most appropriate encoding type, mutation, recombination and selection operator for solving the CPOP with the support of MOEAs.

Then our focus is concentrated on the evaluation of the relevant results from the implementation of the MOEAs to the solution of the constrained portfolio optimization problem. The evaluation of the relevant results is not straightforward due to the multiobjective nature of the portfolio problem. Once again through the systematic study of the state-of-the-art we identify the most popular performance metrics in the field along with their corresponding strengths and weaknesses.

Finally, based on our findings we identify some potential paths for future research. Our aspiration is that the collection and analysis of the best practices in the field of the constrained portfolio optimization problem with the support of MOEAs will contribute to the ongoing research in the field and will encourage the interested researchers to use some of our findings in their corresponding studies.

Chapter 6

A new Probe Guided Mutation Operator and its application for solving the Cardinality Constrained Portfolio Optimization Problem

6.1 Introduction

Evolutionary Algorithms (EAs) have been increasingly applied over the past years for the solution of optimization problems with multiple objectives. The typical Multiobjective Evolutionary Algorithm (MOEA) utilizes three basic operators: selection, crossover and mutation [1]. However, the available literature regarding the variation operators for evolutionary multiobjective optimization remains relatively small. In particular, the mutation operator has received little attention and the majority of MOEAs make use of the Polynomial mutation (PLM) operator proposed by Deb & Goyal [56]. Later, Deb & Tiwari [61] proposed a highly disruptive version of the polynomial mutation that has been utilized in the latest version of NSGA-II [57] and SPEA2 [231]. This chapter proposes a new version of the highly disruptive polynomial mutation (PLM) named Probe Guided Mutation (PGM) operator that produces better results. More recently, Da Ronco & Benini [48] presented a Shrink-Mutation operator for MOEAs that belongs to the Gaussian mutations category. However, as the authors admit the PLM operator when applied to IBEA algorithm gives better results than the Shrink-Mutation operator in terms of convergence towards the True Pareto Front [48]. Das et al. [50] presented a *p*-best mutation strategy for Evolutionary Programming (EP). EP relies mainly on its mutation operator for function optimization. Shortly the *p*-best mutation operator entails that any one of the *p* top-ranked population members according to fitness value is selected randomly for mutation. Tang & Tseng [205] presented a new mutation operator called ADM for real coded Genetic Algorithms (GAs). According to the authors, the ADM mutation operator enhances the abilities of GAs in searching global optima as well as in speeding convergence by integrating a local directional search and adaptive random search strategies.

The majority of the most recent mutation operators have been developed for differential evolutions (DE) algorithms [174]. Thus, Zhou et al. [225] proposed a new mutation operator called intersect mutation differential evolution (IMDE) algorithm. Alguliev et al. [2] proposed a new DE algorithm based on self-adaptive mutation and crossover (DESAMC). The proposed method dynamically adapts scale factor and crossover rate. Gong et al. [103] present a ranking-based mutation operator that makes the DE algorithm to converge faster.

Although the considerable amount of the ongoing work related to mutation operators and their importance for the performance of the entire algorithm, we noticed that the study of mutation operators in the context of MOEAs remains relatively rare. The PLM operator remains undoubtedly the mutation operator of choice when it comes to MOEAs. The motivation of this study is to build on the existing PLM and present a mechanism (the PGM) that allows the better exploration of solution space and is able to generate near optimal solutions that lies very close to the True Efficient Frontier (TEF).

The remainder of the chapter is organized as follows. In section 6.2, a description of the highly disruptive Polynomial Mutation (PLM) is given and in section 6.3 the proposed Probe Guided Mutation (PGM) and the formulation of the cardinality constrained portfolio optimization problem (CCPOP) are presented. In section 6.4 the implementation of the cardinality constraint and lower and upper bound to the MOEA are presented. The parameters setup is presented in section 6.5.1 and in section 6.5.2 we formulate the CCPOP as a Mixed Integer Quadratic Program (MIQP) and we extract the True Efficient Frontier for each one of the examined problems with the assistance of CPLEX 12.5. Section 6.6 presents the performance metrics. In section 6.7 we test the performance of the proposed PGM by using data sets from six different stock markets for the solution of the CCPOP. In section 6.8 the results from the implementation of the cardinality constrained portfolio optimization problem (CCPOP) are analyzed. In section 6.9 we provide an experimental presentation of the PGM operator with the assistance of the ZDT1-4, 6 and DTLZ1-7 families of test functions and the relevant results are analyzed. Finally, section 6.10 concludes the chapter.

6.2 Polynomial Mutation

Mutation operators are being used as variation mechanisms to change the offspring genes. They assist to the better exploration of the search space. For MOEAs solving Multiobjective Problems (MOPs), Deb & Goyal [56] proposed a variation mechanism called polynomial mutation (PLM). This operator was later improved by Deb & Tiwari [61].

In polynomial mutation as introduced by [61] each decision variable x_i, can take values in the interval: $x_i^{(L)} \leq x_i \leq x_i^{(U)}, i = 1, 2, \ldots, n$. Where $x_i^{(L)}$ and $x_i^{(U)}$ stand respectively for the lower and upper bounds for the decision variable i. Moreover, each decision variable has a probability P_m to be perturbed. For each decision variable, a random value *rand* is drawn. If *rand* $\leq P_m$ then using the algorithm described in Fig 6.1, a mutated variable obtains its new value. If random value is $r \leq 0.5$ it samples to the left hand side (region between X_{Low} and X_i), otherwise if $r > 0.5$ it samples to the right hand side (region between X_i and X_{Upper}). The algorithm also calculates the δ_q value to be used in getting the variable its new value. According to Deb & Tiwari [61] one of the main advantages of polynomial mutation is that it allows us to sample the entire search space of the decision variable even though the value to be mutated is close to one of the boundaries (X_{Low} - X_{Upper}).

Moreover, because PLM allows big jumps in the search space of the decision variable, the optimization process has better chances of escaping from local optima and can modify a solution when on the boundary. However, high disruption levels might not be good for achieving smooth approximation of the Pareto front. For instance, if a solution is near an optimal solution then large jumps in the decision space might not be an efficient way of discovering other optimal solutions too. The highly disruptive PLM has been applied in the latest version of NSGAII [57], SPEA2 [231] and a java implementation of PLM algorithm provided by the jMetal framework [77].

```
Begin
mutation_probability = 1/n; (where n is the number of decision variables)
$\eta_m$ = distribution index;

 for i=0 to N; (where N is the population size)
      for z=0 to n;
           $X_p$ = getValue(z);
           $X_l$ = getLowerBound(z);
           $X_u$ = getUpperBound(z);

            rand ⟶ [0, 1];
            if (rand <= mutation_probability) then

            $\delta_1 = \frac{X_p - X_l}{X_u - X_l}$      $\delta_2 = \frac{X_u - X_p}{X_u - X_l}$

             r ⟶ [0, 1];

                 if (r<= 0.5) then
                     $\delta_q = [2r + (1-2r)(1-\delta_1)^{\eta_m+1}]^{\frac{1}{\eta_m+1}}$
                  else
                     $\delta_q = 1 - [2(1-r) + 2(r-0.5)(1-\delta_2)^{\eta_m+1}]^{\frac{1}{\eta_m+1}}$
                 end if

                  $X_c = X_p + \delta_q(X_u - X_l)$
                        if ( $X_c < X_l$ ) then
                             $X_c = X_l$ ;
                          endif
                        if ( $X_c > X_u$ ) then
                             $X_c = X_u$ ;
                         endif

             Child_Solution =  Parent_Solution.setValue(z ,  $X_c$);
            endif
      endfor
endfor
```

Figure 6.1 Polynomial Mutation (PLM) Pseudo code

6.3 Probe Guided Mutation (PGM)

The Probe Guided Mutation (PGM) operator, as its name reveals has been developed in order to facilitate the more efficient exploration of the search space. We will start analyzing PGM mechanism by recalling Fig.6.1, as the first step is common for both methods. In particular, *if rand* $\leq P_m$ then a decision variable is selected to be mutated. Suppose a hypothetical solution vector $\mathbf{x} = (x_1 \, , \, x_2 \, , \dots , x_{10})$ which satisfies the variables bounds $x_i^{(L)} \leq x_i \leq x_i^{(U)}, \; i = 1, 2, \dots , 10$. Also suppose that a random number *rand* $\leq P_m$ occurs for the 6th decision variable. That means that the 6th decision variable of the parent solution should be mutated.

Selected variable for mutation
$\downarrow$

Parent solution	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}

As shown in Fig.6.1 that illustrates the Polynomial Mutation (PLM) operator, if random value is $r \leq 0.5$ it samples to the left hand side (region between X_{Low} and X_i), otherwise if $r > 0.5$ it samples to the right hand side (region between X_i and X_{Upper}). In PGM at this particular point, as shown in Fig.6.2 we follow a different methodology. Specifically, instead of generating a random number $\mathrm{r} \in [0, \; 1]$, we generate two random numbers, $\mathrm{r}_L \in [0, 0.5]$ to sample the left hand side and a random number $\mathrm{r}_R \in (0.5, 1]$ to sample the right hand side. Then, we calculate parameter δ_q^L and δ_q^R as shown in Fig. 2 and their values make possible the calculation of the X_C^L and X_C^R, where X_C^L stands for the child decision variable that samples the region between X_{Low} and X_p. Similarly X_C^R stands for the child decision variable that samples the region between X_p and X_{Upper}. Please notice that X_p is the parent decision variable.

So far by implementing the PGM we obtained two child decision variables, one that samples in the left hand side of the parent variable and one that samples in the right hand side of the parent variable. But, how we decide which is the most prolific way to follow? In order to answer that question we create two new child solution vectors (*Child_Solution*^{L} and *Child_Solution*^{R}) by reinserting into the parent solution vector at the position of the selected variable to be mutated, the X_C^L and X_C^R respectively. Thanks to the generated child solution vectors we are able to evaluate the corresponding fitness for each one of the cases.

Child decision variable X_C^L is reinserted

↓

Child_solution^L	x_1	x_2	x_3	x_4	x_5	x_C^L	x_7	x_8	x_9	x_{10}

Child decision variable X_C^R is reinserted

↓

Child_solution^R	x_1	x_2	x_3	x_4	x_5	x_C^R	x_7	x_8	x_9	x_{10}

The fitness evaluation process is exactly similar to the one implemented by the selection operator so this step does not require further clarification. After completing the fitness evaluation process depending on the examined problem we obtain fitness evaluations for each one of the objective functions. In our case, we decided to implement the proposed methodology to the solution of the cardinality constrained portfolio optimization problem (CCPOP). For that reason we used data sets from the publicly available OR-Library retained by Beasley [28]. In this study we examine the portfolio optimization problem as a bi-objective problem, where the expected return is maximized (*objective 1*) and the risk is minimized (*objective 2*). Below, we provide a formal introduction to our problem. In its bi-objective form, the CCPOP can be formulated as follows. *Let* Ω *be the search space. Consider 2 objective functions* f_1, f_2 *where* $f_i: \Omega \to R^m$. *and* $\Omega \subset R^m$.

Optimize $\quad \mathrm{f(w) = (f_1(w), f_2(w))}$ Eq.1

Maximize portfolio return $\quad \mathrm{f_1(w) = \sum_{i=1}^{m} w_i \overline{r_i}}$ Eq.2

Minimize portfolio risk $\quad \mathrm{f_2(w) = \sum_{i=1}^{m} \sum_{j=1}^{m} w_i w_j \sigma_i \sigma_j \rho_{ij}}$ Eq.3

s.t.

-Budget constraint or summation constraint $\sum_{i=1}^{m} w_i = 1$, *requires all portfolios to have non-negative weights (*$0 \leq w_i \leq 1 \;\; i = 1,2,..,m$*) that sum to 1.*

- *Floor and ceiling constraint* $a_i \leq w_i \leq b_i, \forall = 1,2,..., m$. *Where a_i is the minimum weighting that can be held of asset i (i = 1, ..., m), b_i = the maximum weighting that can be held of asset i (i = 1, ...,m) and* $0 \leq a_i \leq b_i \leq 1 \;\; , \forall = 1,2,..., m$.

- *Cardinality constraint* $C_{\min} \leq \sum_{i=1}^{m} q_i \leq C_{\max}$, *where C_{min} is the minimum number of assets that a portfolio can hold, C_{max} is the maximum number of assets that a portfolio can hold, q_i = 1 , for w_i > 0 and q_i = 0 , for w_i = 0.*

<u>*where:*</u>

- *Decision variables* $w = (w_1,.., w_m)$ subject to $w \subset \Omega$ *and m equal to the number of stocks.*

-*Rate of return of assets:* $r_1, r_2.., r_m$.

- ρ_{ij} *is the correlation between asset i and j and* $-1 \leq \rho_{ij} \leq 1$.

- σ_i, σ_j *represent the standard deviation of stocks returns i and j*

From the formal presentation of the portfolio optimization problem it becomes clear that it is a bi-objective problem where the first objective corresponds to the *Return of assets* and the second objective corresponds to the portfolio risk which is a function of stocks' *Standard deviation* and *Correlation* alike. The higher the portfolio's *return* the better and the lower the portfolio's *risk* the better.

As soon as we obtain, the fitness value evaluations for both child solution vectors, we use the Pareto optimality conditions, as shown in Fig.6.2 in order to choose between the X_C^L and X_C^R, the child decision variable that generates the best results. In particular, we use the following definition.

Definition: Pareto dominance

Consider a maximization problem. Let **x**, **y** be two decision vectors (solutions) from Ω. Solution **x** dominates **y** (also written as $\mathbf{x} \succ \mathbf{y}$) if and only if the following conditions are fulfilled:

(i) The solution **x** is no worse than **y** in all objectives $f_i(\mathbf{x}) \geq f_i(\mathbf{y}), \forall\, i$ = *1, 2,.., n.*

(ii) The solution **x** is strictly better than **y** in at least one objective, $\exists\, j \in \{1,2,\dots,n\}$: $f_j(\mathbf{x}) > f_j(\mathbf{y})$. If any of the above conditions is violated the solution **x** does not dominate the solution **y**.

Referring to Fig. 2, $F_1^L$and F_2^L stand for the fitness evaluations for the *return* and *risk* objective respectively of the *Child_solution^L* vector. Similarly, $F_1^R$and F_2^R stand for the fitness evaluations for the *return* and *risk* objective respectively of the *Child_solution^R* vector. Thus, for example, *if* $F_1^L > F_1^R$ and $F_2^L < F_2^R$, then *Child_solution^L* strongly dominates *Child_solution^R* and the X_C^L is selected. Similarly, *if* $F_1^L > F_1^R$ and $F_2^L = F_2^R$ or, *if* $F_1^L = F_1^R$ and $F_2^L < F_2^R$, then *Child_solution^L* weakly dominates *Child_solution^R* and again the X_C^L is selected. However, *if* $F_1^L > F_1^R$ and $F_2^L > F_2^R$ or, *if* $F_1^L < F_1^R$ and $F_2^L < F_2^R$, then *Child_solution^L* and *Child_solution^R* are non-dominated. In that case a random number r ∈ $[0,\ 1]$ is generated, if *r ≤ 0.5 the* X_C^L is selected, otherwise the X_C^R is selected. The next step is similar for PGM and PLM alike. Specifically, if the resulting X_C is below the lower bound or above the upper bound then the X_C is set equal to X_L or X_U respectively. Where X_L and X_U stand respectively for the lower and upper bounds for the decision variable.

```
Begin
mutation_probability = 1/n;  //where n is the number of decision variables
$\eta_m$ = distribution index;
 for i=0 to N; (where N is the population size)
     for z=0 to n;
         rand → [0, 1];
          if (rand <= mutation_probability) then

             $X_p$ = getValue(z);
             $X_l$  = getLowerBound(z);
             $X_u$  = getUpperBound(z);

             $\delta_1 = \frac{X_p - X_l}{X_u - X_l}$        $\delta_2 = \frac{X_u - X_p}{X_u - X_l}$

             $r_L \rightarrow [0, 0.5];$
             $\delta_q^L = [2r_L + (1 - 2r_L)(1 - \delta_1)^{\eta_m+1}]^{\frac{1}{\eta_m+1}}$
             $X_C^L = X_p + \delta_q^L(X_u - X_l)$
             Child_Solution$^L$ = Parent_Solution.setValue(z , $X_C^L$);
             $F_i^L$= Evaluate Fitness (Child_Solution$^L$); //where i=1, 2 for bi-objective problem

             $r_R \rightarrow (0.5, 1];$
             $\delta_q^R = 1 - [2(1 - r_R) + 2(r_R - 0.5)(1 - \delta_2)^{\eta_m+1}]^{\frac{1}{\eta_m+1}}$
             $X_C^R = X_p + \delta_q^R(X_u - X_l)$
             Child_Solution$^R$ = Parent_Solution.setValue(z , $X_C^R$);
             $F_i^R$ = Evaluate Fitness (Child_Solution$^R$);

             //We examine the portfolio optimization problem as a bi-objective problem,
             //where the return is maximized (objective 1) and the risk minimized (objective 2)

             if ($F_1^L > F_1^R$ && $F_2^L < F_2^R$|| $F_1^L > F_1^R$ && $F_2^L = F_2^R$|| $F_1^L = F_1^R$ && $F_2^L < F_2^R$) then
                     $X_C = X_C^L$
             else if ($F_1^L < F_1^R$ && $F_2^L > F_2^R$|| $F_1^L < F_1^R$ && $F_2^L = F_2^R$|| $F_1^L = F_1^R$ && $F_2^L > F_2^R$) then
                     $X_C = X_C^R$
             else
                     $r \rightarrow [0, 1];$
                     if (r<= 0.5) then
                         $X_C = X_C^L$
                     else if (r > 0.5) then
                         $X_C = X_C^R$
                     end if
              endif


                 if ( $X_c < X_l$ ) then
                         $X_c = X_l$ ;
                 endif
                 if ( $X_c > X_u$ ) then
                         $X_c = X_u$ ;
                 endif
                Child_Solution = Parent_Solution.setValue(z , $X_c$);
          endif
     endfor
endfor
```

Figure 6.2 Probe Guided Mutation (PGM) operator Pseudo code

The motivation behind this study is to present a mechanism (*the PGM*) that allows the better exploration of solution space. The logic is simple, as soon as a parent decision variable (X_p) is selected to be mutated, two random numbers are generated, one in the range $\mathrm{r}_L \in [0, 0.5]$ and the other in the range $\mathrm{r}_R \in (0.5, 1]$. These two random numbers are being used in order to generate two different child decision variables, the X_C^L that samples the region between X_{Low} and X_p and X_C^R that samples the region between X_p and X_{Upper}. These two child decision variables are reinserted into the parent solution vector, at the position of the selected variable to be mutated and subsequently we perform fitness evaluation for both alternative child solutions vectors. Next, based on, the Pareto optimality framework we decide which child decision variable between the X_C^L and X_C^R will be selected. The PGM allows us to probe the search space and move progressively towards higher fitness solutions. Whenever, there is not a clear winner i.e. strong or weak dominance, between the X_C^L and X_C^R, the generation of a random number allows the random choice of one of the two alternative solutions.

Naturally, there is a computational cost associated with the fitness evaluation process of PGM. Table 6.1 provides the total CPU time, in seconds, needed for each one of the presented tests problems (port1-5) under the classical configuration with the PLM operator and under the proposed PGM. All algorithms have been implemented in Java and run on a 2.1GHz Windows Server 2012 machine with 6GB RAM. On average, the proposed PGM has an overhead of 32% in CPU time when applied to the NSGAII and 35% when applied to SPEA2.

Table 6.1 Average runtime (in seconds) required by each algorithm

Constrained	NSGAII			SPEA2		
Problem*	PGM	PLM	Overhead	PGM	PLM	Overhead
port1	296	247	20%	314	256	23%
port2	4556	3478	31%	4923	3674	34%
port3	5118	3878	32%	5610	4095	37%
port4	6450	4778	35%	6931	5023	38%
port5	20997	14892	41%	23268	16047	45%
		Average:	32%		*Average:*	35%

**Minimum cardinality of the portfolio K_{min} = 2 and maximum cardinality K_{max} = 10 for all test problems. Lower bound l_i = 0.01 and the upper bound u_i = 0.99, for each asset i, where i = 1, . . . , n.*

Finally, we investigate how efficient is to solve the CCPOP by using exact methods. For that purpose, we formulated the CCPOP as a Mixed Integer Quadratic Program (MIQP) (see section 6.2) and we solved the problem by using CPLEX version 12.5. In case that the CPLEX were able to solve the CCPOP MIQP in reasonable time, there may be no need for using evolutionary algorithms. For reasons of comparison we tested CPLEX version 12.5 for the same instances (port1-5). We found out that for all test instances the CPLEX required considerably more time than the MOEAs. Table 6.2 indicates the average running time per *approximate efficient point* in the case of MOEAs and *exact efficient point* in the case of CCPOP MIQP for all test instances (port1-5).

Table 6.2 Average runtime (in seconds) required per efficient point

	Running Time (in seconds) per				
	Approximate Efficient Point				Exact Efficient Point
Constrained Problem*	NSGAII		SPEA2		MIQP
	PGM	PLM	PGM	PLM	CPLEX
port1	1.48	1.24	1.57	1.28	15
port2	23	17	25	18	125
port3	26	19	28	20	148
port4	32	24	35	25	213
port5	105	74	116	80	521

**Minimum cardinality of the portfolio K_{min} = 2 and maximum cardinality K_{max} = 10 for all test problems. Lower bound l_i = 0.01 and the upper bound u_i = 0.99, for each asset i, where i = 1, . . . , n.*

According to the results of Table 6.2 we conclude that solving the CCPOP MIQP using CPLEX is not a computationally effective approach. As such we are justified in adopting multiobjective evolutionary approaches to the solution of the CCPOP. Regarding the quality of the approximate efficient frontier, this issue is examined in section 6.7.

6.4 MOEAs implementation

The implementation of evolutionary algorithms for the solution of the cardinality constrained portfolio optimization problem (CCPOP) involves a number of potential challenges like the selection of the most appropriate solution representation type or the correct implementation of the problem's constraints.

For the solution representation we chose the chromosomal data structure, as implemented by the Streichert et al. [c6.a.17]. In particular Streichert et al. proposed a hubrid representation, where an additional binary string is included to reflect the existence of the assets in the portfolio. Thus, two vectors are used for defining a portfolio, a real-valued vector used to compute the proportions of the budget invested in the various securities and a binary vector specifies whether a particular security participates in the portfolio.

$$W = \{w_1, \ldots, w_n\},\ \ 0.01 \leq\ w_i \leq 0.99, i = 1, \ldots, n$$

$$\Delta = \{\delta_1, \ldots, \delta_n\},\ \ \delta_i \in \{0, 1\}, i = 1, \ldots, n$$

In order to generate portfolios that satisfy the imposed minimum and maximum cardinality and lower and upper bound constraints and the budget constraint, we proceed as follows:

1. The real-valued vector is populated with randomly generated weights within the specified lower and upper bounds [0.01, 0.99].
2. A random number r^{card} that satisfy the minimum and maximum cardinality constraint is generated. For instance, if we set the minimum cardinality of the portfolio to two (K_{min} = 2) and the maximum cardinality of the portfolio to ten (K_{max} = 10) then the r^{card} can be any integer number within the r^{card} ∈ [2, 10] range.
3. The binary vector by default is populated with zeroes and only to r^{card} ∈ [2, 10] randomly selected securities, is assigned value of 1.
4. Next, we disregard the securities that are not elements of the portfolio (i.e. $if\ w_i \neq 0\ and\ \delta_i = 0$).

5. The weights of the remaining (i.e. $if\ w_i \neq 0\ and\ \delta_i = 1$) securities are normalized in order to satisfy the budget constraint. Thus, the actual weight w_i^{Actual} is calculated as follows.

$$w_i^{Actual} = \frac{w_i}{\sum_{i=1}^{n} w}$$

So far the proposed encoding mechanism accounts only for budget, non-negativity constraint and cardinality constraint. The normalization process is possible to lead to violations of the lower and upper bounds constraints.

For the requirements of this study we have set the lower bound $l_i = 0.01$ and the upper bound $u_i = 0.99$, for each asset i, where $i = 1, \ldots, n$. That implicitly suggests a minimum cardinality of two (2) stocks per portfolio. We also explicitly have set the minimum cardinality of the portfolio to two ($K_{min} = 2$) and the maximum cardinality of the portfolio to ten ($K_{max} = 10$) for all test problems. Obviously, the upper bound constraint will always be satisfied, as the worst case scenario for the upper bound constraint is K = 2, with $w_1 = 0.99\ and\ w_2 = 0.01$. Thus, there is no need to introduce a repair mechanism regarding the upper bound constraint.

Regarding the lower bound constraint, a repair mechanism is necessary as during the normalization process can be generated weights that violate the lower bound constraint. In order to deal with this issue we introduce the following repair mechanism:

1. After the completion of the normalization process we check if there is any violation of the lower bound constraint, for the portfolio's securities.
2. We determine which particular securities violate the lower bound constraint ($l_i = 0.01$) and the magnitude of the violation for each one of them, as follows $m_i^{viol} = l_i - w_i^{Actual}$, $i = 1,..,q$.
3. We sum up all violations in the portfolio: $v^{total} = \sum_{i=1}^{q} m_i^{viol}$
4. Finally, we subtract the v^{total} from the highest weight security ($w_{highest}^{Actual}$) in the portfolio (i.e. $w_{new}^{Actual} = w_{highest}^{Actual}$ - v^{total}). Thus, the highest weight security in the portfolio obtains a new weight (w_{new}^{Actual}) and we set the weights of the securities that violated the lower bound constraint equal to the l_i .

6.5 Experimental Environment

6.5.1 Parameter Setup

All algorithms have been implemented in Java and run on a 2.1GHz Windows Server 2012 machine with 6GB RAM. The jMetal [77] framework has been used to compare the performance of the proposed, Probe Guided mutation operator against the classical polynomial mutation operator with the assistance of two state-of-the-art MOEAs, namely NSGAII and SPEA2. We set the minimum cardinality of the portfolio to two ($K_{min} = 2$) and the maximum cardinality of the portfolio to ten ($K_{max} = 10$) for all test problems. The participation of each stock in the portfolio is determined by the lower and upper bounds. We set the lower bound $l_i = 0.01$ and the upper bound $u_i = 0.99$, for each asset i, where $i = 1, . . . , n$.

In all tests we use binary tournament and simulated binary crossover (SBX) [54] as selection and crossover operator, respectively. The crossover probability is $P_c = 0.9$ and mutation probability is $P_m = 1/n$, where n is the number of decision variables. The distribution indices for the crossover and mutation operators are $\eta_c = 20$ and $\eta_m = 20$, respectively. We decided to use the aforementioned values for the crossover and mutation probability as they are the most commonly used with MOEAs in the relevant literature. For identifying the best population size and function evaluations, the following configurations were tested for each algorithm: $N^{pop} = N^{arc}$ = 100, 200, 300, 400, for 100, 250, 500 and 1000 generations respectively. After testing the aforementioned configurations we reach the conclusion that the algorithms approach their top performance for $N^{pop} = N^{arc}$ = 200 and 500 generations or 100,000 function evaluations. Beyond, these values the gains in performance are marginal if any. Thus, the Population size is set to *200*, using *100,000* function evaluations with *20* independent runs.

6.5.2 Extracting the True Pareto Front of the CCPOP

The purpose of this study is to highlight the benefits from the implementation of the proposed Probe Guided Mutation (PGM) operator to the solution of the cardinality constrained portfolio optimization problem (CCPOP) when compared with the results derived by classical Polynomial Mutation (PLM) for the same test instances. However, it would be interesting to find out how well the PGM operator performs when compared to the True Pareto front of the CCPOP. For that purpose, we formulated the CCPOP as a Mixed Integer Quadratic Program (MIQP) as follows:

Minimize portfolio risk $$\sum_{i=1}^{N}\sum_{j=1}^{N} w_i w_j \sigma_{ij} \qquad \text{Eq.4}$$

Subject to: $$\sum_{i=1}^{N} w_i \bar{r}_i = R^* \qquad \text{Eq.5}$$

$$\sum_{i=1}^{N} w_i = 1 \qquad \text{Eq.6}$$

$$K_{\min} \le \sum_{i=1}^{N} z_i \le K_{\max} \qquad \text{Eq.7}$$

$$a_i\, z_i \le w_i \le b_i\, z_i, \forall = 1,2,\ldots,N \qquad \text{Eq.8}$$

$$z_i \in [0,1], i = 1,\ldots,N \qquad \text{Eq.9}$$

Eq. (4) minimizes the total variance that represents the portfolio's risk. The return constraint that is represented by the Eq. (5) ensures that the portfolio has an expected return of R^*. Eq. (6) represents the budget constraint. Eq. (7) ensures that the portfolio lies between the K_{min} and K_{max}. In this particular case we set $K_{min} = 2$ and $K_{max} = 10$ for all test instances. Eq. (8) ensures that if any of security *i* is held ($z_i = 1$) its weight must lie between the lower bound a_i and the upper bound b_i. In this particular case we set $a_i = 0.01$ and $b_i = 0.99$ for all test instances. Please, notice that the objective function Eq. (4), involves the covariance matrix (σ_{ij}), which is positive semidefinite and thus we are minimizing a convex function.

The calculation of the true constrained efficient frontier for port1-5 [28] problems is straightforward. We minimize portfolio risk as it is expressed by Eq. (4), subject to Eq. (5)-(9), for different levels of the R^* of Eq. (5). By varying R^* and solving the Mixed Integer Quadratic Program (MIQP) we obtain different levels of minimum portfolio risk. For solving the aforementioned MIQP we used the CPLEX version 12.5.

Analytically, we utilized the following strategy [216], [217] in order to plot the True Efficient Frontiers for port1-5 problems:

1. For each one of the port1-5 problems we know the range of return values that should be examined, as the securities' returns are known.
2. Let's say that we examine the port1 problem and R_L^{port1} represents the security with the lowest return among all securities of port1 problem. Respectively R_U^{port1} represents the security with the maximum return among all securities of port1 problem.
3. Given the return range $[R_L^{port1}, R_U^{port1}]$ let *M* be the number of points that we wish to plot on the frontier. With the assistance of the r^{step} equation we divide the R_U^{port1} - R_L^{port1} interval into M equally distant points:

$$r^{step} = \frac{R_U^{port1} - R_L^{port1}}{M}$$

4. Then, we calculate the minimum portfolio risk for each of the following returns levels: $[R_L^{port1} + r^{step}, R_L^{port1} + 2r^{step}, ..., R_L^{port1} + Mr^{step}]$. Please notice that we start from $R_L^{port1} + r^{step}$ return level and not from R_L^{port1} because we have set $K_{min} = 2$. Solving the MIQP for R_L^{port1} return level and $K_{min} = 2$ would lead to an infeasible solution.
5. Finally, for port1-3 problems we used M = 100, and for port4-5 problems we used M = 50.

6.6 Performance Metrics

Hypervolume

Hypervolume [117], [176], is an indicator of both the convergence and diversity of an approximation set. Thus, given a set S containing m points in n objectives, the hypervolume of S is the size of the portion of objective space that is dominated by at least one point in S. The hypervolume of S is calculated relative to a reference point which is worse than (or equal to) every point in S in every objective. The greater the hypervolume of a solution the better considered the solution. One of the main advantages of hypervolume [233] is that it is able to capture in a single number both the closeness of the solutions to the optimal set and, to some extent, the spread of the solutions across objective space.

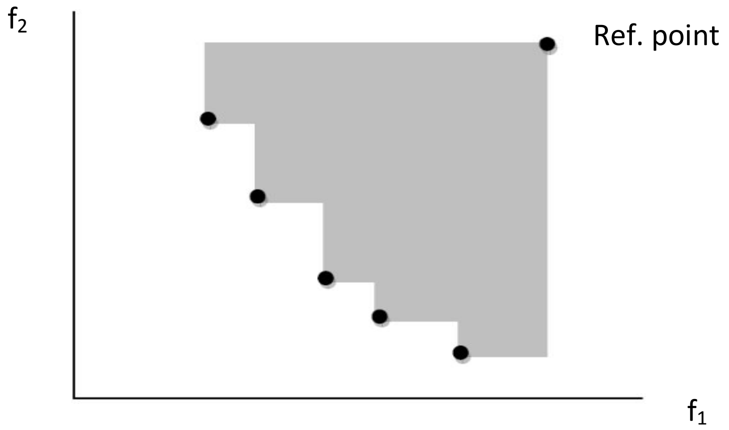


Figure 6.3 Hypervolume of a bi-objective minimization problem

According to a number of studies Hypervolume, also has nicer mathematical properties than many other metrics. In particular, Zitzler et al. [234] state that hypervolume is the only unary metric of which they are aware that is capable of detecting that a set of solutions X is not worse than another set $X^{'}$. Also Fleischer [90] has proved that hypervolume is maximized if and only if the set of solutions contains only Pareto optima.

Spread

Deb et al. [57] introduced the spread of solutions (Δ) as another indicator of the quality of the derived set of solutions. Spread indicator examines whether or not the solutions span the entire Pareto optimal region. First, it calculates the Euclidean distance between the consecutive solutions in the obtained non-dominated set of solutions. Then it calculates the average of these distances. After that, from the obtained set of non-dominated solutions the extreme solutions are calculated. Finally, using the following metric it calculates the nonuniformity in the distribution.

$$\Delta = \frac{d_f + d_l + \sum_{i=1}^{N-1} |d_i - \bar{d}|}{d_f + d_l + (N-1)\bar{d}}$$

Where d_f and d_l are the Euclidean distances between the extreme solutions and the boundary solutions of the obtained nondominated set. The parameter $\bar{d}$ is the average of all distances d_i, $i = 1, 2, \ldots, (N - 1)$, where N is the number of solutions on the best nondominated front. The smaller the value of Spread indicator, the better the distribution of the solutions. Spread indicator takes a zero value for an ideal distribution of the solutions in the Pareto front.

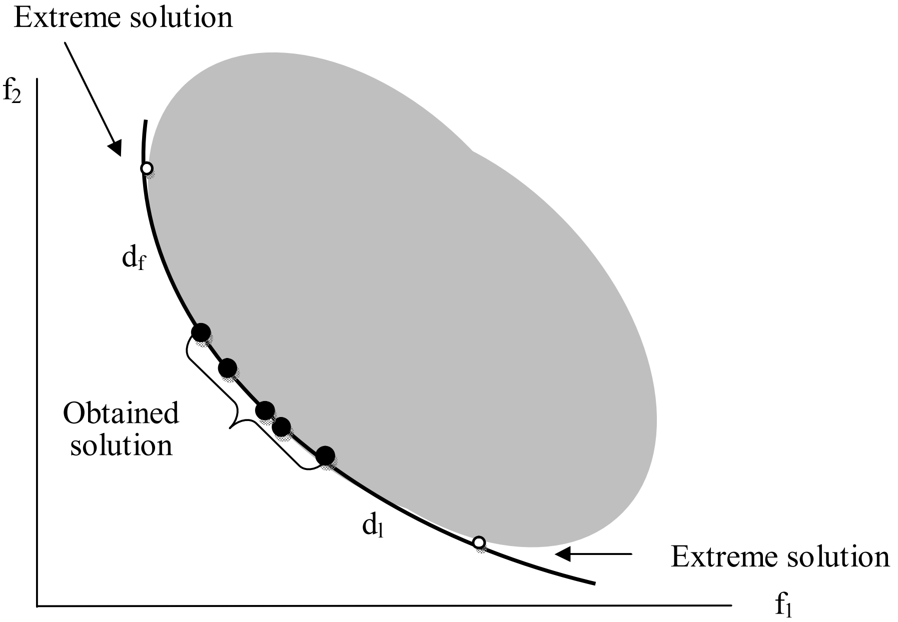


Figure 6.4 Spread (Δ) indicator of a bi-objective minimization problem

Epsilon Indicator I_ε

Zitzler et al. [234] introduced the epsilon indicator (I_ε). There are two versions of epsilon indicator the multiplicative and the additive. In this study we use the unary additive epsilon indicator as it has been implemented in the jMetal framework. The basic usefulness of epsilon indicator of an approximation set A ($I_{\varepsilon+}$) is that it provides the minimum factor ε by which each point in the real front R can be added such that the resulting transformed approximation set is dominated by A. The additive epsilon indicator is a good measure of diversity, since it focuses on the worst case distance and reveals whether or not the approximation set has gaps in its trade-off solution set.

Suppose a minimization problem with *n* positive objectives i.e. $Z \subseteq \mathbb{R}^{+^n}$. An objective vector $\boldsymbol{z^1} = (z_1^1, z_2^1, \dots, z_n^1) \in Z$ is said to *ε-dominate* another objective vector $\boldsymbol{z^2} = (z_1^2, z_2^2, \dots, z_n^2) \in Z$ written as $\boldsymbol{z^1} \succcurlyeq \varepsilon + \boldsymbol{z^2}$ if and only if $\forall\, 1 \leq i \leq n$: $z_i^1 \leq \varepsilon + z_i^2$, where $\varepsilon > 0$. The binary additive epsilon indicator $I_{\varepsilon+}$ which is defined for two Pareto sets A and B as $I_{\varepsilon+}(A, B) = inf_{\varepsilon\in\mathbb{R}}\{\forall\, \boldsymbol{z^2} \in B\ \exists\ \boldsymbol{z^1} \in A\colon \boldsymbol{z^1} \succcurlyeq \varepsilon + \boldsymbol{z^2}\}$. For a reference Pareto set *R*, the unary additive epsilon indicator $I_{\varepsilon+}^1$ can be now defined as $I_{\varepsilon+}^1(A) = I_{\varepsilon+}(A, R)$. The smaller an epsilon indicator, the better is the approximation set. The Pareto-optimal solutions set has $I_{\varepsilon+} = 0$.

6.7 Experimental Results

A number of computational experiments were performed to test the performance of the proposed Probe Guided mutation operator for the solution of the Cardinality constrained Portfolio Optimization Problem (CCPOP). The performance of the proposed PGM operator is assessed in comparison with the classical Polynomial Mutation operator with the assistance of two well-known MOEAs, namely the Non-dominated Sorting Genetic Algorithm II (NSGAII) and Strength Pareto Evolutionary Algorithm 2 (SPEA2). The evaluation of the performance is based on a variety of metrics that assess both the proximity of the solutions to the Pareto front and their dispersion on it. For carrying out the experiments, we used data sets from the publicly available OR-Library retained by Beasley. These data sets correspond to six portfolio optimization problems from six different capital markets as shown in Table 6.3. More specifically, we used weekly price data for the period March 1992 to September 1997, from Hang Seng 31 in Hong Kong, DAX 100 in Germany, FTSE 100 in UK, S&P 100 in USA, Nikkei 225 in Japan and S&P 500 in USA.

Table 6.3 The OR-library portfolio optimization problems

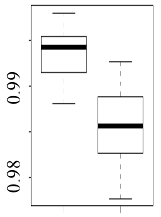
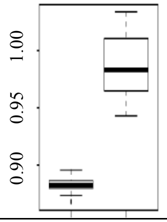
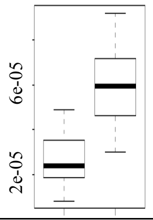
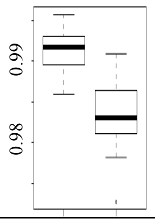
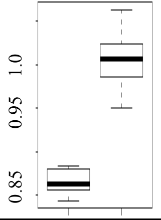
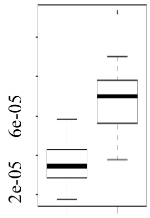
Problem Name	Stock Market Index	Assets
port1	Hang Seng	31
port2	DAX100	85
port3	FTSE100	89
port4	S&P100	98
port5	Nikkei225	225
port6	S&P500	457

6.7.1 Hang Seng (31 stocks) port1 in OR-Library

Table 6.4 Port1 – Mean, std, median and iqr for HV, Spread and Epsilon

Problem		NSGAII		SPEA2	
		PGM	PLM	PGM	PLM
PORT1	HV. Mean and Std	9.94e-01$_{2.6e-03}$	9.85e-01$_{4.3e-03}$	9.91e-01$_{2.6e-03}$	9.84e-01$_{4.3e-03}$
	HV. Median and IQR	9.94e-01$_{4.0e-03}$	9.86e-01$_{6.2e-03}$	9.92e-01$_{3.5e-03}$	9.83e-01$_{5.3e-03}$
	SPREAD. Mean and Std	8.82e-01$_{6.7e-03}$	9.87e-01$_{2.6e-02}$	8.67e-01$_{1.3e-02}$	1.01e+00$_{2.5e-02}$
	SPREAD. Median and IQR	8.83e-01$_{6.9e-03}$	9.83e-01$_{4.5e-02}$	8.63e-01$_{2.5e-02}$	1.01e+00$_{3.7e-02}$
	EPSILON. Mean and Std	2.61e-05$_{1.1e-05}$	6.00e-05$_{1.8e-05}$	3.61e-05$_{1.1e-05}$	6.79e-05$_{1.8e-05}$
	EPSILON. Median and IQR	2.38e-05$_{1.6e-05}$	5.94e-05$_{2.6e-05}$	3.44e-05$_{1.5e-05}$	7.00e-05$_{2.2e-05}$

Table 6.5 Port1 – Boxplots for HV, Spread and Epsilon

Problem	NSGAII						SPEA2					
	HV		SPREAD		EPSILON		HV		SPREAD		EPSILON	
	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM
PORT1												

The results in the Tables 6.4 – 6.21 have been produced by using JMetal [77] framework, a java implementation of many popular MOEAs that has been used in a considerable number of studies in the field. Tables 6.4-6.6, present the results regarding the port1 problem (*Hang Seng index*) in OR-Library. In particular, Table 6.4 presents the results of PGM and PLM alike for a number of performance metrics. Specifically, it presents the mean, standard deviation (STD), median and interquartile range (IQR) of all the independent runs carried out for Hypervolume (HV), Spread (Δ) and Epsilon indicator respectively.

Clearly, regarding the HV [233] indicator the higher the value (i.e. the greater the hypervolume) the better the computed front. HV is considered by many researchers in the field [82], [117], [176], [234], [228] the most important indicator as it captures in a single number both the closeness of the solutions to the optimal set and to a certain degree, the spread of the solutions across objective space. The second indicator the Spread (Δ) [57] examines the spread of solutions across the pareto front. The smaller the value of this indicator, the better the distribution of the solutions. This indicator takes a zero value for an ideal distribution of the solutions in the Pareto front. The third indicator, the Epsilon [126] is a measure of the smaller distance that a solution set A, needs to be changed in such a way that it dominates the optimal Pareto front of this problem. Obviously the smaller the value of this indicator, the better the derived solution set.

Table 6.5 use boxplots to present graphically the performance of NSGAII and SPEA2 under two different configurations, PGM and PLM respectively, for the three performance indicators, namely: HV, Spread and Epsilon. Boxplot is a convenient way of graphically depicting groups of numerical data through their quartiles. Boxplots provide a 5-number summary of the data (min, max, Q1, Q3, median) and information about outliers.

Table 6.6, presents if the results of NSGAII and SPEA2 derived under the two different configurations (PGM and PLM) are statistically significant or not. For that reason, we use the Wilcoxon rank-sum test as it is implemented by the jMetal framework [77]. In Table 6.6, three different symbols are used. In particular “▬” indicates that there is not statistical significance between the algorithms. “▲” means that the algorithm in the row has yielded better results than the algorithm in the column with confidence and “ ▽“ is used when the algorithm in the column is statistically better than the algorithm in the row. The same interpretation holds for the rest of the Tables of section 6.7. Analysis of the relevant results provided in section 6.8.

Table 6.6 Wilcoxon test in Port1 for HV, Spread and Epsilon

Problem			NSGAII with PLM		SPEA2 with PLM
PORT1	HV. Mean and Std	NSGAII with PGM	▲	SPEA2 with PGM	▲
	HV. Median and IQR		▲		▲
	SPREAD. Mean and Std		▲		▲
	SPREAD. Median and IQR		▲		▲
	EPSILON. Mean and Std		▲		▲
	EPSILON. Median and IQR		▲		▲

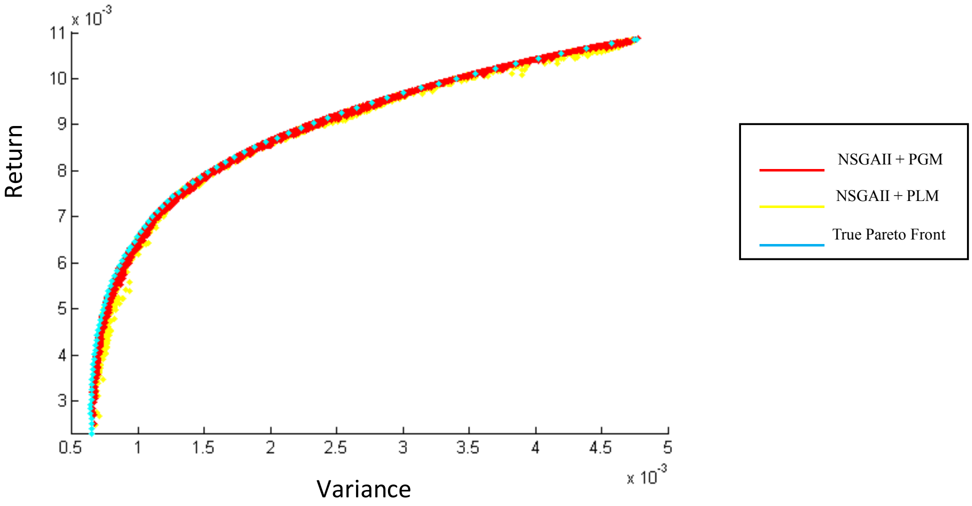


Figure 6.5 NSGAII: Mean-Variance Efficient Frontier for the port1 problem, with n = 31 securities under different configurations.

Fig. 6.5 illustrates the approximate efficient frontier under two different configurations: the NSGAII + PGM and the NSGAII + PLM for the port1 test instance together with the exact efficient points generated by using CPLEX version 12.5 software. The approximate frontiers that are illustrated in Fig.5, have been formulated by merging the approximate sets for the 20 independent runs of the algorithm without removing the dominated solutions. We selected to illustrate the approximate efficient frontiers for the two different configurations of the algorithm (NSGAII + PGM and the NSGAII + PLM) without removing the dominated solutions, because that way we are able to see not only which method generates the best results in term of the Pareto domination conditions, but also to extract useful information about the concentration of the search effort toward the higher fitness regions of the search space for the one and the other configuration of the algorithm. From Fig. 5 it becomes clear that the proposed configuration of the algorithm with the PGM clearly outperforms the typical configuration of the NSGAII with the PLM, as the approximate frontier that is generated by the PLM is dominated by the corresponding approximate frontier of the proposed methodology (PGM). We also notice that the classical PLM operator generates a considerable number of points far away from the efficient frontier. On the other hand the PGM operator demonstrates a remarkable concentration of the search effort very close to the true efficient frontier.

To conclude, the port1 problem results indicate that the proposed PGM operator outperforms the classical PLM in HV, Spread and Epsilon performance metrics with confidence, according to the relevant results as revealed by the Tables 6.4-6.6.

6.7.2 DAX100 (85 stocks) port2 in OR-Library

Table 6.7 Port2 – Mean, std, median and iqr for HV, Spread and Epsilon

Problem		NSGAII		SPEA2	
		PGM	PLM	PGM	PLM
PORT2	HV. Mean and Std	9.84e-01$_{4.1e-03}$	9.68e-01$_{6.9e-03}$	9.79e-01$_{4.4e-03}$	9.67e-01$_{4.6e-03}$
	HV. Median and IQR	9.84e-01$_{5.3e-03}$	9.67e-01$_{1.0e-02}$	9.78e-01$_{7.0e-03}$	9.69e-01$_{7.0e-03}$
	SPREAD. Mean and Std	9.78e-01$_{1.9e-02}$	1.11e+00$_{3.2e-02}$	1.01e+00$_{1.9e-02}$	1.14e+00$_{3.7e-02}$
	SPREAD. Median and IQR	9.78e-01$_{2.1e-02}$	1.11e+00$_{4.1e-02}$	1.01e+00$_{2.8e-02}$	1.14e+00$_{4.3e-02}$
	EPSILON. Mean and Std	4.41e-05$_{1.1e-05}$	8.66e-05$_{1.8e-05}$	5.65e-05$_{1.2e-05}$	8.85e-05$_{1.2e-05}$
	EPSILON. Median and IQR	4.20e-05$_{1.4e-05}$	8.83e-05$_{2.7e-05}$	5.93e-05$_{1.9e-05}$	8.38e-05$_{1.9e-05}$

Table 6.8 Port2 – Boxplots for HV, Spread and Epsilon

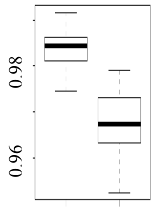
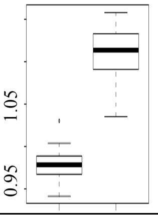
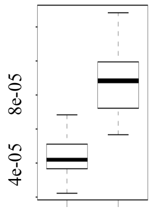
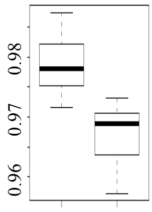
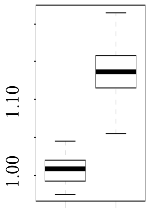
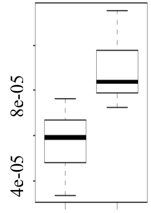
Problem	NSGAII						SPEA2					
	HV		SPREAD		EPSILON		HV		SPREAD		EPSILON	
	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM
PORT2												

Table 6.9 Wilcoxon test in Port2 for HV, Spread and Epsilon

Problem			NSGAII with PLM		SPEA2 with PLM
PORT2	HV. Mean and Std	NSGAII with PGM	▲	SPEA2 with PGM	▲
	HV. Median and IQR		▲		▲
	SPREAD. Mean and Std		▲		▲
	SPREAD. Median and IQR		▲		▲
	EPSILON. Mean and Std		▲		▲
	EPSILON. Median and IQR		▲		▲

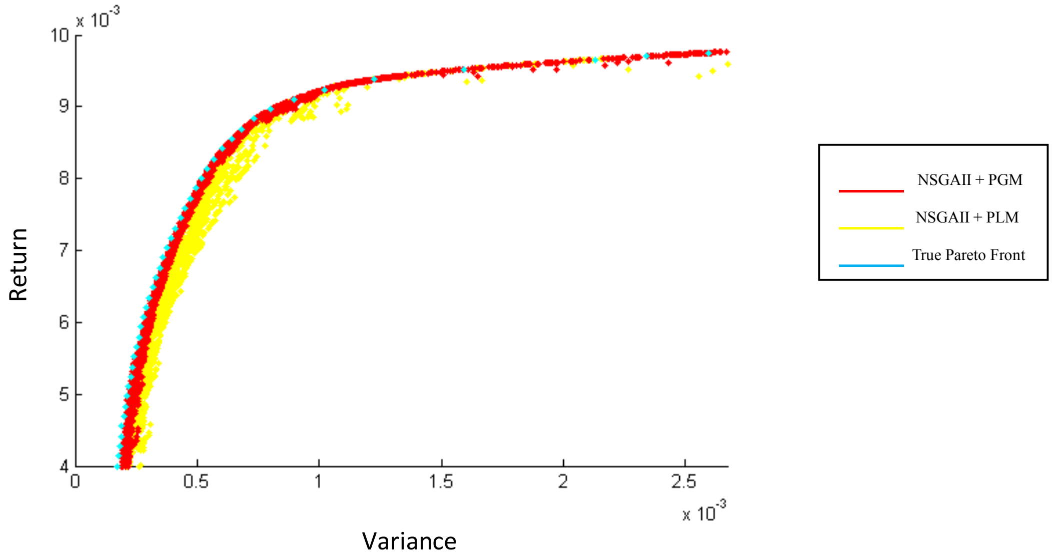


Figure 6.6 NSGAII: Mean-Variance Efficient Frontier for the port2 problem, with n = 85 securities under different configurations

The port2 problem uses data from DAX100 index in Germany. Fig. 6.6 illustrates the approximate efficient frontier under two different configurations: the NSGAII + PGM and the NSGAII + PLM for the port2 test problem together with the exact efficient points generated by using CPLEX 12.5. The approximate frontiers that are illustrated in Fig.6.6, 6.7, 6.8 and 6.9, have been emerged by joining the approximate sets for the 20 independent runs of the algorithm without removing the dominated solutions. From Fig. 6 we can see that the proposed configuration of the NSGAII with the PGM clearly outperforms the typical configuration of the algorithm with the PLM, as the approximate frontier that is generated by the PLM is dominated by the corresponding approximate frontier that is produced by using the PGM. Fig.6.7 and 6.8 illustrate separately the approximate frontiers for the two different configurations of the NSGAII. Fig. 6.6 and 6.7 show us that the NSGAII coupled with the PGM achieves a remarkable concentration of the search effort very close to the true efficient frontier.

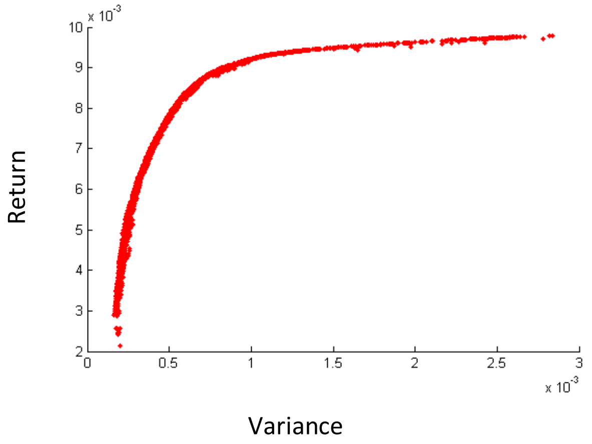


Figure 6.7 NSGAII + PGM: Mean-Variance Efficient Frontier for the port2 problem, with n = 85 securities

On the other hand, as shown in Fig. 6.8, the NSGAII combined with the classical PLM operator does not demonstrates the exploratory strength of PGM. A considerable part of the computational effort has been wasted in discovering suboptimal solutions and on top of that the approximate frontier presents substantial gaps, especially for higher levels of return and risk combinations.

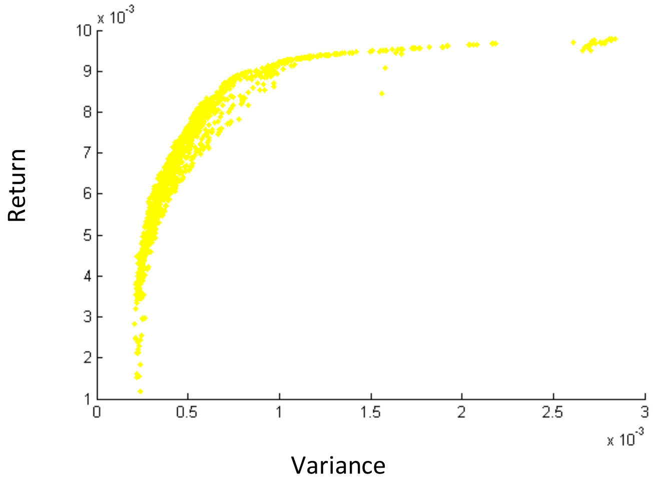


Figure 6.8 NSGAII + PLM: Mean-Variance Efficient Frontier for the port2 problem, with n = 85 securities

Finally, Fig. 6.9 illustrates the approximate efficient frontier under the SPEA2 + PGM and the SPEA2 + PLM configuration for the port2 test problem together with the exact efficient points generated by using CPLEX 12.5. Fig. 6.9 confirms all previous findings.

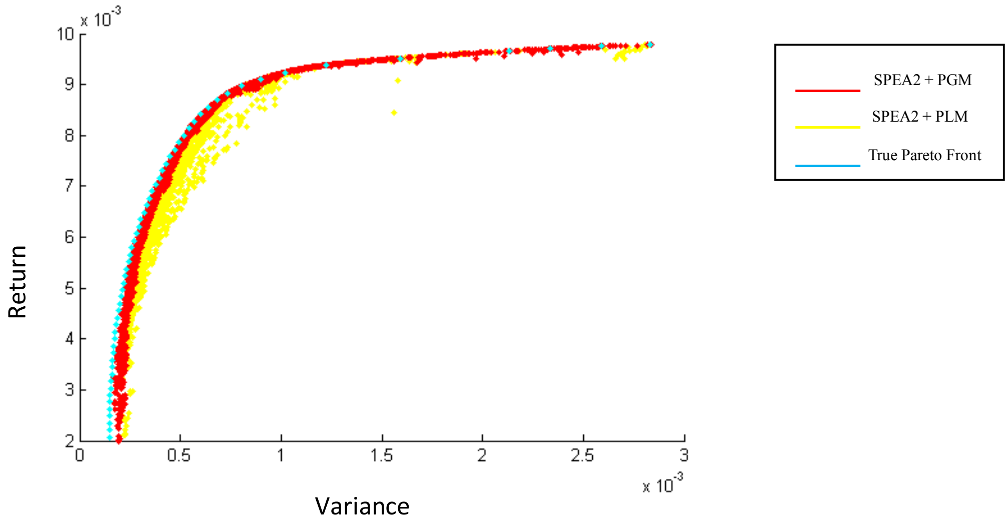


Figure 6.9 SPEA2: Mean-Variance Efficient Frontier for the port2 problem, with n = 85 securities under different configurations

To conclude, as shown in Tables 6.7-6.9 the proposed PGM operator outperforms with confidence the performance of classical PLM when applied to NSGAII and SPEA2 respectively for HV, Spread and Epsilon performance metrics regarding the port2, cardinality constrained portfolio optimization problem (CCPOP).

6.7.3 FTSE100 (89 stocks) port3 in OR-Library

Table 6.10 Port3 – Mean, std, median and iqr for HV, Spread and Epsilon

Problem		NSGAII		SPEA2	
		PGM	PLM	PGM	PLM
PORT3	HV. Mean and Std	9.72e-01$_{4.6e-03}$	9.45e-01$_{1.2e-02}$	9.64e-01$_{5.9e-03}$	9.42e-01$_{8.8e-03}$
	HV. Median and IQR	9.71e-01$_{4.9e-03}$	9.46e-01$_{1.9e-02}$	9.66e-01$_{7.7e-03}$	9.44e-01$_{1.3e-02}$
	SPREAD. Mean and Std	9.07e-01$_{9.3e-03}$	1.05e+00$_{3.5e-02}$	9.20e-01$_{1.6e-02}$	1.07e+00$_{2.5e-02}$
	SPREAD. Median and IQR	9.05e-01$_{1.2e-02}$	1.05e+00$_{6.3e-02}$	9.20e-01$_{2.3e-02}$	1.07e+00$_{3.8e-02}$
	EPSILON. Mean and Std	3.72e-05$_{6.0e-06}$	7.28e-05$_{1.6e-05}$	4.70e-05$_{7.8e-06}$	7.69e-05$_{1.2e-05}$
	EPSILON. Median and IQR	3.78e-05$_{6.5e-06}$	7.15e-05$_{2.5e-05}$	4.54e-05$_{1.0e-05}$	7.40e-05$_{1.7e-05}$

Table 6.11 Port3 – Boxplots for HV, Spread and Epsilon

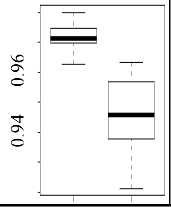
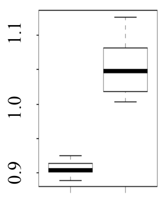
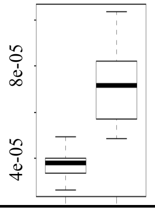
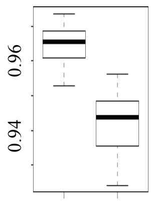
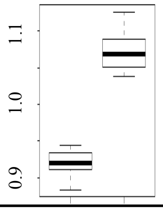
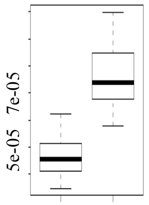
Problem	NSGAII						SPEA2					
	HV		SPREAD		EPSILON		HV		SPREAD		EPSILON	
	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM
PORT3												

Table 6.12 Wilcoxon test in Port3 for HV, Spread and Epsilon

Problem			NSGAII with PLM		SPEA2 with PLM
PORT3	HV. Mean and Std	NSGAII with PGM	▲	SPEA2 with PGM	▲
	HV. Median and IQR		▲		▲
	SPREAD. Mean and Std		▲		▲
	SPREAD. Median and IQR		▲		▲
	EPSILON. Mean and Std		▲		▲
	EPSILON. Median and IQR		▲		▲

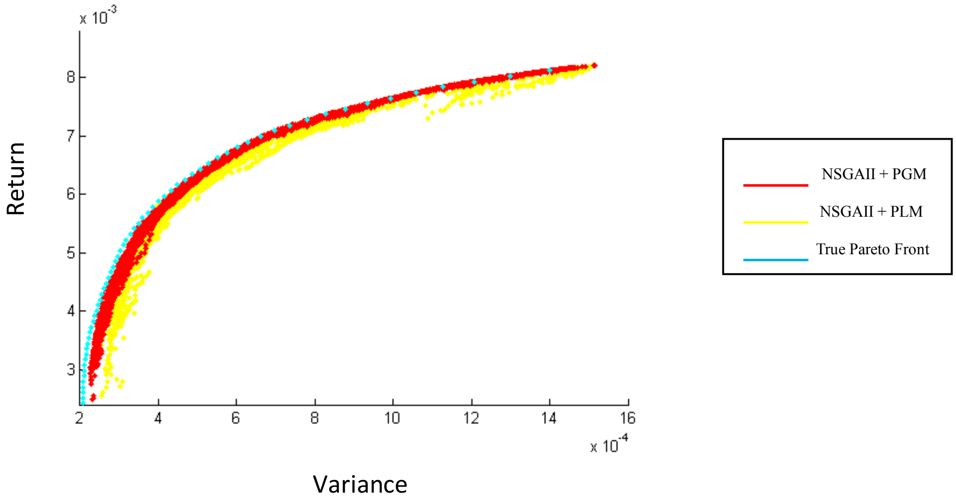


Figure 6.10 NSGA-II: Mean-Variance Efficient Frontier for the port3 problem, with n = 89 securities under different configurations.

The port3 problem uses data from FTSE100 index in the UK. Fig. 6.10 illustrates the approximate efficient frontier under two different configurations: the NSGAII + PGM and the NSGAII + PLM for the port3 test problem together with the exact efficient points generated by using CPLEX version 12.5. The approximate frontiers that are illustrated in Fig.6.10, 6.11, 6.12 and 6.13, have been created by joining the approximate sets for the 20 independent runs of the algorithm without removing the dominated solutions. From Fig. 6.10 we can see that the proposed configuration of the NSGAII with the PGM clearly outperforms the typical configuration of the algorithm with the PLM, as the approximate frontier that is generated by the PLM is dominated by the corresponding approximate frontier that is produced by using the PGM. Fig. 6.10 clearly illustrates that the NSGAII combined with the PGM achieves a remarkable concentration of the search effort very close to the true efficient frontier. Fig.6.11 and 6.12 illustrate separately the approximate frontiers for the two different configurations of the NSGAII.

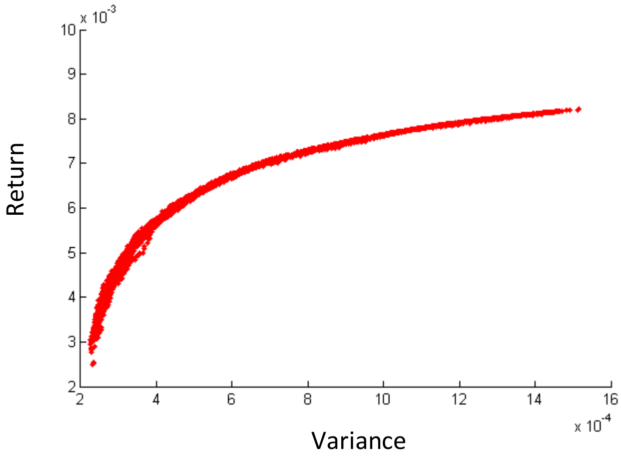


Figure 6.11 NSGAII + PGM: Mean-Variance Efficient Frontier for the port3 problem, with n = 89 securities

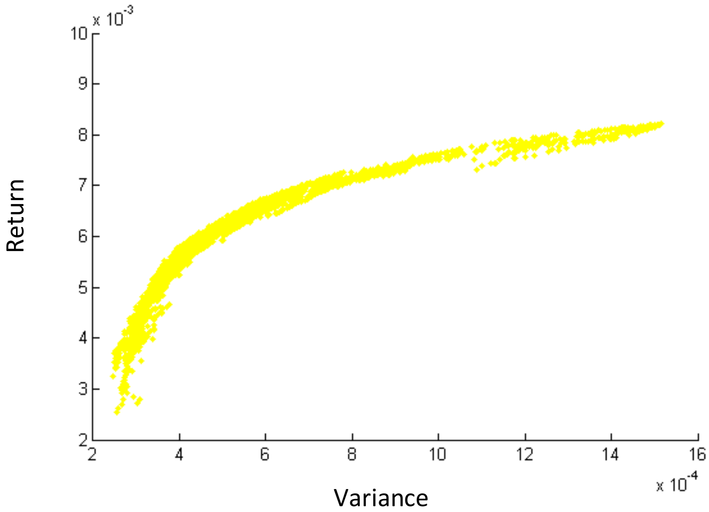


Figure 6.12 NSGAII + PLM: Mean-Variance Efficient Frontier for the port3 problem, with n = 89 securities

Fig.6.12, reveals that the configuration of the NSGAII with the classical PLM operator lacks focus and does not demonstrates the same exploratory strength as when the NSGAII combined with the proposed PGM operator. As shown by Fig.6.12, a considerable part of the computational effort has been wasted in discovering suboptimal solutions and on top of that the approximate frontier presents discontinuities, especially for higher levels of return and risk combinations.

Finally, Fig. 6.13 illustrates the approximate efficient frontier under the SPEA2 + PGM and the SPEA2 + PLM configuration for the port3 test problem together with the exact efficient points generated by using CPLEX 12.5. Fig. 6.13 confirms all previous findings.

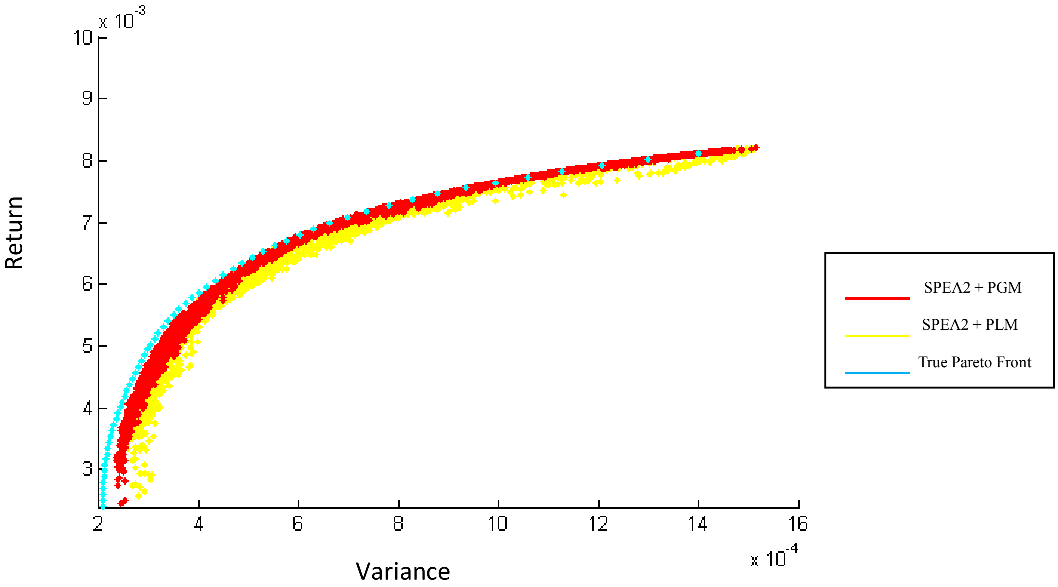


Figure 6.13 SPEA2: Mean-Variance Efficient Frontier for the port3 problem, with n = 89 securities under different configurations

To conclude, the port3 problem uses data from FTSE100 index in the UK. As shown in Tables 6.10-6.12 the proposed PGM operator outperforms with confidence the performance of classical PLM when applied to NSGAII and SPEA2 respectively for all three performance metrics.

6.7.4 S&P100 (98 stocks) port4 in OR-Library

Table 6.13 Port4 – Mean, std, median and iqr for HV, Spread and Epsilon

Problem		NSGAII		SPEA2	
		PGM	PLM	PGM	PLM
PORT4	HV. Mean and Std	9.83e-01$_{4.5e-03}$	9.70e-01$_{3.5e-03}$	9.80e-01$_{3.1e-03}$	9.69e-01$_{3.5e-03}$
	HV. Median and IQR	9.84e-01$_{5.0e-03}$	9.70e-01$_{4.6e-03}$	9.81e-01$_{5.1e-03}$	9.69e-01$_{4.3e-03}$
	SPREAD. Mean and Std	9.24e-01$_{1.8e-02}$	1.04e+00$_{2.8e-02}$	9.46e-01$_{2.0e-02}$	1.06e+00$_{2.9e-02}$
	SPREAD. Median and IQR	9.26e-01$_{2.1e-02}$	1.04e+00$_{3.6e-02}$	9.46e-01$_{2.9e-02}$	1.06e+00$_{4.1e-02}$
	EPSILON. Mean and Std	4.71e-05$_{1.3e-05}$	8.47e-05$_{9.9e-06}$	5.53e-05$_{8.8e-06}$	8.73e-05$_{9.7e-06}$
	EPSILON. Median and IQR	4.56e-05$_{1.4e-05}$	8.34e-05$_{1.3e-05}$	5.42e-05$_{1.4e-05}$	8.84e-05$_{1.2e-05}$

Table 6.14 Port4 – Boxplots for HV, Spread and Epsilon

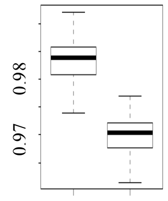
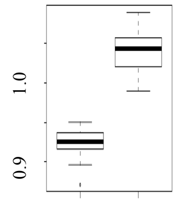
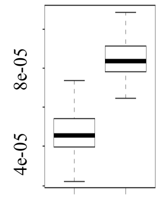
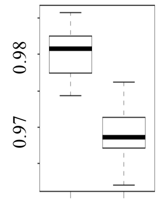
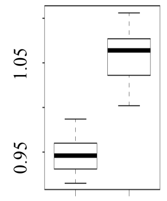
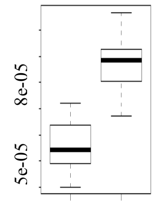
Problem	NSGAII						SPEA2					
	HV		SPREAD		EPSILON		HV		SPREAD		EPSILON	
	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM
PORT4												

Table 6.15 Wilcoxon test in Port4 for HV, Spread and Epsilon

Problem			NSGAII with PLM		SPEA2 with PLM
PORT4	HV. Mean and Std	NSGAII with PGM	▲	SPEA2 with PGM	▲
	HV. Median and IQR		▲		▲
	SPREAD. Mean and Std		▲		▲
	SPREAD. Median and IQR		▲		▲
	EPSILON. Mean and Std		▲		▲
	EPSILON. Median and IQR		▲		▲

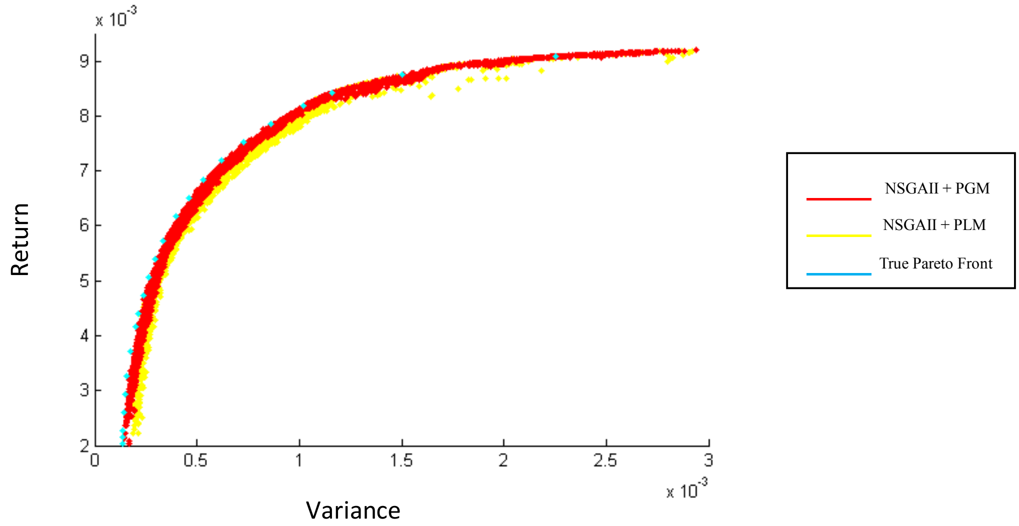


Figure 6.14 NSGA-II: Mean-Variance Efficient Frontier for the port4 problem, with n = 98 securities under different configurations.

The port4 problem uses data from S&P100 index in the USA. Fig. 6.14 illustrates the approximate efficient frontier under two different configurations: the NSGAII + PGM and the NSGAII + PLM for the port4 test problem together with the exact efficient points generated by using CPLEX 12.5. The approximate frontier that is illustrated in Fig.6.14 has been created by joining the approximate sets for the 20 independent runs of the algorithm without removing the dominated solutions. From Fig.6.14 we can see that the proposed configuration of the NSGAII with the PGM clearly outperforms the typical configuration of the algorithm with the PLM, as the approximate frontier that is generated by the PLM is dominated by the corresponding approximate frontier that is produced by using the PGM. On top of everything else, the points of the approximate frontier that are generated with the assistance of PGM overlap with the points of the true efficient frontier, indicating that the proposed methodology (PGM) is capable of producing solutions for the CCPOP as good as if they were calculated with exact methods.

To conclude, regarding the solution of the port4 problem as shown in Tables 6.13-6.15 the proposed PGM outperforms with confidence the performance of the classical PLM for all performance metrics when applied to NSGAII and SPEA2.

6.7.5 Nikkei225 (225 stocks) port5 in OR-Library

Table 6.16 Port5 – Mean, std, median and iqr for HV, Spread and Epsilon

Problem		NSGAII		SPEA2	
		PGM	PLM	PGM	PLM
PORT5	HV. Mean and Std	9.61e-01$_{9.9e-03}$	9.19e-01$_{1.9e-02}$	9.54e-01$_{1.6e-02}$	9.15e-01$_{3.1e-02}$
	HV. Median and IQR	9.63e-01$_{1.8e-02}$	9.22e-01$_{3.9e-02}$	9.60e-01$_{2.7e-02}$	9.16e-01$_{5.1e-02}$
	SPREAD. Mean and Std	1.04e+00$_{4.0e-02}$	1.20e+00$_{3.5e-02}$	1.09e+00$_{2.9e-02}$	1.22e+00$_{2.9e-02}$
	SPREAD. Median and IQR	1.04e+00$_{7.9e-02}$	1.19e+00$_{5.9e-02}$	1.09e+00$_{4.5e-02}$	1.23e+00$_{3.4e-02}$
	EPSILON. Mean and Std	5.24e-05$_{1.3e-05}$	1.11e-04$_{2.5e-05}$	6.29e-05$_{2.1e-05}$	1.15e-04$_{4.1e-05}$
	EPSILON. Median and IQR	5.04e-05$_{2.4e-05}$	1.06e-04$_{4.7e-05}$	5.39e-05$_{3.2e-05}$	1.13e-04$_{6.8e-05}$

Table 6.17 Port5 – Boxplots for HV, Spread and Epsilon

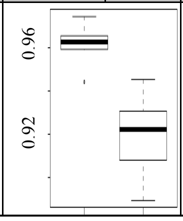
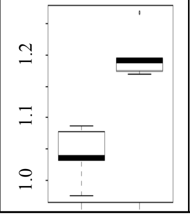
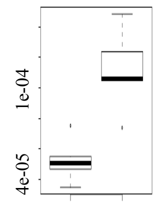
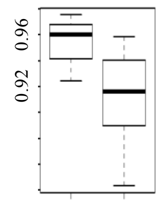
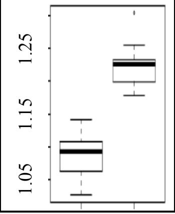
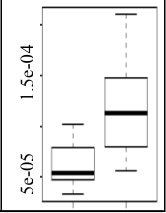
Problem	NSGAII						SPEA2					
	HV		SPREAD		EPSILON		HV		SPREAD		EPSILON	
	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM
PORT5												

Table 6.18 Wilcoxon test in Port5 for HV, Spread and Epsilon

Problem			NSGAII with PLM		SPEA2 with PLM
PORT5	HV. Mean and Std	NSGAII with PGM	▲	SPEA2 with PGM	▲
	HV. Median and IQR		▲		▲
	SPREAD. Mean and Std		▲		▲
	SPREAD. Median and IQR		▲		▲
	EPSILON. Mean and Std		▲		▲
	EPSILON. Median and IQR		▲		▲

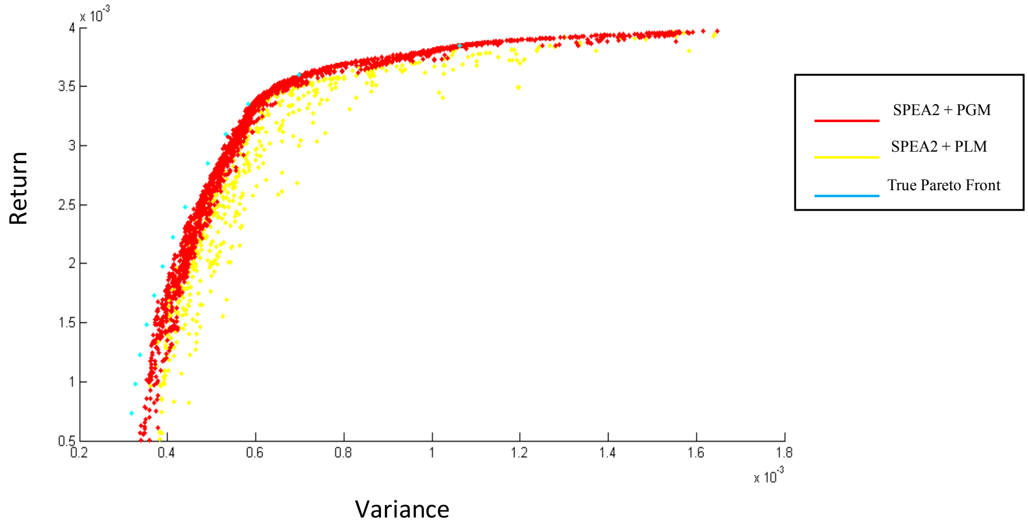


Figure 6.15 SPEA2: Mean-Variance Efficient Frontier for the port5 problem, with n = 225 securities under different configurations.

The port5 problem uses data from Nikkei225 index in Japan. Fig. 6.15 illustrates the approximate efficient frontier under two different configurations: the SPEA2 + PGM and the SPEA2 + PLM for the port5 test problem together with the exact efficient points generated by using CPLEX 12.5. The approximate frontier that is illustrated in Fig.15 has been created by joining the approximate sets for the 20 independent runs of the algorithm without removing the dominated solutions. From Fig.6.15 we can see that the proposed configuration of the SPEA2 with the PGM outperforms the typical configuration of the algorithm with the PLM, as the approximate frontier that is generated by the PLM is dominated by the corresponding approximate frontier that is produced by using the PGM. Finally, please notice that the points of the approximate frontier that are generated with the assistance of PGM are very close to the exact points of the true efficient frontier, meaning that the results of the proposed methodology (PGM) are comparable to the ones that are generated by using exact methods.

Fig.6.16 and 6.17 illustrate separately the approximate frontiers for the two different configurations of the SPEA2. Fig. 6.16 illustrates that the SPEA2 when combined with the PGM succeed to concentrate the search effort towards the higher fitness regions of the search space, as the vast majority of points of the 20 independent runs lie on the approximate efficient frontier.

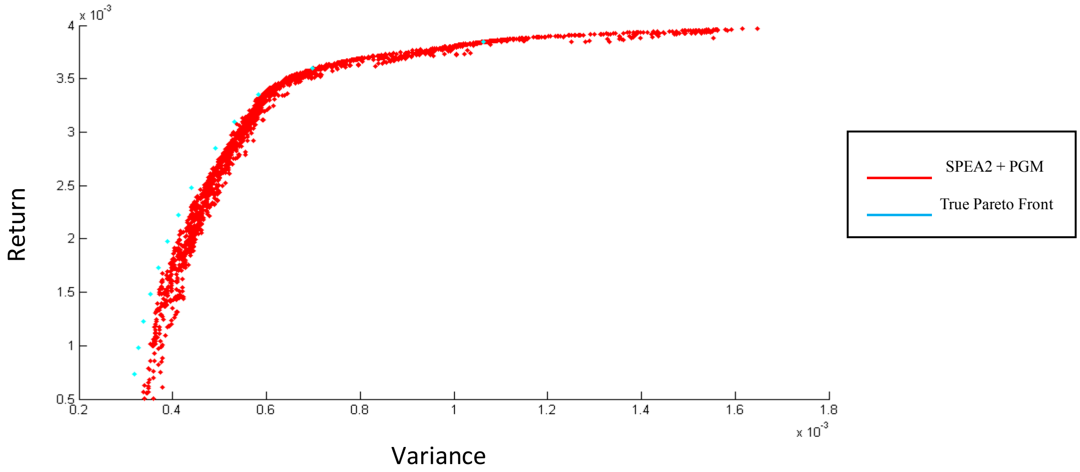


Figure 6.16 SPEA2 + PGM: Mean-Variance Efficient Frontier for the port5 problem, with n = 225 securities.

On the other hand, as shown by Fig.6.17 in the case of the classical PLM, a considerable part of the computational effort is wasted in discovering suboptimal solutions and on top of that the approximate frontier presents gaps, especially for higher levels of return and risk combinations.

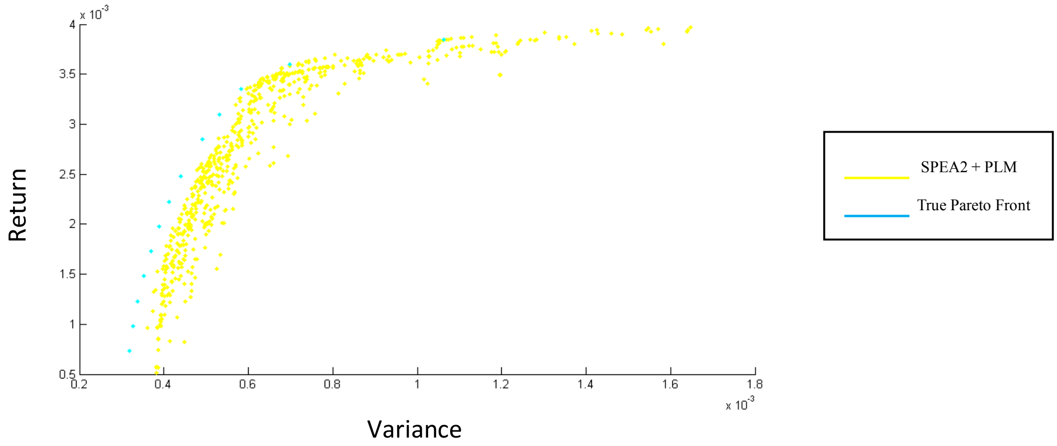


Figure 6.17 SPEA2 + PLM: Mean-Variance Efficient Frontier for the port5 problem, with n = 225 securities

To conclude, the port5 problem uses data from Nikkei225 index in Japan. The relevant results in Tables 6.16-6.18 indicate that the proposed methodology generates better results with confidence for all performance metrics, when applied to the NSGAII and SPEA2.

6.7.6 S&P500 (457 stocks) port6 in OR-Library

Table 6.19 Port6 – Mean, std, median and iqr for HV, Spread and Epsilon

Problem		NSGAII		SPEA2	
		PGM	PLM	PGM	PLM
PORT6	HV. Mean and Std	7.90e-01$_{1.6e-03}$	7.38e-01$_{1.4e-02}$	7.88e-01$_{4.1e-03}$	7.28e-01$_{4.0e-02}$
	HV. Median and IQR	7.89e-01$_{3.7e-03}$	7.39e-01$_{3.3e-02}$	7.87e-01$_{9.9e-03}$	7.48e-01$_{9.3e-02}$
	SPREAD. Mean and Std	7.76e-01$_{1.4e-01}$	1.14e+00$_{5.6e-02}$	8.60e-01$_{9.1e-03}$	1.39e+00$_{1.4e-01}$
	SPREAD. Median and IQR	7.81e-01$_{3.5e-01}$	1.17e+00$_{1.3e-01}$	8.54e-01$_{2.0e-02}$	1.40e+00$_{3.4e-01}$
	EPSILON. Mean and Std	3.66e-04$_{8.8e-05}$	1.28e-03$_{1.5e-04}$	4.65e-04$_{1.5e-04}$	1.90e-03$_{1.1e-03}$
	EPSILON. Median and IQR	3.59e-04$_{2.2e-04}$	1.22e-03$_{3.4e-04}$	3.67e-04$_{3.2e-04}$	1.57e-03$_{2.5e-03}$

Table 6.20 Port6 – Boxplots for HV, Spread and Epsilon

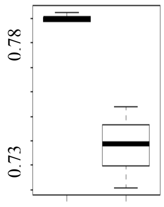
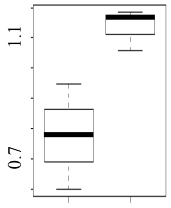
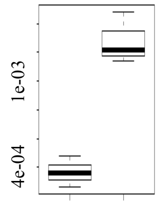
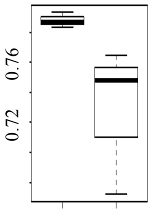
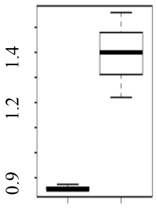
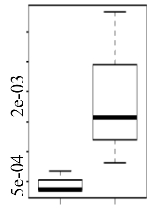
Problem	NSGAII			SPEA2		
	HV	SPREAD	EPSILON	HV	SPREAD	EPSILON
	PGM | PLM	PGM | PLM	PGM | PLM	PGM | PLM	PGM | PLM	PGM | PLM
PORT6						

Table 6.21 Wilcoxon test in Port6 for HV, Spread and Epsilon

Problem			NSGAII with PLM		SPEA2 with PLM
PORT6	HV. Mean and Std	NSGAII with PGM	▲	SPEA2 with PGM	▲
	HV. Median and IQR		▲		▲
	SPREAD. Mean and Std		▲		▲
	SPREAD. Median and IQR		▲		▲
	EPSILON. Mean and Std		▲		▲
	EPSILON. Median and IQR		▲		▲

The port6 problem uses data from S&P500 index in the USA. As shown in Tables 6.19-6.21 the proposed PGM outperforms with confidence the performance of the classical PLM for all performance metrics when applied to NSGAII and SPEA2.

6.8 Analysis of the Results

In this section, we analyze the results obtained by applying the Probe Guided Mutation (PGM) operator and the Polynomial Mutation (PLM) operator respectively to the NSGAII and SPEA2. Three well known performance indicators of MOEAs namely *Hypervolume*, *Spread* and *Epsilon* have been applied to assess the quality of the proposed variation operator.

Examining the results (Tables 6.4, 6.7, 6.10, 6.13, 6.16 and 6.19) of the first indicator, the *HV*, which is considered by many in the relevant literature [82], [117], [176], [228], [234] as the most representative indicator of the performance of a MOEA we notice that both the NSGAII and SPEA2 perform better with the application of PGM operator for all test problems examined (port1-6), compared with results derived by applying the classical PLM operator. The Wilcoxon rank-sum test (see Tables 6.6, 6.9, 6.12, 6.15, 6.18 and 6.21) validates that the observed difference in PGM and PLM performance is statistically significant with 95% confidence.

Regarding the *Spread indicator* the same methodology was applied. The Wilcoxon rank-sum test helped us to identify that in all cases the PGM yields better results with confidence than the conventional configuration of NSGAII and SPEA2 with the PLM operator.

Finally, regarding the *Epsilon* indicator, the Wilcoxon rank-sum test helped us to identify that in all test instances the PGM yields better results with confidence than the conventional configuration of NSGAII and SPEA2 with the PLM operator. We should highlight that the PLM in none of the examined test problems (port1-6) outperformed the PGM operator in any one of the three examined performance metrics, namely HV, Spread and Epsilon indicator.

On the downsides of the proposed methodology is the computational cost associated with the fitness evaluation process of the PGM operator. Table 6.1 and 6.2 provide indicative CPU times, in seconds, needed for each one of the presented tests problems (port1-5) under the classical configuration with the PLM operator and under the proposed PGM. In general, the overhead of the proposed methodology is not stable but depends on the objective functions of each particular problem. However, the computational cost associated with the PGM operator is well compensated by the higher quality results. Indeed, as indicated by the approximate efficient frontiers (see

Fig. 6.5-6.17) the proposed methodology (PGM) generates better results than the conventional PLM operator both in terms of Pareto domination conditions and in terms of the spread of solutions across the frontier. In particular the approximate frontiers that are generated by the classical PLM produce a considerable number of points far away from the true efficient frontier and on top of that they present discontinuities especially for the higher levels of risk and return combinations. On the contrary, the approximate efficient frontiers that are generated by the proposed PGM operator demonstrate a remarkable concentration of the search effort very close or in some cases on top of the true efficient frontier (CPLEX).

Below, we provide a last test in order to demonstrate the exploratory capabilities of the proposed methodology. In particular, we revisit the port2 test problem, but this time we use *40,000* evaluations for the PGM and *200,000* evaluations for the PLM to count for the extra fitness functions evaluations introduced by the proposed methodology. Any other parameter remains the same as it was described in section 6.5.1.

6.8.1 DAX100 (85 stocks) port2 in OR-Library (PGM: *40,000 evaluations* / PLM: *200,000 evaluations*)

Table 6.22 Port2 – Mean, std, median and iqr for HV, Spread and Epsilon

Problem		NSGAII		SPEA2	
		PGM (Eval: 40,000)	PLM (Eval: 200,000)	PGM (Eval: 40,000)	PLM (Eval: 200,000)
PORT2	HV. Mean and Std	9.85e-01$_{3.3e-03}$	9.71e-01$_{5.8e-03}$	9.82e-01$_{3.3e-03}$	9.68e-01$_{5.3e-03}$
	HV. Median and IQR	9.85e-01$_{2.9e-03}$	9.68e-01$_{9.2e-03}$	9.84e-01$_{3.2e-03}$	9.70e-01$_{4.5e-03}$
	SPREAD. Mean and Std	9.76e-01$_{1.7e-02}$	1.05e+00$_{4.8e-02}$	1.00e+00$_{1.2e-02}$	1.09e+00$_{3.1e-02}$
	SPREAD. Median and IQR	9.76e-01$_{2.2e-02}$	1.03e+00$_{9.2e-02}$	1.00e+00$_{1.9e-02}$	1.10e+00$_{4.8e-02}$
	EPSILON. Mean and Std	4.17e-05$_{8.9e-06}$	7.73e-05$_{1.6e-05}$	4.74e-05$_{8.9e-06}$	8.53e-05$_{1.4e-05}$
	EPSILON. Median and IQR	4.04e-05$_{7.8e-06}$	8.56e-05$_{2.5e-05}$	4.44e-05$_{8.7e-06}$	8.17e-05$_{1.2e-05}$

Table 6.23 Port2 – Boxplots for HV, Spread and Epsilon

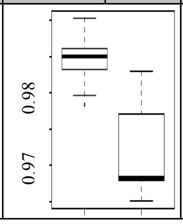
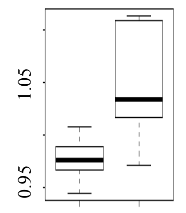
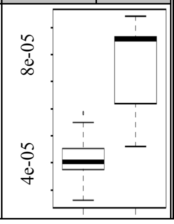
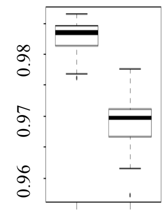
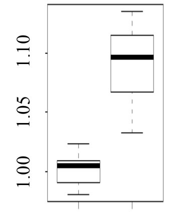
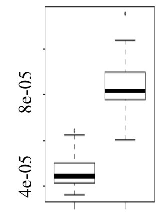
Problem	NSGAII						SPEA2					
	HV		SPREAD		EPSILON		HV		SPREAD		EPSILON	
	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM
PORT2												

Table 6.24 Wilcoxon test in Port2 for HV, Spread and Epsilon

Problem			NSGAII with PLM		SPEA2 with PLM
PORT2	HV. Mean and Std	NSGAII with PGM	▲	SPEA2 with PGM	▲
	HV. Median and IQR		▲		▲
	SPREAD. Mean and Std		▲		▲
	SPREAD. Median and IQR		▲		▲
	EPSILON. Mean and Std		▲		▲
	EPSILON. Median and IQR		▲		▲

Based on the results in Tables 6.22-6.24 the proposed PGM operator outperforms with confidence the performance of classical PLM for the port2 problem when applied to NSGAII and SPEA2 respectively for all three performance metrics, although we only use *40,000* evaluations for the PGM compared to *200,000* evaluations for the PLM. This experiment proved that the proposed methodology (PGM) is able to generate higher quality results than the classical PLM for only a fraction of the evaluations used by the PLM operator.

The experimental results in section 6.7 have been produced by assuming *100,000* function evaluations for both (PGM and PLM) configurations of the algorithm. However, due to the the fitness evaluation process introduced in the PGM operator the total number of the actual fitness evaluations for the configuration of the algorithm with the PGM operator is higher than *100,000*. Specifically, for the port2 test problem, we have estimated that for every *100* evaluations of the algorithm the PGM is executed on average *79* times, given that the mutation probability is $P_m = 1/n$, where n is the number of decision variables. Furthermore, each execution of the PGM entails two (2) separate fitness functions evaluations. Thus, for each 100 evaluations of the algorithm, the PGM entails on average *2x79 = 158* additional fitness functions evaluations. Based on the above, someone might think that for reasons of fairness we should allow some extra evaluations for the PLM when compared with the PGM operator. For example for each 100,000 evaluations of the PGM we should allow *258,000 (= 100,000 + 100,000 x 1.58)* evaluations of the PLM. Is that so? In order to answer that question we will make use of the NSGAII flowchart. As shown from the NSGAII flowchart a complete generational circle implies not only equal number of fitness functions evaluations but also equal number in every other aspect/ component of the NSGAII, like for example number of crossovers (SBX).Thus by comparing *100,000* evaluations of the NSGAII+PGM is not equivalent in any case with *258,000* evaluations of the NSGAII+PLM.

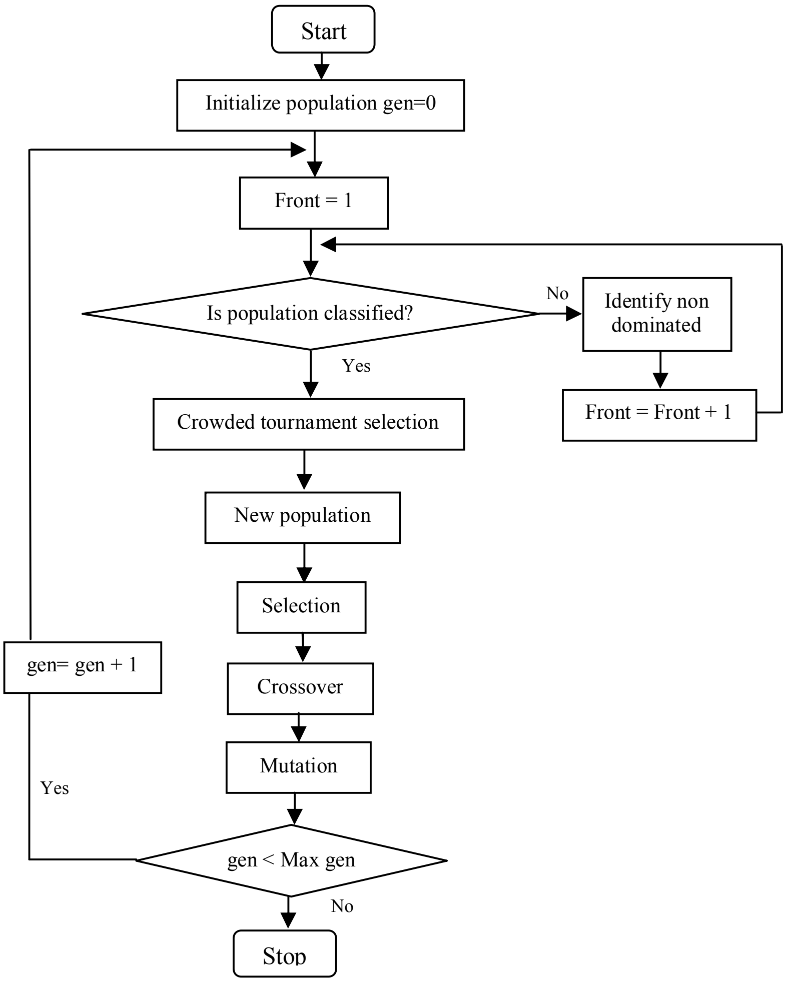


Figure 6.18 Flow chart of the NSGAII

For that reason we opted to use equal number of evaluations for each configuration of the algorithm and we cited separately the additional computational cost that the proposed methodology entails.

However, because we wanted to shed any doubt regarding the efficacy of the proposed PGM operator, we re-examined the port2 problem, but this time by using *40,000* evaluations for the configuration of NSGAII and SPEA2 with the PGM and *200,000* evaluations for the configuration of NSGAII and SPEA2 with the PLM. The *40,000* evaluations for the configuration of the algorithm with the PGM correspond to *103,200* (*=40,000 + 40,000x1.58*) actual fitness functions evaluations, which is almost half of the evaluations used for the PLM. The relevant results for the above configuration of the algorithms, are shown in Tables 6.22-6.24. The PGM clearly outperforms with confidence the performance of classical PLM for the port2 problem when applied to NSGAII and SPEA2 respectively for all three performance metrics, although we only use a fraction of the evaluations that are being used for the PLM. We also tested the proposed methodology (PGM) for only 20,000 evaluations (51,600 actual fitness functions evaluations) against to 200,000 evaluation of the algorithm with the PLM operator and we found out that the PGM still outperforms the classical PLM operator.

6.9 Experimental presentation of the PGM operator with the assistance of the ZDT1-4, 6 and DTLZ1-7 families of test functions

6.9.1 Experimental Environment

All algorithms have been implemented in Java and run on a personal computer Core 2 Duo at 1.83 GHz. The jMetal [77] framework has been used to compare the performance of the proposed, Probe Guided mutation operator against the classical polynomial mutation operator with the assistance of two state-of-the-art MOEAs, namely NSGAII and SPEA2. In all tests we use, binary tournament and simulated binary crossover (SBX) [54] as, selection and crossover operator, respectively. The crossover probability is $P_c = 0.9$ and mutation probability is $P_m = 1/n$, where n is the number of decision variables. The distribution indices for the crossover and mutation operators are $\eta_c = 20$ and $\eta_m = 20$, respectively. Population size is set to *100*, using *25,000* function evaluations with *100* independent runs.

Obviously, there is a computational cost associated with the fitness evaluation process of PGM. Table 6.25 provides the total CPU time, in seconds, needed for each one of the presented tests problems (ZDT1-6, DTLZ 1-7) under the typical configuration with the polynomial mutation operator and under the proposed PGM. The hardware used was a personal computer Core 2 Duo at 1.83 GHz. On average the proposed PGM has an overhead of 3.62% in CPU time when applied to the NSGAII and 13.94% when applied to SPEA2.

Table 6.25 Mean and std total cpu times (in seconds) for the test problems for 100 independent tuns

Problem	NSGAII			SPEA2		
	PGM	PLM	Overhead	PGM	PLM	Overhead
ZDT1	1.8020e+00$_{4.2426e-02}$	1.7402e+00$_{2.8747e-02}$	3.55%	9.9715e+00$_{1.7607e-01}$	8.3176e+00$_{1.23285e-01}$	19.88%
ZDT2	1.7581e+00$_{2.8280e-02}$	1.7164e+00$_{1.18905e-01}$	2.42%	9.5751e+00$_{2.3335e-01}$	7.728e+00$_{2.70603e-01}$	23.90%
ZDT3	1.7543e+00$_{2.1210e-02}$	1.7390e+00$_{4.4716e-02}$	0.88%	9.7354e+00$_{2.3971e-01}$	8.4512e+00$_{1.40685e-01}$	15.19%
ZDT4	1.5067e+00$_{2.9703e-02}$	1.4149e+00$_{3.6598e-02}$	6.48%	6.1209e+00$_{5.0487e-01}$	5.3914e+00$_{2.98521e-01}$	13.53%
ZDT6	1.3728e+00$_{2.4041e-02}$	1.4032e+00$_{5.9322e-02}$	-2.16%	6.6631e+00$_{4.5679e-01}$	5.4370e+00$_{1.01872e-01}$	22.55%
DTLZ1	1.3721e+00$_{2.6870e-02}$	1.3632e+00$_{4.7347e-02}$	0.65%	8.0720e+00$_{9.3904e-01}$	7.3504e+00$_{6.53571e-01}$	9.81%
DTLZ2	1.4662e+00$_{6.9304e-02}$	1.4331e+00$_{3.1768e-02}$	2.30%	18.5152e+00$_{8.3440e-02}$	16.1554e+00$_{1.80169e-01}$	14.60%
DTLZ3	1.5414e+00$_{3.8183e-02}$	1.4683e+00$_{6.0019e-02}$	4.97%	5.2613e+00$_{2.5460e-02}$	5.3094e+00$_{2.079e-01}$	-0.90%
DTLZ4	1.7137e+00$_{4.8083e-02}$	1.5441e+00$_{1.07981e-01}$	10.98%	17.2105e+00$_{1.1102e-01}$	15.013e+00$_{2.1727e+00}$	14.63%
DTLZ5	1.4275e+00$_{3.1823e-02}$	1.3638e+00$_{1.8984e-02}$	4.67%	13.9080e+00$_{1.6829e-01}$	11.9748e+00$_{3.08771e-01}$	16.14%
DTLZ6	1.7225e+00$_{8.1324e-02}$	1.6217e+00$_{1.02639e-01}$	6.21%	9.6712e+00$_{1.8526e-01}$	8.4328e+00$_{9.0276e-02}$	14.68%
DTLZ7	1.7594e+00$_{4.3840e-02}$	1.7158e+00$_{2.14047e-01}$	2.51%	13.3275e+00$_{4.1931e-01}$	12.8962e+00$_{3.72519e-01}$	3.34%
		Average:	**3.62%**		**Average:**	**13.94%**

6.9.2 Experimental Results

A number of computational experiments [72] were performed to test the performance of the proposed Probe Guided mutation operator for the solution of two different sets of test functions, namely ZDT1-4, 6 [229] and DTLZ1-7 [60]. The performance of the proposed PGM operator is assessed in comparison with the classical Polynomial Mutation operator with the assistance of two well-known MOEAs, namely the Non-dominated Sorting Genetic Algorithm II (NSGAII) and Strength Pareto Evolutionary Algorithm 2 (SPEA2). The evaluation of the performance is based on a variety of metrics that assess both the proximity of the solutions to the Pareto front and their dispersion on it.

6.9.2.1 The Zitzler-Deb-Theile (ZDT) test suite

The Zitzler-Deb-Theile (ZDT) test suite [229] is widely used for evaluating algorithms solving MOPs. The following five bi-objective MOPs named ZDT1, ZDT2, ZDT3, ZDT4 and ZDT6 were used for comparing the proposed PGM operator against the polynomial mutation (PLM). They have been used extensively for testing MOEAs and their Pareto front shapes are convex, nonconvex, disconnected, multimodal and non-uniform. ZDT1, ZDT2 and ZDT3 use 30 decision variables and ZDT4 and ZDT6 use 10 decision variables respectively.

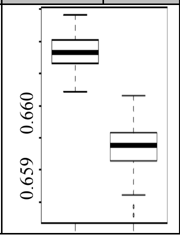
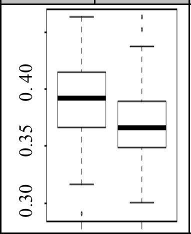
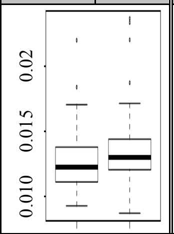
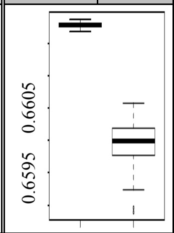
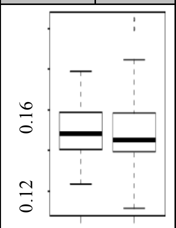
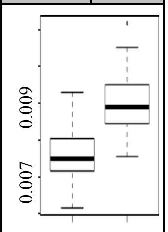
Zitzler-Deb-Thiele's function N.1 (ZDT1) problem:

$$Minimize = \begin{cases} f_1(x) = x_1 \\ f_2(x) = g(x)h(f_1(x), g(x)) \\ g(x) = 1 + \frac{9}{29}\sum_{i=2}^{30} x_i \\ h\big(f_1(x), g(x)\big) = 1 - \sqrt{\frac{f_1(x)}{g(x)}} \end{cases} \quad for\ 0 \le x_i \le 1\ and\ 1 \le i \le 30$$

Table 6.26 ZDT1 – Mean, std, median and iqr for HV, Spread and Epsilon – NSGAII and SPEA2

Problem		NSGAII		SPEA2	
		PGM	PLM	PGM	PLM
ZDT1	HV. Mean and Std	6.61e-01$_{2.4e-04}$	6.59e-01$_{3.4e-04}$	6.62e-01$_{4.1e-05}$	6.60e-01$_{3.3e-04}$
	HV. Median and IQR	6.61e-01$_{3.7e-04}$	6.59e-01$_{4.4e-04}$	6.62e-01$_{5.5e-05}$	6.60e-01$_{4.2e-04}$
	SPREAD. Mean and Std	3.89e-01$_{3.5e-02}$	3.71e-01$_{3.1e-02}$	1.49e-01$_{1.3e-02}$	1.49e-01$_{1.6e-02}$
	SPREAD. Median and IQR	3.92e-01$_{4.9e-02}$	3.66e-01$_{4.1e-02}$	1.48e-01$_{1.8e-02}$	1.45e-01$_{1.9e-02}$
	EPSILON. Mean and Std	1.26e-02$_{2.1e-03}$	1.34e-02$_{2.4e-03}$	7.61e-03$_{5.9e-04}$	9.01e-03$_{7.8e-04}$
	EPSILON. Median and IQR	1.22e-02$_{2.7e-03}$	1.30e-02$_{2.4e-03}$	7.50e-03$_{8.7e-04}$	8.90e-03$_{1.1e-03}$

Table 6.27 ZDT1 – Boxplots for HV, Spread and Epsilon – NSGAII and SPEA2

Problem	NSGAII						SPEA2					
	HV		SPREAD		EPSILON		HV		SPREAD		EPSILON	
	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM
ZDT1												

The results in the Tables 6.25 – 6.61 have been produced by using JMetal [77] framework, a java implementation of many popular MOEAs that has been used in a considerable number of studies in the field [22], [111], [219], [221]. Table 6.25 presents the results of ZDT1 test function. Specifically, it presents the mean, standard deviation (STD), median and interquartile range (IQR) of all the independent runs carried out for Hypervolume (HV), Spread (Δ) and Epsilon indicator respectively.

Clearly, regarding the HV [233] indicator the higher the value (i.e. the greater the hypervolume) the better the computed front. HV considered by many researchers in the field [82], [117], [176], [228], [234] as the most important indicator, because of its ability to capture in a single number both the closeness of the solutions to the optimal set and to a certain degree, the spread of the solutions across the objective space. The second indicator the Spread (Δ) [57] examines the spread of solutions across the pareto front. The smaller the value of this indicator, the better the distribution of the solutions. This indicator takes a zero value for an ideal distribution of the solutions in the Pareto front. The third indicator, the Epsilon [126] is a measure of the smaller distance that a solution set A, needs to be changed in such a way that it dominates the optimal Pareto front of this problem. Obviously the smaller the value of this indicator, the better the derived solution set.

Table 6.27 use boxplots to present graphically the performance of NSGAII and SPEA2 under two different configurations, PGM and PLM respectively, for the three performance indicators, namely: HV, Spread and Epsilon. Boxplot is a convenient way of graphically depicting groups of numerical data through their quartiles. Boxplots provide a 5-number summary of the data (min, max, Q1, Q3, median) and information about outliers.

Table 6.28, presents if the results of NSGAII and SPEA2 derived under the two different configurations (PGM and PLM) are statistically significant or not. For that reason, we use the Wilcoxon rank-sum test as it is implemented by the jMetal framework [77]. In Table 6.28, three different symbols are used. In particular "–" indicates that there is not statistical significance between the algorithms. "▲" means that the algorithm in the row has yielded better results than the algorithm in the column with confidence and "▽" is used when the algorithm in the column is statistically better than the algorithm in the row. The same interpretation holds for the rest of the Tables of section 6.9.2. Analysis of the relevant results provided in section 6.9.3.

Table 6.28 Wilcoxon test in ZDT1 for HV, Spread and Epsilon

Problem			NSGAII with PLM		SPEA2 with PLM
ZDT1	HV. Mean and Std	NSGAII with PGM	▲	SPEA2 with PGM	▲
	HV. Median and IQR		▲		▲
	SPREAD. Mean and Std		▽		–
	SPREAD. Median and IQR		▽		–
	EPSILON. Mean and Std		▲		▲
	EPSILON. Median and IQR		▲		▲

Zitzler-Deb-Thiele's function N.2 (ZDT2) problem:

$$Minimize = \begin{cases} f_1(x) = x_1 \\ f_2(x) = g(x)h(f_1(x), g(x)) \\ g(x) = 1 + \frac{9}{29}\sum_{i=2}^{30} x_i & for\ 0 \le x_i \le 1\ and\ 1 \le i \le 30 \\ h\big(f_1(x), g(x)\big) = 1 - \left(\frac{f_1(x)}{g(x)}\right)^2 \end{cases}$$

Table 6.29 ZDT2 – Mean, std, median and iqr for HV, Spread and Epsilon – NSGAII and SPEA2

Problem		NSGAII		SPEA2	
		PGM	PLM	PGM	PLM
ZDT2	HV. Mean and Std	$3.28e\text{-}01_{2.1e\text{-}04}$	$3.26e\text{-}01_{3.2e\text{-}04}$	$3.28e\text{-}01_{5.0e\text{-}03}$	$3.26e\text{-}01_{5.6e\text{-}04}$
	HV. Median and IQR	$3.28e\text{-}01_{3.0e\text{-}04}$	$3.26e\text{-}01_{3.7e\text{-}04}$	$3.28e\text{-}01_{7.3e\text{-}05}$	$3.26e\text{-}01_{5.9e\text{-}04}$
	SPREAD. Mean and Std	$3.99e\text{-}01_{3.2e\text{-}02}$	$3.81e\text{-}01_{3.2e\text{-}02}$	$1.49e\text{-}01_{1.2e\text{-}02}$	$1.55e\text{-}01_{1.9e\text{-}02}$
	SPREAD. Median and IQR	$4.04e\text{-}01_{4.0e\text{-}02}$	$3.82e\text{-}01_{3.8e\text{-}02}$	$1.50e\text{-}01_{1.9e\text{-}02}$	$1.54e\text{-}01_{2.1e\text{-}02}$
	EPSILON. Mean and Std	$1.29e\text{-}02_{2.3e\text{-}03}$	$1.30e\text{-}02_{2.1e\text{-}03}$	$7.81e\text{-}03_{2.6e\text{-}03}$	$1.05e\text{-}02_{1.3e\text{-}02}$
	EPSILON. Median and IQR	$1.23e\text{-}02_{2.8e\text{-}03}$	$1.28e\text{-}02_{2.6e\text{-}03}$	$7.53e\text{-}03_{9.0e\text{-}04}$	$8.83e\text{-}03_{9.4e\text{-}04}$

Table 6.30 ZDT2 – Boxplots for HV, Spread and Epsilon – NSGAII and SPEA2

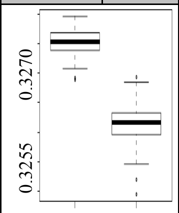
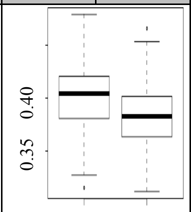
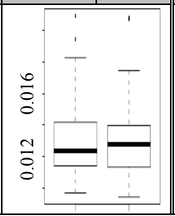
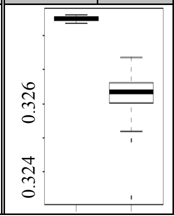
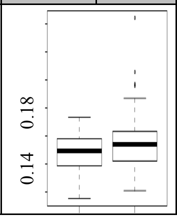
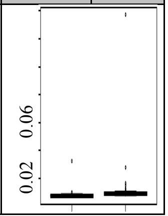
Problem	NSGAII						SPEA2					
	HV		SPREAD		EPSILON		HV		SPREAD		EPSILON	
	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM
ZDT2												

Table 6.31 Wilcoxon test in ZDT2 for HV, Spread and Epsilon

Problem			NSGAII with PLM		SPEA2 with PLM
ZDT2	HV. Mean and Std	NSGAII with PGM	▲	SPEA2 with PGM	▲
	HV. Median and IQR		▲		▲
	SPREAD. Mean and Std		▽		▲
	SPREAD. Median and IQR		▽		▲
	EPSILON. Mean and Std		–		▲
	EPSILON. Median and IQR		–		▲

Zitzler-Deb-Thiele's function N.3 (ZDT3) problem

$$Minimize = \begin{cases} f_1(x) = x_1 \\ f_2(x) = g(x)h(f_1(x), g(x)) \\ g(x) = 1 + \frac{9}{29}\sum_{i=2}^{30} x_i \\ h(f_1(x), g(x)) = 1 - \sqrt{\frac{f_1(x)}{g(x)}} - \left(\frac{f_1(x)}{g(x)}\right)\sin(10\pi f_1(x)) \end{cases} \quad for\ 0 \le x_i \le 1\ and\ 1 \le i \le 30$$

Table 6.32 ZDT3 – Mean, std, median and iqr for HV, Spread and Epsilon – NSGAII and SPEA2

Problem		NSGAII		SPEA2	
		PGM	PLM	PGM	PLM
ZDT3	HV. Mean and Std	5.16e-01$_{8.3e-05}$	5.15e-01$_{3.6e-04}$	5.16e-01$_{4.3e-05}$	5.14e-01$_{6.6e-04}$
	HV. Median and IQR	5.16e-01$_{1.0e-04}$	5.15e-01$_{2.7e-04}$	5.16e-01$_{5.4e-05}$	5.14e-01$_{3.6e-04}$
	SPREAD. Mean and Std	7.50e-01$_{1.5e-02}$	7.45e-01$_{1.4e-02}$	7.06e-01$_{1.0e-02}$	7.11e-01$_{1.6e-02}$
	SPREAD. Median and IQR	7.49e-01$_{2.3e-02}$	7.45e-01$_{1.9e-02}$	7.04e-01$_{3.5e-03}$	7.09e-01$_{7.3e-03}$
	EPSILON. Mean and Std	8.20e-03$_{1.7e-03}$	1.15e-02$_{3.1e-02}$	8.64e-03$_{1.2e-03}$	1.90e-02$_{5.2e-02}$
	EPSILON. Median and IQR	7.88e-03$_{2.2e-03}$	8.19e-03$_{2.0e-03}$	8.42e-03$_{1.5e-03}$	9.38e-03$_{1.9e-03}$

Table 6.33 ZDT3 – Boxplots for HV, Spread and Epsilon – NSGAII and SPEA2

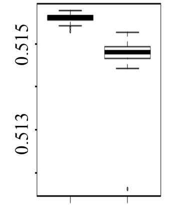
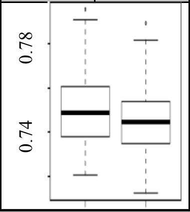
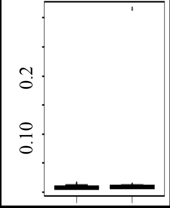
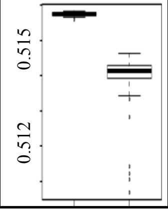
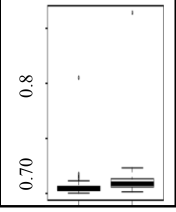
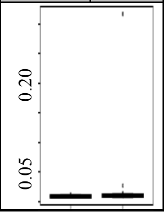
Problem	NSGAII						SPEA2					
	HV		SPREAD		EPSILON		HV		SPREAD		EPSILON	
	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM
ZDT3												

Table 6.34 Wilcoxon test in ZDT3 for HV, Spread and Epsilon

Problem			NSGAII with PLM		SPEA2 with PLM
ZDT3	HV. Mean and Std	NSGAII with PGM	▲	SPEA2 with PGM	▲
	HV. Median and IQR		▲		▲
	SPREAD. Mean and Std		▽		▲
	SPREAD. Median and IQR		▽		▲
	EPSILON. Mean and Std		–		▲
	EPSILON. Median and IQR		–		▲

Zitzler-Deb-Thiele's function N.4 (ZDT4) problem

$$Minimize = \begin{cases} f_1(x) = x_1 \\ f_2(x) = g(x)h(f_1(x), g(x)) \qquad for\ 0 \le x_1 \le 1, -5 \le x_i \le 5,\ 2 \le i \le 10 \\ g(x) = 91 + \sum_{i=2}^{10} (x_i^2 - 10\cos(4\pi x_i)) \\ h\big(f_1(x), g(x)\big) = 1 - \sqrt{\dfrac{f_1(x)}{g(x)}} \end{cases}$$

Table 6.35 ZDT4 – Mean, std, median and iqr for HV, Spread and Epsilon – NSGAII and SPEA2

Problem		NSGAII		SPEA2	
		PGM	PLM	PGM	PLM
ZDT4	HV. Mean and Std	6.58e-01$_{1.6e-03}$	6.54e-01$_{4.3e-03}$	6.55e-01$_{7.1e-03}$	6.48e-01$_{1.4e-02}$
	HV. Median and IQR	6.58e-01$_{1.7e-03}$	6.55e-01$_{5.6e-03}$	6.57e-01$_{4.9e-03}$	6.52e-01$_{1.0e-02}$
	SPREAD. Mean and Std	3.95e-01$_{3.8e-02}$	4.00e-01$_{3.8e-02}$	2.14e-01$_{8.5e-02}$	2.90e-01$_{1.3e-01}$
	SPREAD. Median and IQR	3.93e-01$_{5.0e-02}$	4.01e-01$_{4.6e-02}$	1.83e-01$_{1.1e-01}$	2.59e-01$_{1.8e-01}$
	EPSILON. Mean and Std	1.37e-02$_{2.1e-03}$	1.77e-02$_{1.3e-02}$	4.04e-02$_{4.7e-02}$	5.80e-02$_{6.7e-02}$
	EPSILON. Median and IQR	1.36e-02$_{2.7e-03}$	1.54e-02$_{3.7e-03}$	1.27e-02$_{4.5e-02}$	2.10e-02$_{6.6e-02}$

Table 6.36 ZDT4 – Boxplots for HV, Spread and Epsilon – NSGAII and SPEA2

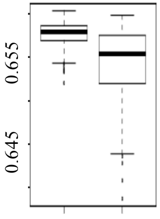
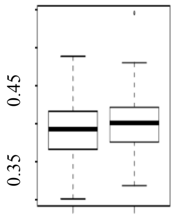
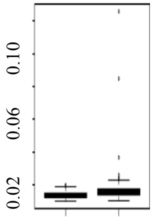
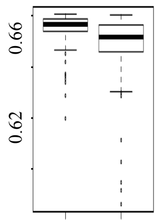
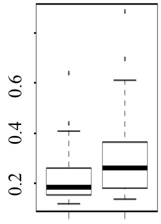
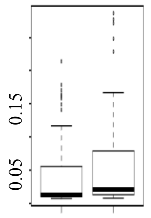
<table>
<tr><th rowspan="3">Problem</th><th colspan="6">NSGAII</th><th colspan="6">SPEA2</th></tr>
<tr><th colspan="2">HV</th><th colspan="2">SPREAD</th><th colspan="2">EPSILON</th><th colspan="2">HV</th><th colspan="2">SPREAD</th><th colspan="2">EPSILON</th></tr>
<tr><th>PGM</th><th>PLM</th><th>PGM</th><th>PLM</th><th>PGM</th><th>PLM</th><th>PGM</th><th>PLM</th><th>PGM</th><th>PLM</th><th>PGM</th><th>PLM</th></tr>
<tr><td>ZDT4</td>
<td colspan="2"> </td>
<td colspan="2"> </td>
<td colspan="2"> </td>
<td colspan="2"> </td>
<td colspan="2"> </td>
<td colspan="2"> </td>
</tr>
</table>

Table 6.37 Wilcoxon test in ZDT4 for HV, Spread and Epsilon

Problem			NSGAII with PLM		SPEA2 with PLM
ZDT4	HV. Mean and Std	NSGAII with PGM	▲	SPEA2 with PGM	▲
	HV. Median and IQR		▲		▲
	SPREAD. Mean and Std		–		▲
	SPREAD. Median and IQR		–		▲
	EPSILON. Mean and Std		▲		▲
	EPSILON. Median and IQR		▲		▲

Zitzler-Deb-Thiele's function N.6 (ZDT6) problem

$$Minimize = \begin{cases} f_1(x) = 1 - \exp(-4x_1) sin^6(6\pi x_1) \\ f_2(x) = g(x) h(f_1(x), g(x)) \\ g(x) = 1 + 9\left[\frac{\sum_{i=2}^{10} x_i}{9}\right]^{0.25} \\ h(f_1(x), g(x)) = 1 - \left(\frac{f_1(x)}{g(x)}\right)^2 \end{cases} \quad for\ 0 \le x_i \le 1\ and\ 1 \le i \le 10$$

Table 6.38 ZDT6 – Mean, std, median and iqr for HV, Spread and Epsilon – NSGAII and SPEA2

Problem		NSGAII		SPEA2	
		PGM	PLM	PGM	PLM
ZDT6	HV. Mean and Std	3.99e-01$_{4.5e-04}$	3.88e-01$_{1.6e-03}$	4.00e-01$_{3.1e-04}$	3.79e-01$_{3.0e-03}$
	HV. Median and IQR	3.99e-01$_{5.7e-04}$	3.89e-01$_{2.2e-03}$	4.00e-01$_{3.8e-04}$	3.79e-01$_{4.7e-03}$
	SPREAD. Mean and Std	5.13e-01$_{6.3e-02}$	3.59e-01$_{3.0e-02}$	1.56e-01$_{1.4e-02}$	2.35e-01$_{8.4e-02}$
	SPREAD. Median and IQR	5.10e-01$_{9.2e-02}$	3.59e-01$_{4.7e-02}$	1.56e-01$_{1.6e-02}$	2.24e-01$_{3.0e-02}$
	EPSILON. Mean and Std	1.15e-02$_{2.7e-03}$	1.43e-02$_{2.0e-03}$	6.26e-03$_{5.1e-04}$	2.52e-02$_{4.1e-03}$
	EPSILON. Median and IQR	1.08e-02$_{3.0e-03}$	1.39e-02$_{2.4e-03}$	6.26e-03$_{7.8e-04}$	2.46e-02$_{5.9e-03}$

Table 6.39 ZDT6 – Boxplots for HV, Spread and Epsilon – NSGAII and SPEA2

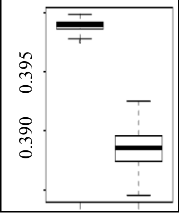
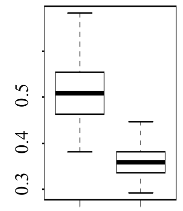
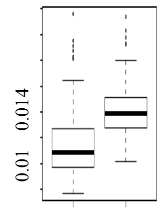
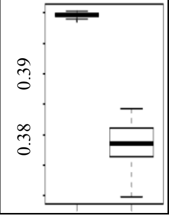
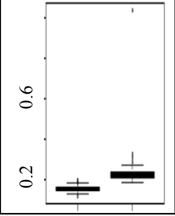
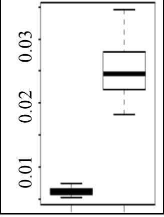
Problem	NSGAII						SPEA2					
	HV		SPREAD		EPSILON		HV		SPREAD		EPSILON	
	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM
ZDT6												

Table 6.40 Wilcoxon test in ZDT6 for HV, Spread and Epsilon

Problem			NSGAII with PLM		SPEA2 with PLM
ZDT6	HV. Mean and Std	NSGAII with PGM	▲	SPEA2 with PGM	▲
	HV. Median and IQR		▲		▲
	SPREAD. Mean and Std		▽		▲
	SPREAD. Median and IQR		▽		▲
	EPSILON. Mean and Std		▲		▲
	EPSILON. Median and IQR		▲		▲

6.9.2.2 The Deb-Theile-Laumanns- Zitzler (DTLZ) test suite

Above, we used the Zitzler-Deb-Theile (ZDT) test suite [229] in order to demonstrate the efficiency of the proposed PGM operator to solve different types of bi-objective optimization problems. Below, we will utilize the Deb-Theile-Laumanns-Zitzler (DTLZ) [60] test suite to examine the efficacy of the proposed mutation mechanism in handling problems having more than two objectives. Specifically, the following seven MOPs named DTLZ1, DTLZ2, DTLZ3, DTLZ4, DTLZ5, DTLZ6 and DTLZ7 were used for comparing the proposed PGM operator against the polynomial mutation (PLM). The DTLZ test functions are a set of scalable problems with the ability to control difficulties in converging to the Pareto front and maintain the diversity of solutions [60].

Deb-Theile-Laumanns- Zitzler's function N.1 (DTLZ1) problem:

$$Minimize = \begin{cases} f_1(x) = \frac{1}{2}\, x_1 x_2 \dots x_{M-1}\big(1 + g(X_M)\big), \\ f_2(x) = \frac{1}{2}\, x_1 x_2 \dots\ (1 - x_{M-1})\big(1 + g(X_M)\big), \\ \vdots \\ f_{M-1}(x) = \frac{1}{2}\, x_1\,(1 - x_2)\big(1 + g(X_M)\big), \\ f_M(x) = \frac{1}{2}\,(1 - x_1)\big(1 + g(X_M)\big), \end{cases}$$

$$where \quad g(X_M) = 100\left[|X_M| + \sum_{x_i \in X_M} (x_i - 0.5)^2 - \cos\left(20\pi\,(x_i - 0.5)\right)\right],$$

$$for\ 0 \le x_i \le 1\ and\ 1 \le i \le n$$

Table 6.41 DTLZ1 – Mean, std, median and iqr for HV, Spread and Epsilon – NSGAII and SPEA2

Problem		NSGAII		SPEA2	
		PGM	PLM	PGM	PLM
DTLZ1	HV. Mean and Std	7.38e-01$_{5.4e-02}$	6.04e-01$_{2.5e-01}$	7.74e-01$_{5.5e-03}$	7.33e-01$_{1.3e-01}$
	HV. Median and IQR	7.47e-01$_{1.4e-02}$	7.29e-01$_{7.8e-02}$	7.74e-01$_{7.5e-03}$	7.67e-01$_{2.0e-02}$
	SPREAD. Mean and Std	8.53e-01$_{1.1e-01}$	9.05e-01$_{1.7e-01}$	8.95e-01$_{3.1e-01}$	9.44e-01$_{3.3e-01}$
	SPREAD. Median and IQR	8.44e-01$_{9.3e-02}$	8.54e-01$_{1.1e-01}$	7.82e-01$_{5.0e-01}$	8.24e-01$_{5.6e-01}$
	EPSILON. Mean and Std	6.51e-02$_{2.2e-02}$	1.16e-01$_{8.8e-02}$	3.95e-02$_{3.3e-03}$	5.76e-02$_{4.3e-02}$
	EPSILON. Median and IQR	6.01e-02$_{1.3e-02}$	7.08e-02$_{5.4e-02}$	3.95e-02$_{4.3e-03}$	4.39e-02$_{1.2e-02}$

Table 6.42 DTLZ1 – Boxplots for HV, Spread and Epsilon – NSGAII and SPEA2

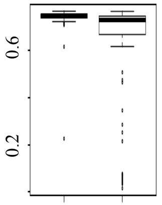
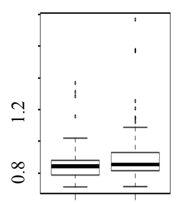
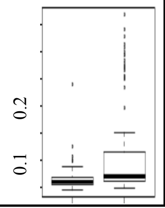
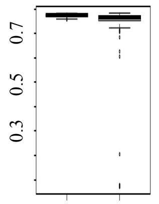
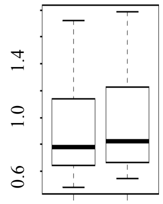
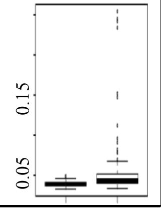
Problem	NSGAII						SPEA2					
	HV		SPREAD		EPSILON		HV		SPREAD		EPSILON	
	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM
DTLZ1												

Table 6.43 Wilcoxon test in DTLZ1 for HV, Spread and Epsilon

Problem			NSGAII with PLM		SPEA2 with PLM
DTLZ1	HV. Mean and Std	NSGAII with PGM	▲	SPEA2 with PGM	▲
	HV. Median and IQR		▲		▲
	SPREAD. Mean and Std		▲		–
	SPREAD. Median and IQR		▲		–
	EPSILON. Mean and Std		▲		▲
	EPSILON. Median and IQR		▲		▲

Deb-Theile-Laumanns- Zitzler's function N.2 (DTLZ2) problem:

$$Minimize = \begin{cases} f_1(x) = \left(1 + g(X_M)\right)\cos\left(x_1\frac{\pi}{2}\right)\cos\left(x_2\frac{\pi}{2}\right)\ldots.\cos\left(x_{M-2}\frac{\pi}{2}\right)\cos\left(x_{M-1}\frac{\pi}{2}\right), \\ f_2(x) = \left(1 + g(X_M)\right)\cos\left(x_1\frac{\pi}{2}\right)\cos\left(x_2\frac{\pi}{2}\right)\ldots.\cos\left(x_{M-2}\frac{\pi}{2}\right)sin\left(x_{M-1}\frac{\pi}{2}\right), \\ f_3(x) = \left(1 + g(X_M)\right)\cos\left(x_1\frac{\pi}{2}\right)\cos\left(x_2\frac{\pi}{2}\right)\ldots.sin\left(x_{M-2}\frac{\pi}{2}\right), \\ \vdots \\ f_{M-1}(x) = \left(1 + g(X_M)\right)\cos\left(x_1\frac{\pi}{2}\right)sin\left(x_2\frac{\pi}{2}\right), \\ f_M(x) = \left(1 + g(X_M)\right)sin\left(x_1\frac{\pi}{2}\right), \end{cases}$$

$$where \quad g(X_M) = \textstyle\sum_{x_i \in X_M}(x_i - 0.5)^2$$

$$for\ 0 \le x_i \le 1\ and\ 1 \le i \le n$$

Table 6.44 DTLZ2 – Mean, std, median and iqr for HV, Spread and Epsilon – NSGAII and SPEA2

Problem		NSGAII		SPEA2	
		PGM	PLM	PGM	PLM
DTLZ2	HV. Mean and Std	3.81e-01$_{5.4e-03}$	3.73e-01$_{6.9e-03}$	4.11e-01$_{1.1e-03}$	4.05e-01$_{2.0e-03}$
	HV. Median and IQR	3.80e-01$_{7.2e-03}$	3.74e-01$_{1.0e-02}$	4.11e-01$_{1.7e-03}$	4.05e-01$_{2.3e-03}$
	SPREAD. Mean and Std	7.11e-01$_{5.3e-02}$	6.99e-01$_{5.2e-02}$	5.28e-01$_{3.3e-02}$	5.28e-01$_{2.8e-02}$
	SPREAD. Median and IQR	7.14e-01$_{7.8e-02}$	7.03e-01$_{6.9e-02}$	5.26e-01$_{5.1e-02}$	5.31e-01$_{3.9e-02}$
	EPSILON. Mean and Std	1.29e-01$_{1.9e-02}$	1.29e-01$_{1.9e-02}$	7.94e-02$_{8.0e-03}$	8.40e-02$_{8.6e-03}$
	EPSILON. Median and IQR	1.27e-01$_{2.8e-02}$	1.27e-01$_{2.9e-02}$	7.79e-02$_{8.5e-03}$	8.26e-02$_{1.1e-02}$

Table 6.45 DTLZ2 – Boxplots for HV, Spread and Epsilon – NSGAII and SPEA2

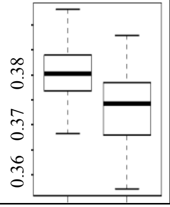
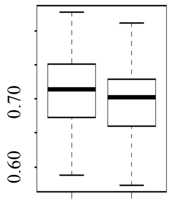
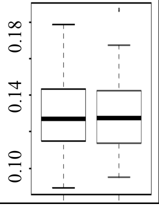
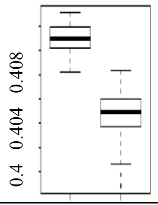
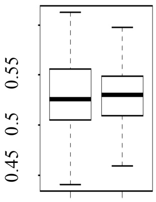
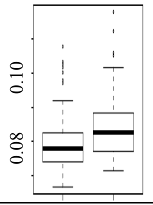
Problem	NSGAII						SPEA2					
	HV		SPREAD		EPSILON		HV		SPREAD		EPSILON	
	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM
DTLZ2												

Table 6.46 Wilcoxon test in DTLZ2 for HV, Spread and Epsilon

Problem			NSGAII with PLM		SPEA2 with PLM
DTLZ2	HV. Mean and Std	NSGAII with PGM	▲	SPEA2 with PGM	▲
	HV. Median and IQR		▲		▲
	SPREAD. Mean and Std		–		–
	SPREAD. Median and IQR		–		–
	EPSILON. Mean and Std		–		▲
	EPSILON. Median and IQR		–		▲

Deb-Theile-Laumanns- Zitzler's function N.3 (DTLZ3) problem:

$$Minimize = \begin{cases} f_1(x) = \left(1 + g(X_M)\right)\cos\left(x_1\frac{\pi}{2}\right)\cos\left(x_2\frac{\pi}{2}\right)\ldots.\cos\left(x_{M-2}\frac{\pi}{2}\right)\cos\left(x_{M-1}\frac{\pi}{2}\right), \\ f_2(x) = \left(1 + g(X_M)\right)\cos\left(x_1\frac{\pi}{2}\right)\cos\left(x_2\frac{\pi}{2}\right)\ldots.\cos\left(x_{M-2}\frac{\pi}{2}\right)sin\left(x_{M-1}\frac{\pi}{2}\right), \\ f_3(x) = \left(1 + g(X_M)\right)\cos\left(x_1\frac{\pi}{2}\right)\cos\left(x_2\frac{\pi}{2}\right)\ldots.sin\left(x_{M-2}\frac{\pi}{2}\right), \\ \vdots \\ f_{M-1}(x) = \left(1 + g(X_M)\right)\cos\left(x_1\frac{\pi}{2}\right)sin\left(x_2\frac{\pi}{2}\right), \\ f_M(x) = \left(1 + g(X_M)\right)sin\left(x_1\frac{\pi}{2}\right), \end{cases}$$

$$where \quad g(X_M) = 100\left[|X_M| + \textstyle\sum_{x_i \in X_M}(x_i - 0.5)^2 - \cos\left(20\pi\,(x_i - 0.5)\right)\right]$$

$$for\ 0 \le x_i \le 1\ and\ 1 \le i \le n$$

Table 6.47 DTLZ3 – Mean, std, median and iqr for HV, Spread and Epsilon – NSGAII and SPEA2

Problem		NSGAII		SPEA2	
		PGM	PLM	PGM	PLM
DTLZ3	HV. Mean and Std	$2.67e-03_{1.2e-02}$	$0.00e+00_{0.0e+00}$	$1.24e-03_{1.2e-02}$	$5.63e-04_{5.6e-03}$
	HV. Median and IQR	$0.00e+00_{0.0e+00}$	$0.00e+00_{0.0e+00}$	$0.00e+00_{0.0e+00}$	$0.00e+00_{0.0e+00}$
	SPREAD. Mean and Std	$1.04e+00_{1.1e-01}$	$1.03e+00_{1.3e-01}$	$1.20e+00_{1.1e-01}$	$1.25e+00_{1.5e-01}$
	SPREAD. Median and IQR	$1.02e+00_{1.5e-01}$	$1.04e+00_{1.8e-01}$	$1.19e+00_{1.6e-01}$	$1.24e+00_{1.9e-01}$
	EPSILON. Mean and Std	$2.38e+00_{1.3e+00}$	$5.00e+00_{2.4e+00}$	$3.16e+00_{1.8e+00}$	$5.94e+00_{2.4e+00}$
	EPSILON. Median and IQR	$2.24e+00_{1.4e+00}$	$4.59e+00_{2.6e+00}$	$2.87e+00_{2.1e+00}$	$5.73e+00_{3.6e+00}$

Table 6.48 DTLZ3 – Boxplots for HV, Spread and Epsilon – NSGAII and SPEA2

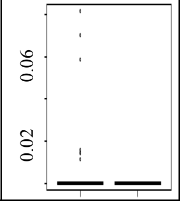
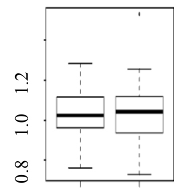
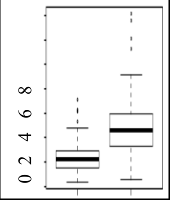
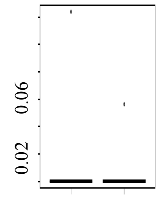
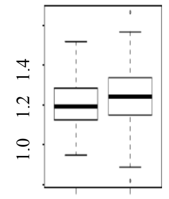
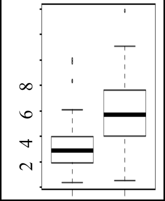
Problem	NSGAII						SPEA2					
	HV		SPREAD		EPSILON		HV		SPREAD		EPSILON	
	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM
DTLZ3												

Table 6.49 Wilcoxon test in DTLZ3 for HV, Spread and Epsilon

Problem			NSGAII with PLM		SPEA2 with PLM
DTLZ3	HV. Mean and Std	NSGAII with PGM	▲	SPEA2 with PGM	–
	HV. Median and IQR		▲		–
	SPREAD. Mean and Std		–		▲
	SPREAD. Median and IQR		–		▲
	EPSILON. Mean and Std		▲		▲
	EPSILON. Median and IQR		▲		▲

Deb-Theile-Laumanns- Zitzler's function N.4 (DTLZ4) problem:

$$Minimize = \begin{cases} f_1(x) = \left(1 + g(X_M)\right)\cos\left(x_1^a\frac{\pi}{2}\right)\cos\left(x_2^a\frac{\pi}{2}\right)\ldots.\cos\left(x_{M-2}^a\frac{\pi}{2}\right)\cos\left(x_{M-1}^a\frac{\pi}{2}\right), \\ f_2(x) = \left(1 + g(X_M)\right)\cos\left(x_1^a\frac{\pi}{2}\right)\cos\left(x_2^a\frac{\pi}{2}\right)\ldots.\cos\left(x_{M-2}^a\frac{\pi}{2}\right)sin\left(x_{M-1}^a\frac{\pi}{2}\right), \\ f_3(x) = \left(1 + g(X_M)\right)\cos\left(x_1^a\frac{\pi}{2}\right)\cos\left(x_2^a\frac{\pi}{2}\right)\ldots.sin\left(x_{M-2}^a\frac{\pi}{2}\right), \\ \vdots \\ f_{M-1}(x) = \left(1 + g(X_M)\right)\cos\left(x_1^a\frac{\pi}{2}\right)sin\left(x_2^a\frac{\pi}{2}\right), \\ f_M(x) = \left(1 + g(X_M)\right)sin\left(x_1^a\frac{\pi}{2}\right), \end{cases}$$

$$where \quad g(X_M) = \textstyle\sum_{x_i \in X_M}(x_i - 0.5)^2$$

$$for\ 0 \le x_i \le 1\ and\ 1 \le i \le n$$

Table 6.50 DTLZ4 – Mean, std, median and iqr for HV, Spread and Epsilon – NSGAII and SPEA2

Problem		NSGAII		SPEA2	
		PGM	PLM	PGM	PLM
DTLZ4	HV. Mean and Std	$3.80e\text{-}01_{4.9e\text{-}03}$	$3.74e\text{-}01_{5.2e\text{-}03}$	$3.28e\text{-}01_{1.1e\text{-}01}$	$3.17e\text{-}01_{1.2e\text{-}01}$
	HV. Median and IQR	$3.81e\text{-}01_{6.9e\text{-}03}$	$3.75e\text{-}01_{7.0e\text{-}03}$	$4.05e\text{-}01_{2.0e\text{-}01}$	$3.98e\text{-}01_{1.9e\text{-}01}$
	SPREAD. Mean and Std	$6.83e\text{-}01_{4.8e\text{-}02}$	$6.72e\text{-}01_{4.6e\text{-}02}$	$4.69e\text{-}01_{1.5e\text{-}01}$	$5.21e\text{-}01_{1.7e\text{-}01}$
	SPREAD. Median and IQR	$6.82e\text{-}01_{7.4e\text{-}02}$	$6.72e\text{-}01_{7.5e\text{-}02}$	$5.23e\text{-}01_{5.5e\text{-}02}$	$5.27e\text{-}01_{5.2e\text{-}02}$
	EPSILON. Mean and Std	$1.10e\text{-}01_{1.7e\text{-}02}$	$1.11e\text{-}01_{1.8e\text{-}02}$	$2.86e\text{-}01_{2.8e\text{-}01}$	$2.94e\text{-}01_{3.1e\text{-}01}$
	EPSILON. Median and IQR	$1.09e\text{-}01_{2.2e\text{-}02}$	$1.09e\text{-}01_{2.5e\text{-}02}$	$7.76e\text{-}02_{5.6e\text{-}01}$	$7.97e\text{-}02_{5.6e\text{-}01}$

Table 6.51 DTLZ4 – Boxplots for HV, Spread and Epsilon – NSGAII and SPEA2

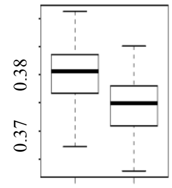
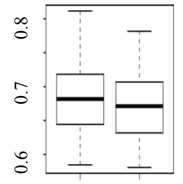
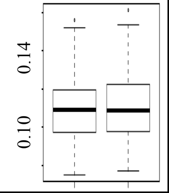
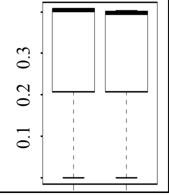
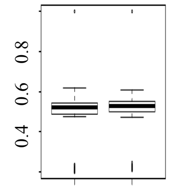
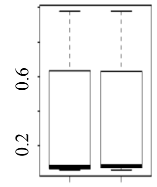
Problem	NSGAII						SPEA2					
	HV		SPREAD		EPSILON		HV		SPREAD		EPSILON	
	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM
DTLZ4												

Table 6.52 Wilcoxon test in DTLZ4 for HV, Spread and Epsilon

Problem			NSGAII with PLM		SPEA2 with PLM
DTLZ4	HV. Mean and Std	NSGAII with PGM	▲	SPEA2 with PGM	▲
	HV. Median and IQR		▲		▲
	SPREAD. Mean and Std		–		–
	SPREAD. Median and IQR		–		–
	EPSILON. Mean and Std		–		–
	EPSILON. Median and IQR		–		–

Deb-Theile-Laumanns- Zitzler's function N.5 (DTLZ5) problem:

$$Minimize = \begin{cases} f_1(x) = \left(1 + g(X_M)\right)\cos\left(\theta_1\frac{\pi}{2}\right)\cos\left(\theta_2\frac{\pi}{2}\right)\ldots.\cos\left(\theta_{M-2}\frac{\pi}{2}\right)\cos\left(\theta_{M-1}\frac{\pi}{2}\right), \\ f_2(x) = \left(1 + g(X_M)\right)\cos\left(\theta_1\frac{\pi}{2}\right)\cos\left(\theta_2\frac{\pi}{2}\right)\ldots.\cos\left(\theta_{M-2}\frac{\pi}{2}\right)sin\left(\theta_{M-1}\frac{\pi}{2}\right), \\ f_3(x) = \left(1 + g(X_M)\right)\cos\left(\theta_1\frac{\pi}{2}\right)\cos\left(\theta_2\frac{\pi}{2}\right)\ldots.sin\left(\theta_{M-2}\frac{\pi}{2}\right), \\ \vdots \\ f_{M-1}(x) = \left(1 + g(X_M)\right)\cos\left(\theta_1\frac{\pi}{2}\right)sin\left(\theta_2\frac{\pi}{2}\right), \\ f_M(x) = \left(1 + g(X_M)\right)sin\left(\theta_1\frac{\pi}{2}\right), \end{cases}$$

$$where \quad \theta_i = \frac{\pi}{4(1+g(X_M))}(1 + 2g(X_M)x_i) \qquad for\ i = 2, 3, \ldots, (M-1),$$

$$g(X_M) = \textstyle\sum_{x_i \in X_M}(x_i - 0.5)^2 \quad for\ 0 \le x_i \le 1\ and\ 1 \le i \le n$$

Table 6.53 DTLZ5 – Mean, std, median and iqr for HV, Spread and Epsilon – NSGAII and SPEA2

Problem		NSGAII		SPEA2	
		PGM	PLM	PGM	PLM
DTLZ5	HV. Mean and Std	9.29e-02$_{1.7e\text{-}04}$	9.28e-02$_{2.1e\text{-}04}$	9.33e-02$_{1.2e\text{-}04}$	9.32e-02$_{1.4e\text{-}04}$
	HV. Median and IQR	9.29e-02$_{1.9e\text{-}04}$	9.28e-02$_{3.0e\text{-}04}$	9.33e-02$_{1.5e\text{-}04}$	9.32e-02$_{2.0e\text{-}04}$
	SPREAD. Mean and Std	4.42e-01$_{3.9e\text{-}02}$	4.42e-01$_{4.7e\text{-}02}$	2.07e-01$_{3.0e\text{-}02}$	2.34e-01$_{3.9e\text{-}02}$
	SPREAD. Median and IQR	4.37e-01$_{5.3e\text{-}02}$	4.32e-01$_{5.9e\text{-}02}$	2.03e-01$_{3.3e\text{-}02}$	2.24e-01$_{4.5e\text{-}02}$
	EPSILON. Mean and Std	1.06e-02$_{1.6e\text{-}03}$	1.10e-02$_{1.9e\text{-}03}$	7.16e-03$_{9.5e\text{-}04}$	7.69e-03$_{9.3e\text{-}04}$
	EPSILON. Median and IQR	1.04e-02$_{2.0e\text{-}03}$	1.08e-02$_{2.5e\text{-}03}$	7.03e-03$_{1.3e\text{-}03}$	7.62e-03$_{1.3e\text{-}03}$

Table 6.54 DTLZ5 – Boxplots for HV, Spread and Epsilon – NSGAII and SPEA2

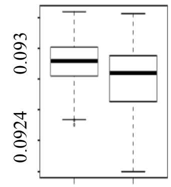
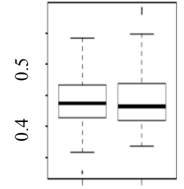
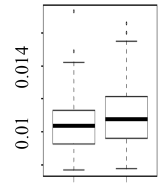
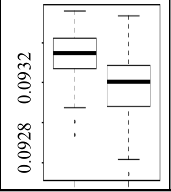
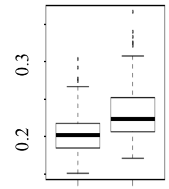
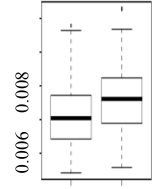
Problem	NSGAII						SPEA2					
	HV		SPREAD		EPSILON		HV		SPREAD		EPSILON	
	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM
DTLZ5												

Table 6.55 Wilcoxon test in DTLZ5 for HV, Spread and Epsilon

Problem			NSGAII with PLM		SPEA2 with PLM
DTLZ5	HV. Mean and Std	NSGAII with PGM	▲	SPEA2 with PGM	▲
	HV. Median and IQR		▲		▲
	SPREAD. Mean and Std		–		▲
	SPREAD. Median and IQR		–		▲
	EPSILON. Mean and Std		–		▲
	EPSILON. Median and IQR		–		▲

Deb-Theile-Laumanns- Zitzler's function N.6 (DTLZ6) problem:

$$Minimize = \begin{cases} f_1(x) = \left(1 + g(X_M)\right) \cos\left(\theta_1 \frac{\pi}{2}\right) \cos\left(\theta_2 \frac{\pi}{2}\right) \ldots . \cos\left(\theta_{M-2} \frac{\pi}{2}\right) \cos\left(\theta_{M-1} \frac{\pi}{2}\right), \\ f_2(x) = \left(1 + g(X_M)\right) \cos\left(\theta_1 \frac{\pi}{2}\right) \cos\left(\theta_2 \frac{\pi}{2}\right) \ldots . \cos\left(\theta_{M-2} \frac{\pi}{2}\right) sin\left(\theta_{M-1} \frac{\pi}{2}\right), \\ f_3(x) = \left(1 + g(X_M)\right) \cos\left(\theta_1 \frac{\pi}{2}\right) \cos\left(\theta_2 \frac{\pi}{2}\right) \ldots . sin\left(\theta_{M-2} \frac{\pi}{2}\right), \\ \vdots \\ f_{M-1}(x) = \left(1 + g(X_M)\right) \cos\left(\theta_1 \frac{\pi}{2}\right) sin\left(\theta_2 \frac{\pi}{2}\right), \\ f_M(x) = \left(1 + g(X_M)\right) sin\left(\theta_1 \frac{\pi}{2}\right), \end{cases}$$

$$where \quad \theta_i = \frac{\pi}{4(1+g(X_M))}(1 + 2g(X_M)x_i) \qquad for\ i = 2, 3, \ldots, (M-1),$$

$$g(X_M) = \textstyle\sum_{x_i \in X_M} x_i^{0.1} \quad for\ 0 \le x_i \le 1\ and\ 1 \le i \le n$$

Table 6.56 DTLZ6 – Mean, std, median and iqr for HV, Spread and Epsilon – NSGAII and SPEA2

Problem		NSGAII		SPEA2	
		PGM	PLM	PGM	PLM
DTLZ6	HV. Mean and Std	8.73e-02$_{9.8e-03}$	0.00e+00$_{0.0e+00}$	4.32e-02$_{1.9e-02}$	0.00e+00$_{0.0e+00}$
	HV. Median and IQR	9.36e-02$_{1.6e-02}$	0.00e+00$_{0.0e+00}$	3.86e-02$_{2.7e-02}$	0.00e+00$_{0.0e+00}$
	SPREAD. Mean and Std	6.21e-01$_{9.1e-02}$	8.16e-01$_{4.9e-02}$	6.71e-01$_{9.5e-02}$	5.79e-01$_{3.8e-02}$
	SPREAD. Median and IQR	5.97e-01$_{1.3e-01}$	8.15e-01$_{5.6e-02}$	6.54e-01$_{1.2e-01}$	5.78e-01$_{4.6e-02}$
	EPSILON. Mean and Std	1.86e-02$_{1.2e-02}$	8.73e-01$_{8.1e-02}$	1.11e-01$_{4.6e-02}$	7.82e-01$_{6.0e-02}$
	EPSILON. Median and IQR	1.27e-02$_{1.7e-02}$	8.59e-01$_{1.1e-01}$	1.11e-01$_{6.5e-02}$	7.90e-01$_{7.4e-02}$

Table 6.57 DTLZ6 – Boxplots for HV, Spread and Epsilon – NSGAII and SPEA2

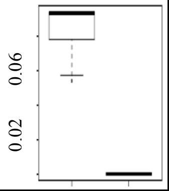
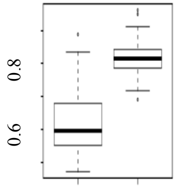
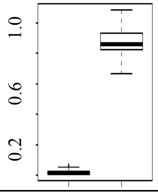
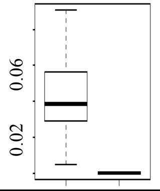
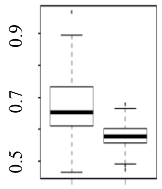
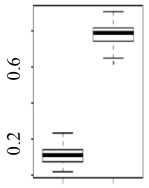
Problem	NSGAII			SPEA2		
	HV PGM | PLM	SPREAD PGM | PLM	EPSILON PGM | PLM	HV PGM | PLM	SPREAD PGM | PLM	EPSILON PGM | PLM
DTLZ6						

Table 6.58 Wilcoxon test in DTLZ6 for HV, Spread and Epsilon

Problem			NSGAII with PLM		SPEA2 with PLM
DTLZ6	HV. Mean and Std	NSGAII with PGM	▲	SPEA2 with PGM	▲
	HV. Median and IQR		▲		▲
	SPREAD. Mean and Std		▲		▽
	SPREAD. Median and IQR		▲		▽
	EPSILON. Mean and Std		▲		▲
	EPSILON. Median and IQR		▲		▲

Deb-Theile-Laumanns- Zitzler's function N.7 (DTLZ7) problem:

$$Minimize = \begin{cases} f_1(X_1) = x_1, \\ f_2(X_2) = x_2, \\ \quad\vdots \\ f_{M-1}(X_{M-1}) = x_{M-1}, \\ f_M(X) = \big(1 + g(X_M)\big)\, h(f_1, f_2, \dots, f_{M-1}, g), \end{cases}$$

$$where \quad g(X_M) = 1 + \frac{9}{|X_M|} \sum_{x_i \in X_M} x_i$$

$$h(f_1, f_2, \dots, f_{M-1}, g) = M - \sum_{i=1}^{M-1} \left[\frac{f_i}{1+g}\,(1 + \sin(3\pi f_i))\right]$$

$$for\ 0 \le x_i \le 1\ and\ 1 \le i \le n$$

Table 6.59 DTLZ7 – Mean, std, median and iqr for HV, Spread and Epsilon – NSGAII and SPEA2

Problem		NSGAII		SPEA2	
		PGM	PLM	PGM	PLM
DTLZ7	HV. Mean and Std	2.80e-01$_{4.4e-03}$	2.80e-01$_{3.4e-03}$	2.91e-01$_{4.0e-03}$	2.89e-01$_{6.9e-03}$
	HV. Median and IQR	2.80e-01$_{6.4e-03}$	2.81e-01$_{3.9e-03}$	2.91e-01$_{3.7e-03}$	2.90e-01$_{3.6e-03}$
	SPREAD. Mean and Std	7.37e-01$_{5.0e-02}$	7.50e-01$_{4.9e-02}$	5.98e-01$_{3.8e-02}$	6.09e-01$_{4.4e-02}$
	SPREAD. Median and IQR	7.37e-01$_{5.8e-02}$	7.44e-01$_{7.7e-02}$	5.96e-01$_{4.3e-02}$	6.10e-01$_{5.2e-02}$
	EPSILON. Mean and Std	1.43e-01$_{4.5e-02}$	1.39e-01$_{4.1e-02}$	1.11e-01$_{1.2e-01}$	1.47e-01$_{2.3e-01}$
	EPSILON. Median and IQR	1.34e-01$_{3.2e-02}$	1.30e-01$_{3.7e-02}$	9.84e-02$_{1.2e-02}$	9.88e-02$_{1.1e-02}$

Table 6.60 DTLZ7 – Boxplots for HV, Spread and Epsilon – NSGAII and SPEA2

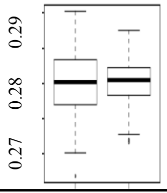
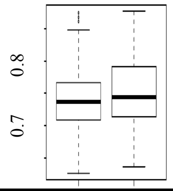
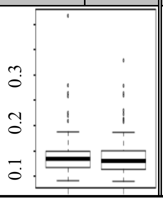
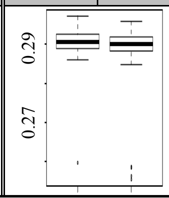
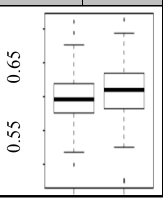
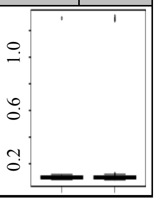
Problem	NSGAII						SPEA2					
	HV		SPREAD		EPSILON		HV		SPREAD		EPSILON	
	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM	PGM	PLM
DTLZ7												

Table 6.61 Wilcoxon test in DTLZ7 for HV, Spread and Epsilon

Problem			NSGAII with PLM		SPEA2 with PLM
DTLZ7	HV. Mean and Std	NSGAII with PGM	–	SPEA2 with PGM	–
	HV. Median and IQR		–		–
	SPREAD. Mean and Std		–		▲
	SPREAD. Median and IQR		–		▲
	EPSILON. Mean and Std		–		–
	EPSILON. Median and IQR		–		–

6.9.2.3 Analysis of the Results

In this section, we analyze the results obtained by applying the Probe Guided Mutation (PGM) operator and the Polynomial Mutation (PLM) operator respectively to the NSGAII and SPEA2 for solving two sets of benchmark problems namely ZDT1-4, 6 and DTLZ 1-7. The assessment of the performance of the proposed mutation operator is done with the assistance of three well known performance indicators, namely *Hypervolume*, *Spread* and *Epsilon* indicator.

Examining the results of the first indicator, the *HV*, we notice that the PGM operator performs better under both different configurations i.e. NSGAII and SPEA2 for the majority of the test functions examined, compared with results derived by applying the classical PLM operator. Please notice that according to a number of studies [82], [117], [138], [176], [228], [234] HV has nicer mathematical properties than many other metrics and is able to capture in a single number both the closeness of the solutions to the optimal set and, to some extent, the spread of the solutions across objective space. Figure 6.19.

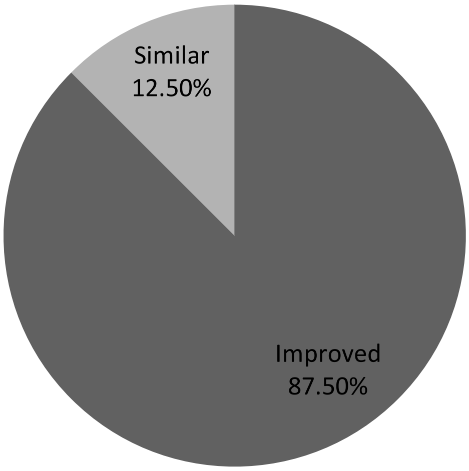


Figure 6.19 Hypervolume: PGM performance is compared with PLM

reveals the relevant figures for the performance of PGM regarding the *HV* indicator. The Wilcoxon rank-sum test (see Tables 6.27, 6.30, 6.33, 6.36, 6.39, 6.42, 6.45, 6.48, 6.51, 6.54, 6.57 and 6.60) validates that the observed difference in PGM and PLM performance is statistically significant with 95% confidence. Analytically, in 87.50% of the cases the PGM yields better results with confidence than the conventional configuration of NSGAII and SPEA2 with the PLM operator. Also, in 12.50% of the cases there was not statistical significance between the PGM and PLM. We should highlight that in none of the examined test functions PLM outperformed the PGM operator regarding the *HV metric*. Regarding the *Spread indicator* (see Fig. 6.20) the same methodology applied.

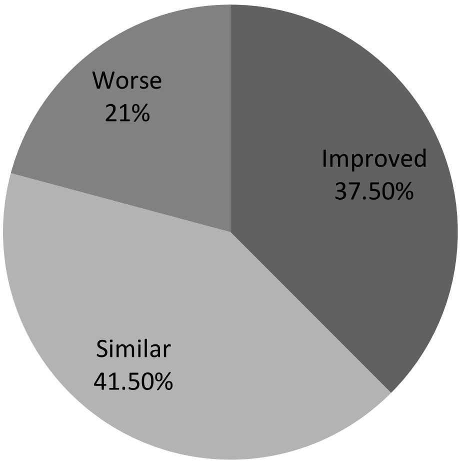


Figure 6.20 Spread: PGM performance is compared with PLM

In particular the Wilcoxon rank-sum test helped us to identify that in 37.50% of the cases the PGM yields better results with confidence than the conventional configuration of NSGAII and SPEA2 with the PLM operator. In 41.50% of the cases there was not statistical significance between the PGM and PLM regarding the *Spread metric* and in 21% of the examined cases the classical PLM operator generated better results with confidence than the proposed methodology. The relevant results regarding the Spread indicator, suggest that in certain cases the exploration of the most promising regions of the search space can occur at the expense of the spread of solutions. Figure 6.21 reveals the relevant numbers for the performance of PGM regarding the *Epsilon* indicator. In particular the Wilcoxon rank-sum test helped us to identify that in 66.67% of the cases

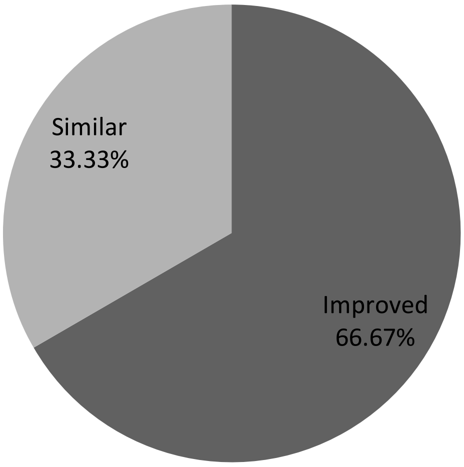


Figure 6.21 Epsilon: PGM performance is compared with PLM

the PGM yields better results with confidence than the conventional configuration of NSGAII and SPEA2 with the PLM operator. Also, in 33.33% of the cases there was not statistical significance between the PGM and PLM regarding the *Epsilon metric.* We should highlight that the PLM in none of the examined test functions outperformed the PGM operator regarding the *Epsilon metric*. Please note that the boxplots (Tables 6.27, 6.30, 6.33, 6.36, 6.39, 6.42, 6.45, 6.48, 6.51, 6.54, 6.57 and 6.60) provide graphical representation of the relevant results for the three metrics examined, namely *HV*, *Spread* and *Epsilon*. Fig. 6.22 presents the results regarding the application of PGM to NSGAII, for all performance indicators collectively (HV, Spread, Epsilon).

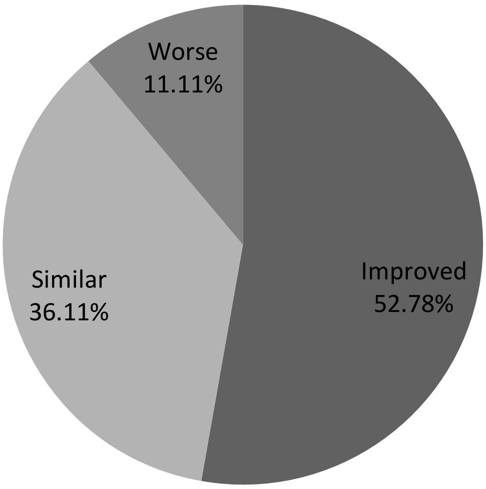


Figure 6.22 NSGAII: PGM performance is compared with PLM

Thus, for the 52.78% of the examined cases the NSGAII with the PGM operator yields better results with confidence than the conventional configuration of NSGAII with the PLM operator. Also, in 36.11% of the cases there was not statistical significance in the results from the application of PGM to the NSGAII. Finally, in 11.11% of the cases the NSGAII with its typical configuration generated better results with confidence than the proposed methodology. Figure 6.23 presents the results regarding the application of PGM to SPEA2, for all three performance indicators collectively (HV, Spread, Epsilon). Thus, for the 75% of the examined cases the SPEA2 with the PGM operator yields better results with confidence than the conventional configuration of SPEA2 with the PLM operator. Also, in 22.22% of the cases there was not statistical significance in the results from the application of PGM to the SPEA2.

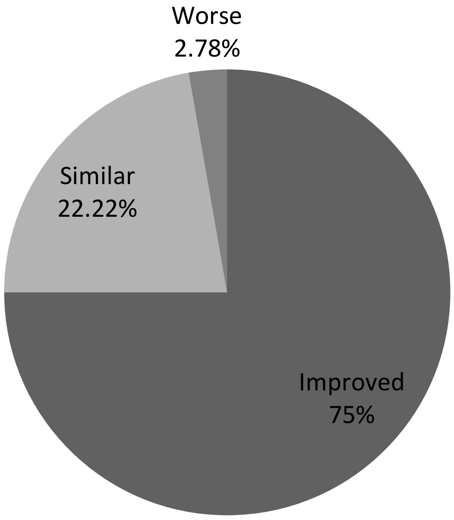


Figure 6.23 SPEA2: PGM performance is compared with PLM

Finally, in 2.78% of the cases the SPEA2 with its typical configuration generated better results with confidence than the proposed methodology. Taking into consideration that the implementation of PGM to SPEA2 generates better results with confidence for the 75% of the examined cases, as shown in Fig. 6.23, it seems reasonable to accept an average computational overhead of 13.94% that involves the proposed methodology as shown in Table 6.25.

6.10 Conclusions

This chapter presents a new Probe Guided Mutation (PGM) operator and its application for solving the cardinality constrained portfolio optimization problem (CCPOP) and two well-known families of test functions (i.e. ZDT1-4, 6 and DTLZ1-7) alike. The proposed mutation operator incorporates a fitness functions evaluation mechanism that allows the evaluation of the corresponding fitness for the left hand side region and the right hand side region of the parent solution. The selection between the two alternative child solutions is done with the assistance of the Pareto optimality conditions. Thanks to the PGM operator the algorithm is able to move fast towards the higher fitness regions of the search landscape and discover near optimal solutions.

The evaluation of the proposed mutation operator is done with the assistance of six portfolio optimization problems (port1-6) that correspond to six different capital markets. The size of these problems varies from 31 stocks for the smallest and up to 457 stocks for the largest problem. The performance of the proposed PGM operator is assessed in comparison with the classical Polynomial Mutation operator (PLM) with the assistance of two well-known MOEAs, namely the Non-dominated Sorting Genetic Algorithm II (NSGAII) and Strength Pareto Evolutionary Algorithm 2 (SPEA2) for the solution of the CCPOP. The evaluation of the performance is based on a variety of metrics that assess both the proximity of the solutions to the Pareto front and their dispersion on it.

The relevant results indicate that the proposed mutation operator outperforms with confidence the PLM operator for all performance metrics, when applied to the NSGAII and SPEA2. We also wanted to find out how well the PGM operator performs when compared to the True Pareto Efficient frontier of the CCPOP. For that purpose, we formulated the CCPOP as a Mixed Integer Quadratic Program (MIQP) and we calculated a number of exact points for each one of the port1-5 portfolio optimization problems with the assistance of CPLEX version 12.5. We found out that the approximate efficient frontiers that are generated by the proposed PGM operator demonstrate a remarkable concentration of the search effort very close or in some cases on top of the true efficient frontier (CPLEX). On the contrary, the approximate efficient frontiers that are generated by the classical PLM produce a considerable number of points far away from the true efficient frontier and on top of that they demonstrate discontinuities especially for the higher levels of risk and return combinations.

We also carried out a final test in order to demonstrate the exploratory capabilities of the proposed methodology. In particular, we revisited the port2 test problem, but this time we used only *40,000* evaluations for the PGM and *200,000* evaluations for the PLM to compensate for the extra fitness functions evaluations introduced by the PGM. We found out that the PGM operator still outperforms with confidence the PLM operator for all performance metrics, when applied to the NSGAII and SPEA2, although we only used a fraction of the evaluations that are being used for the PLM.

Finally, we tested the efficiency of PGM operator with the assistance of the ZDT1-4, 6 and DTLZ1-7 families of test functions. Compared with the conventional PLM operator the proposed methodology (PGM) generates better results with confidence for the 87.50% of the test functions examined regarding the *HV indicator*. It also produces better results for the 66.67% of the test functions examined regarding the *Epsilon indicator* and not worse results than the classical PLM operator for the rest of the cases. Regarding the *Spread metric*, the proposed methodology generates better results with confidence for the 37.50% of the test functions examined and worse results for the 21% of the examined problems.

When we examine the effect by the implementation of PGM to NSGAII and SPEA2 individually, we notice that the proposed methodology proved efficient regarding the SPEA2 as for a computational overhead of 13.94% generates better results with confidence for the 75% of the examined cases. Finally, very efficient appears to be the implementation of the proposed mutation operator to the NSGAII, as for a computational overhead of 3.62% compared with the typical configuration of the NSGAII with the polynomial mutation operator, generates better results with confidence for the 52.78% of the examined cases.

We are interested in comparing the PGM with other mutation techniques. In our future work, we will attempt to develop a technique that will update the mutation probability (P_m) at run-time according to the performance of the algorithm in a number of metrics such as the hypervolume or epsilon indicator.

Chapter 7

An Experimental Analysis of a new Two Stage Crossover operator for Multiobjective Optimization

7.1 Introduction

In recent years multiobjective problems (MOPs) have received growing attention. The increasing attention can be attributed to the development of intelligent techniques that are capable of handling such difficult problems. In particular, Evolutionary Algorithms for MOPs have been proved very efficient in finding near optimal solutions in hard, real world optimization problems with multiple objectives and constraints. The Multiobjective Evolutionary Algorithms (MOEAs) are population based and they work by evolving a population of candidate solutions to satisfy the objectives and constraints of a particular problem [140], [155]. The recombination operator plays a very important role in MOEAs as exploitation force. One of the most popular real coded crossover operators that has been applied to a considerable number of MOEAs, is the Simulated Binary Crossover (SBX) [54]. In this chapter, we revisit the classical SBX and we propose a new strategy that demonstrates enhanced exploratory properties.

Over the past years a number of studies proposed different recombination operators that each one of them has its own strength and weaknesses. Tsutsui, Yamamura and Higuchi (1999) [208] proposed a Simplex Crossover (SPX) operator for real coded GAs. They found that the SPX works well on functions having multimodality. However, the authors did not test the SPX operator to multiobjective problems. Da Ronco and Benini (2013) [48] proposed the Genetic Diversity Evolutionary Algorithm II (GeDEA-II) which features a modified Simplex-Crossover operator (SPX). Unlike the SPX presented in [208], the SPX exploited in GeDEA-II requires only two parents to form a new child. The performance of the GeDEA-II was tested against other different state-of-the-art MOEAs

A novel hybrid harmony search HS [98] with arithmetic crossover operator (AC) [3] called ACHS is proposed by Niu et al. [167]. The proposed ACHS enhances the performance and applicability of the conventional HS. In HS, the global best information, which can improve the quality of the new harmony and speed up the convergence rate, is not fully utilized. According to the Niu et al. [167] the ACHS overcome this shortcome of the HS by using the global best information to update the new generated harmony through crossover operation.

Yoon et al. [220] proposed a new crossover operator for real-coded genetic algorithms employing a novel methodology to remove the inherent bias of pre-existing crossover operators. According to the authors this is done by transforming the topology of the hyper-rectangular real space by gluing opposite boundaries and designing a boundary extension method for making the fitness function smooth at the glued boundary.

Qi et al [168] proposed an immune multi-objective optimization algorithm with differential evolution inspired recombination (IMADE). In the proposed IMADE, the novel crossover provides two types of candidate searching directions by taking three parents which distribute along the current Pareto set (PS) within a local area. According to the authors [168] the new recombination operator utilizes the regularity of continuous MOPs and the distributions of current population, which helps IMADE maintain a more uniformly distributed Pareto front (PF) and converge much faster.

Hervas-Martinez et al. [112] proposed a confidence interval based crossover for real-coded GAs. This crossover is based on defining a confidence interval for a localization estimator. According to the authors the operator will converge to fittest individuals of the population if the parameters are correctly chosen. As the authors report the major disadvantage of the operator is the assumption of the normality and independence of the genes.

Amit Banerjee [14] proposed a new context-sensitive crossover operator for clustering that uses domain specific information about cluster quality to drive the crossover process. According to the author this crossover operator ensures that high quality genetic information is inherited by the child. This operator also achieves quick convergence to a local optimal solution however at the cost of diversity of the population and the higher computational cost.

Deb et al. (2002) [55] proposed a parent-centric recombination (PCX) operator that favours solutions close to parents. The authors found that the PCX operator is a meaningful and efficient way of solving real-parameter optimization problems.

Ono and Kobayashi (1997) [170] proposed the unimodal normal distribution crossover (UNDX) operator. The UNDX operator uses multiple parents and creates offspring solutions with the assistance of a normal probability distribution around the center of mass of these parents.

In another study, Eshelman and Schaffer (1993) [81] proposed a blend crossover (BLX-α) operator for real coded GAs. The authors introduced the notion of interval schemata. Specifically, for two parent solutions $x_i^{(1,\ t)}$ and $x_i^{(2,\ t)}$, assuming $x_i^{(1,\ t)} < x_i^{(2,\ t)}$ the BLX-α randomly picks a solution in the range [$x_i^{(1,\ t)}$ - α($x_i^{(2,\ t)}$ - $x_i^{(1,\ t)}$), $x_i^{(2,\ t)}$ + α($x_i^{(2,\ t)}$ - $x_i^{(1,\ t)}$)]. The BLX-α for α = 0.5 obtains a balanced relationship between the convergence (exploitation) and divergence (exploration), since the probability that an offspring shall lie outside its parents becomes equal to the probability that it shall lie between its parents.

The BLX-α operator belongs to the family of the neighborhood based real-parameter crossover operators that use probability distributions for creating the genes of the offspring in restricted search spaces around the regions marked by the genes of the parents.

Deep and Thakur [62] introduced a crossover operator that uses the Laplace distribution. They named the new operator Laplace crossover (LX) and belongs to the family of parent-centric crossover operators. The LX operator generates two offspring symmetrically placed with respect to the position of the parents. Depending on the value of a parameter *b,* the offspring can be produced near or far from the parents.

Relatively recently, Differential Evolution (DE) [175] has emerged as one of the most popular reproduction operators for real coded evolutionary multiobjective optimization. DE demonstrates a special ability to deal with non-differentiable and multimodal optimization problems. In DE is maintained a population of candidate solutions. These solutions are updated according to their fitness value. In DE new individuals are generated with the assistance of differential mutation (or perturbation) and crossover.

Another popular evolutionary algorithm technique is the Particle swarm optimization (PSO) [122]. In PSO each candidate solution is associated with a velocity. The candidate solutions are called particles and the position of each particle is changed according to its own experience and that of its neighbors. Through this process the particles move towards better solutions of the search space.

Estimation of Distribution Algorithm (EDA) [23][133] is an optimization algorithm that extracts statistical information from the population of solutions. In EDA the algorithm starts by generating a population of solutions. A set of solutions is selected from the population according to their fitness value. The algorithm then constructs a probabilistic model which attempts to estimate the probability distribution of the selected solutions. Once the model is constructed, new solutions are generated by sampling the distribution encoded by this model. The new solutions are then incorporated back into the old population by replacing some of the old solutions.

Although the sizeable amount of work on the recombination operators, one particular recombination operation appears to dominate the relevant field. Indeed, the SBX is the recombination operator of choice by many popular contemporary MOEAs as indicated by numerous studies in the field. In this chapter we revisit the SBX operator and we propose a Two Stage Crossover (TSX) operator for more efficient exploration of the search space.

The rest of the chapter is structure as follows. In section 2, a description of the Simulated Binary Crossover (SBX) is given and in section 3 the proposed Two Stage Crossover (TSX) operator is presented. The experimental environment is presented in section 4. Section 5 presents the performance metrics. In section 6 we test the performance of the proposed TSX by using the Deb, Thiele, Laumanns and Zitzler (DTLZ) set of test functions and we analyze the relevant results. Finally, section 7 concludes the chapter.

7.2 Simulated Binary Crossover (SBX)

The simulated binary crossover (SBX) operator was introduced by Deb and Agrawal [54] in 1995. It uses a probability distribution around two parents to create two child solutions. In essence the use of a probability distribution by the SBX simulates the working principle of creating child solutions in crossover operators used in binary-coded algorithms. In SBX as introduced by [54] each decision variable x_i, can take values in the interval: $x_i^{(L)} \leq x_i \leq x_i^{(U)}, i = 1, 2, \dots, n$. Where $x_i^{(L)}$ and $x_i^{(U)}$ stand respectively for the lower and upper bounds for the decision variable i. In

SBX, two parent solutions $y^{(1)}$ and $y^{(2)}$ generate two child solutions $c^{(1)}$ and $c^{(2)}$ as follows:

1. Calculate the spread factor β:

$$\beta = 1 + \frac{2}{y^{(2)} - y^{(1)}}\, min\left[\left(y^{(1)} - y^{(l)}\right), \left(y^{(u)} - y^{(2)}\right)\right]$$

2. Calculate parameter a :

$$\alpha = 2 - \beta^{-(\eta_c+1)}$$

3. Create a random number u between 0 and 1.

   $u \longrightarrow [0, 1];$

4. Find a parameter β_q with the assistance of the following polynomial probability distribution:

$$\beta_q = \begin{cases} (au)^{1/(\eta_c+1)} & if\ u \leq \frac{1}{a}, \\ \left(\frac{1}{2 - au}\right)^{1/(\eta_c+1)} & otherwise \end{cases}$$

   The above procedure allows a zero probability of creating any child solutions outside the prescribed range $\left[x^{(L)}, x^{(U)}\right]$. Where η_c is the distribution index for SBX and can take any nonnegative value. In particular, small values of η_c allow child solutions to be created far away from parents and large values of η_c allow child solutions to be created near the parent solutions.

5. The child solutions are then calculated as follows:

$$c^{(1)} = 0.5\left[\left(y^{(1)} + y^{(2)}\right) - \beta_q |y^{(2)} - \ y^{(1)}|\right]$$

$$c^{(2)} = 0.5\left[\left(y^{(1)} + y^{(2)}\right) + \beta_q |y^{(2)} - \ y^{(1)}|\right]$$

The probability distributions as shown in *step 4*, do not create any solution outside the given bounds $\left[x^{(L)}, x^{(U)}\right]$ instead they scale up the probability of solutions inside the bounds, as shown by the solid line in Fig. 7.1.

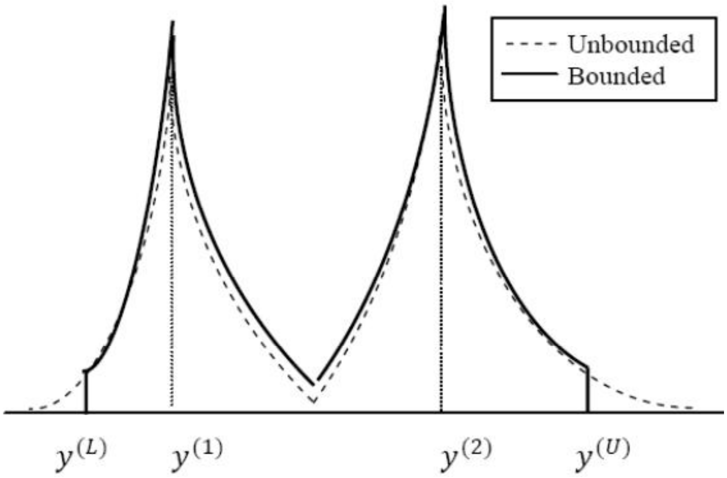


Fig. 7.1. Probabilities distributions for bounded and unbounded cases of SBX operator

SBX is one of the most popular recombination operators and has been utilized by a considerable number of studies in the field. Specifically, SBX has been tested and found to work well in many problems that have continuous search space when compared to other real-coded recombination operators. Also the SBX operator can restrict child solutions to any arbitrary closeness to the parent solutions and thus does not require any separate mating restriction scheme [80]. SBX has been proved to work well with problems where the bounds of the optimum point are not known and where there are multiple optima [54]. Finally, SBX has been successfully applied to solving many constrained optimization problems [16], [192].

The purpose of this study is to use all the good properties of the SBX operator and on top of that to reinforce them by incorporating an efficient exploratory mechanism that increases considerably the performance of the SBX operator.

7.3 Two Stage Crossover (TSX) operator

In this section, we propose a new Two Stage Crossover (TSX) operator for more efficient exploration of the search space. The new operator uses all good properties of the well-known simulated binary crossover (SBX) operator and on top of that we reinforce them by incorporating an efficient exploratory mechanism that increases considerably the performance of the MOEAs. The TSX algorithm starts by calculating the spread factor β and then the parameter α in the same manner as the SBX. However, in *step 3* we follow a different strategy. In particular, a random r_b boolean value of true or false is generated with equal probability. If r_b is *false* the SBX operator performed, otherwise if r_b is *true* we proceed to *step 4.* We have already explained the SBX operator's basic functionally in section 2, thus below we are focusing our attention in the case that the r_b take a *true* value.

As shown in *step 4* two random numbers are generated. A random number $\mathrm{u}_L \in [0,\ 1/a]$ to sample the left hand side and a random number $\mathrm{u}_R \in (1/a, 1]$ to sample the right hand side of the probability distribution. Please, notice that the SBX, at this particular point as shown in section 2, follows a different methodology. In particular in SBX operator, a random number $u \in [0, 1]$ is generated. If $u \leq 1/a,$ it samples to the left hand side (region between $y^{(L)}$ and $y^{(i)}$), otherwise if $u > 1/a$ it samples to the right hand side (region between $y^{(i)}$ and $y^{(U)}$), where $y^{(i)}$is the i^{th} parent solution.

As shown in *step 5,* emerge two values of β_q . The β_q^L that samples the left hand side of the polynomial probability distribution and the β_q^R that samples the right hand side of the polynomial probability distribution. Next, as shown below in *step 6* with the assistance of β_q^L and β_q^R are formulated two variants for each child solution. Specifically, $\mathrm{c}_L^{(1)}$ and $\mathrm{c}_R^{(1)}$ are the two variants that emerge by substituting the β_q^L and β_q^R to $\mathrm{c}^{(1)}$. Respectively $\mathrm{c}_L^{(2)}$ and $\mathrm{c}_R^{(2)}$ are the two variants that emerge by substituting the β_q^L and β_q^R to $\mathrm{c}^{(2)}$.

Then, by substituting to the parent solution vector at the position of the selected variable to be crossovered, respectively the $\mathrm{c}_L^{(1)}$and $\mathrm{c}_R^{(1)}$ we create two different child solution vectors (*csv*), the $\mathrm{csv}_L^{(1)}$and $\mathrm{csv}_R^{(1)}$. Thanks to the generated $\mathrm{csv}_L^{(1)}$and $\mathrm{csv}_R^{(1)}$

we are able to perform fitness evaluation for each one of the corresponding cases. As soon as we complete the fitness evaluation process, we select the best child solution between the two variants $c_L^{(1)}$and $c_R^{(1)}$ with the assistance of the Pareto optimality framework. The same procedure is followed for $c_L^{(2)}$and $c_R^{(2)}$. The proposed methodology allows us to explore more efficiently the search space and move progressively towards higher fitness solutions. Whenever, there is not a clear winner i.e. strong or weak dominance, between the $c_L^{(1)}$and $c_R^{(1)}$, or respectively between the $c_L^{(2)}$and $c_R^{(2)}$ the generation of a random number allows the random choice of one of the two alternative child solutions.

The procedure of computing child solutions $c^{(1)}$ and $c^{(2)}$ from two parent solutions $y^{(1)}$ and $y^{(2)}$ under the Two Stage Crossover (TSX) operator is as follows:

1. Calculate the spread factor β:

$$\beta = 1 + \frac{2}{y^{(2)} - y^{(1)}} \, min\left[(y^{(1)} - y^{(l)}), (y^{(u)} - y^{(2)})\right]$$

2. Calculate parameter a :

$$\alpha = 2 - \beta^{-(\eta_c+1)}$$

3. Create a random r_b boolean value of true or false with equal probability

```
$if\ r_b = true$
    Go to *step 4. Execute steps 4 through 9.*
$if\ r_b = false$
    Go to *step 10*
```

4. Create 2 random numbers $u_L \in [0,\ 1/a]$ and $u_R \in (1/a, 1]$.

$u_L \longrightarrow$ [0, 1/α];

$u_R \longrightarrow$ (1/α, 1];

5. Find 2 parameters β_q^L and β_q^R with the assistance of the following polynomial probability distribution:

$$\beta_q^L = \left(au_L\right)^{1/(\eta_c+1)} \quad , \quad u_L \in [0,\ 1/a],$$

$$\beta_q^R = \left(\frac{1}{2 - au_R}\right)^{1/(\eta_c+1)} \quad , \quad u_R \in (1/a, 1]$$

6. Thus, instead of a unique value for $c^{(1)}$ and $c^{(2)}$, we obtain two evaluations for each child solution that correspond to β_q^L and β_q^R respectivelly:

$$c_L^{(1)} = 0.5\left[\left(y^{(1)} + y^{(2)}\right) - \beta_q^L |y^{(2)} - y^{(1)}|\right]$$
$$c_R^{(1)} = 0.5\left[\left(y^{(1)} + y^{(2)}\right) - \beta_q^R |y^{(2)} - y^{(1)}|\right]$$

$$c_L^{(2)} = 0.5\left[\left(y^{(1)} + y^{(2)}\right) + \beta_q^L |y^{(2)} - y^{(1)}|\right]$$
$$c_R^{(2)} = 0.5\left[\left(y^{(1)} + y^{(2)}\right) + \beta_q^R |y^{(2)} - y^{(1)}|\right]$$

7. We perform fitness evaluation for each variant child solution, by substituting the candidate solutions into the parent solution vectors.
8. We select the best variant between the $c_L^{(1)}$and $c_R^{(1)}$, based on the Pareto optimality framework. The same procedure is followed for $c_L^{(2)}$and $c_R^{(2)}$.
9. Whenever, there is not a clear winner i.e. strong or weak dominance, between the $c_L^{(1)}$and $c_R^{(1)}$, or respectively between the $c_L^{(2)}$and $c_R^{(2)}$ the generation of a random number allows the random choice of one of the two alternative child solutions.

10. Perform the SBX operator normally as described in section 2.

A typical crossover operator generates offsprings by recombining information from two or more parents. Crossover is probably the most important exploratory mechanism of a MOEA. This study intends to even further enhance the exploratory capabilities of the crossover operator, in particular of the well-known SBX operator by incorporating an efficient fitness evaluation mechanism.

For the fare comparison of the algorithms we made sure that the total number of fitness evaluations is the same for either configuration of the algorithm (i.e. either with the TSX operator or the SBX operator). For more details about this issue please advise section 4 of the chapter.

However, although we keep the total number of fitness evaluations the same, the proposed methodology differentiates the allocation of these function evaluations within the algorithm.

In particular, each generational cycle, for the configuration of the algorithm with the proposed methodology requires more fitness function evaluations, due to the additional evaluations introduced by the TSX operator, than the configuration of the algorithm with the SBX operator. As a result, the configuration of the algorithm with the TSX operator requires less generational cycles for the same number of real fitness function evaluations than the configuration of the algorithm with the SBX operator. This discrepancy in generational cycles between the configuration of the algorithm with the TSX operator and the configuration of the algorithm with the SBX operator creates a net benefit in terms of computational time for the proposed methodology. More details about this matter can be found in section 7.6.1 and Table 7.4.

Moreover, given the equal number of real fitness evaluations, any improvement in the performance of the algorithm from the implementation of the TSX operator will indicate the net benefit of the proposed methodology over the classical SBX operator. Figure 7.2 presents the probability of child solutions around the parent solutions for the bounded case of the TSX operator. For more details about the performance of the proposed TSX operator please advise sections 7.6.1 – 7.6.4 of the chapter.

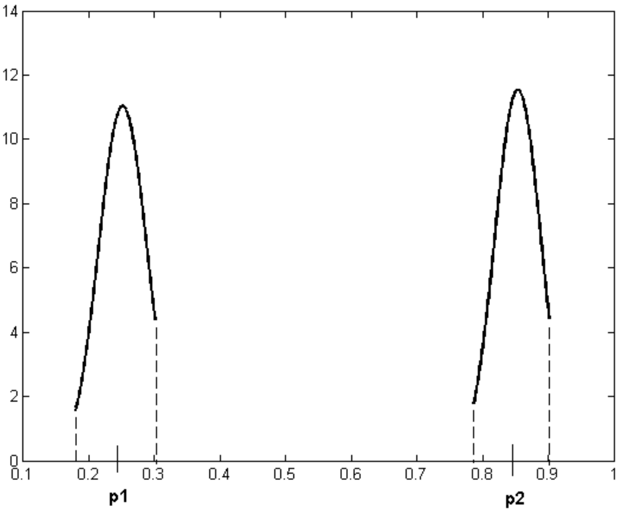


Fig. 7.2. Probabilities distributions for bounded cases of TSX operator

7.4 Experimental Environment

All algorithms have been implemented in Java and run on a 2.1GHz Windows Server 2012 machine with 6GB RAM. The jMetal [77] framework has been used to compare the performance of the proposed, Two Stage Crossover (TSX) operator against the Simulated Binary Crossover (SBX), the Differential Evolution (DE) and the Particle Swarm Optimization (PSO) with the assistance of a number of MOEAs. In all tests with the NSGAII, SPEA2 and MOCELL we use, binary tournament and polynomial mutation (PLM) [54] as, selection and mutation operator, respectively. The crossover probability is $P_c = 0.9$ and mutation probability is $P_m = 1/n$, where n is the number of decision variables. The distribution indices for the crossover and mutation operators are $\eta_c = 20$ and $\eta_m = 20$, respectively. For the MOEA/D we used DE with CR: 1.0 and F: 0.5 and PLM with $P_m = 1/n$ and $\eta_m = 20$. Finally, the SMPSO uses PLM with $P_m = 1/n$ and $\eta_m = 20$.

For identifying the best population size and function evaluations, the following configurations were tested for each algorithm: $N^{pop} = N^{arc} = 100, 200, 300, 400$, for *500, 1,000, 1,500* and *2,000* generations respectively. Accroding to Deb et al. [57] the archive size is usually chosen proportional to the population size. Please notice that an external archive (N^{arc}) is not applicable in the case of NSGAII. After testing the aforementioned configurations we reached the conclusion that the MOEAs approached their top performance for N^{pop} = 100, N^{arc} = *100* (where applicable) and *2,000* generations or *200,000* function evaluations. Beyond, these values the gains in performance are marginal if any. Thus, the Population size is set to *100*, using *200,000* function evaluations with *100* independent runs.

At this stage, we should highlight that for reasons of fairness regarding the function evaluations for the configuration of the MOEAs with the TSX when is compared with the SBX, DE and the PSO respectively, we applied the following strategy. In particular, as we explained in section 3, the Two Stage Crossover (TSX) operator incorporates a fitness function evaluation mechanism. As a result of this fitness function evaluation mechanism if we assign equal numbers of nominal function evaluations for the configuration of the MOEAs with the TSX and the configuration of the MOEAs with SBX, DE and PSO respectively, in reality we favor the configuration of the algorithm with TXS operator. Thus, in order to compensate for the additional fitness function evaluations introduced by the TSX operator we decided to use different number of function evaluations for the one and other configuration of the algorithm.

In order to determine the number of function evaluations for the one and the other configuration of the algorithm, the first step was to determine the additional burden imposed by the TSX operator. Assuming a crossover probability of $P_c = 0.9$ we found out that for every *1,000* evaluations of the algorithm the crossover operator is accessed on average *802* times. Please, recall from the presentation of the TSX operator in section 3 that a random r_b boolean value of true or false with equal probability determines if the SBX operator performed as normally or if we proceed with the TSX operator. Because the random r_b boolean variable assigns true or false with equal probability it is reasonable to assume that if the TSX operator is accessed on average *802* times half of them will be performed the SBX and the other half will be performed the fitness evaluation mechanism introduced by the TSX operator. Thus, so far we found out that for every *1,000* evaluations of the algorithm the fitness evaluation mechanism of the TSX operator is accessed on average *401* times. Moreover, from section 3, we notice that each time that the fitness evaluation mechanism of the TSX operator is accessed, 4 separate fitness function evaluations are performed. Thus, we reach the conclusion that on average every *1,000* function evaluations for the configuration of the MOEAs with SBX, DE or PSO correspond to

1,000 + 401x4 = 2,604 function evaluations for the configuration of the MOEAs with the TSX operator.

Please recall that above we found out that the configuration of the MOEAs reach its top performance for $N^{pop} = 100$, $N^{arc} = 100$ (where applicable) and *200,000* function evaluations. If we take into account the extra fitness function evaluations introduced by the TSX operator the 200,000 function evaluations of the configuration of the MOEAs with the SBX, DE or PSO correspond to *76,805* nominal function evaluations for the configuration of the MOEAs with the TSX operator. Thus, for reasons of fairness in the experimental part of this study we set *200,000* function evaluations for the configuration of the MOEAs with the SBX, DE or PSO against *76,805* nominal function evaluations for the configuration of the MOEAs with the TSX operator.

7.5 Performance Metrics

According to Liagkouras and Metaxiotis [142] a major issue that is related with the various performance metrics is the different focus of each particular metric. It is not uncommon an algorithm to perform well in a particular performance metric and not so well or even badly in another.

For that exact reason are favored by the researchers in the field performance metrics that are able to catch different aspects of the performance of a multiobjective algorithm [57] like Hypervolume (HV) or Epsilon indicator. According to the same study [142] 37.50% and 16.67% of the studies in the field make use of HV and Epsilon indicator respectively. Based on the above justification we decided to evaluate the performance of the examined algorithms with the assistance of Hypervolume and Epsilon indicator. Following the same principles we also employed the IGD indicator, as it is able to provide a measure for both convergence and diversity of the derived Pareto optimal set. Below, we provide a concise presentation of the aforementioned performance metrics.

Hypervolume

Hypervolume [228], is an indicator of both the convergence and diversity of an approximation set. Thus, given a set *S* containing *m* points in *n* objectives, the hypervolume of S is the size of the portion of objective space that is dominated by at least one point in S. The hypervolume of S is calculated relative to a reference point which is worse than (or equal to) every point in S in every objective. The greater the hypervolume of a solution the better considered the solution.

Inverted Generalization Distance (IGD)

Zitzler et al. [234] introduced the IGD quality indicator to measure how far the elements are in the Pareto optimal set from those in the set of non-dominated vectors found. A value of IGD equal to zero indicates that all the generated elements are in the Pareto front and they cover all the extension of the Pareto front. The IGD can be defined as follows:

$$\mathrm{IGD(P, S)} = \frac{\left(\sum_{i=1}^{|P|} d_i^q\right)^{1/q}}{|P|}$$

where $d_i = min_{\vec{s}\in S}||F(\vec{p}_i) - F(\vec{s}))||, \vec{p}_i \in P, q = 2\ and\ d_i$ is the smallest distance of $\vec{p} \in P$ to the closest solutions in S. The smaller the IGD value the better is the performance of the approach. The IGD metric is able to provide a measure for both convergence and diversity.

Epsilon Indicator I_ε

Zitzler et al. [234] introduced the epsilon indicator (I_ε). There are two versions of epsilon indicator the multiplicative and the additive. In this study we use the unary additive epsilon indicator. The basic usefulness of epsilon indicator of an approximation set A ($I_{\varepsilon+}$) is that it provides the minimum factor ε by which each point in the real front R can be added such that the resulting transformed approximation set is dominated by A. The additive epsilon indicator is a good measure of diversity, since it focuses on the worst case distance and reveals whether or not the approximation set has gaps in its trade-off solution set.

7.6 Experimental Results

A number of computational experiments were performed to test the performance of the proposed Two Stage Crossover (TSX) for the solution of the DTLZ1-7 set of test functions [60]. In section 6.1 the performance of the proposed TSX operator is assessed in comparison with the Simulated Binary Crossover (SBX) operator with the assistance of three well-known MOEAs, namely the NSGAII, the SPEA2 and the MOCELL, for the solution of the DTLZ1-7 set of test functions, with 3 objectives. In section 6.2 we test the proposed TSX operator against SBX with 5 objectives, as the DTLZ test instances are scalable to an arbitrary number of objectives. In section 6.3 and 6.4 we compare the performance of the proposed TSX operator against other popular and well-established reproduction techniques like the Differential Evolution (DE) and the Particle Swarm Optimization (PSO) respectively. In all cases the evaluation of the performance is based on a variety of metrics that assess both the proximity of the solutions to the Pareto front and their dispersion on it.

7.6.1 Experimental results of the TSX operator against the SBX with the assistance of DTLZ test instances with 3 objectives

Below, we utilize the Deb-Theile-Laumanns-Zitzler (DTLZ) [60] test suite to examine the efficacy of the proposed crossover operator in handling problems having more than two objectives. Specifically, the following seven MOPs named DTLZ1, DTLZ2, DTLZ3, DTLZ4, DTLZ5, DTLZ6 and DTLZ7 were used for comparing the proposed TSX operator against the SBX operator. The DTLZ test functions are a set of scalable problems with the ability to control difficulties in converging to the Pareto front and maintain the diversity of solutions. Please note that in this section we illustrate the DTLZ test functions for three objectives, for easiness of illustration, although they are generic and scalable to an arbitrary number of objectives [60].

Deb-Theile-Laumanns- Zitzler's function N.1 (DTLZ1) problem:

$$Min = \begin{cases} f_1(x) = \frac{1}{2}\, x_1 x_2 \ldots\, x_{M-1}\big(1 + g(X_M)\big), \\ f_2(x) = \frac{1}{2}\, x_1 x_2 \ldots\, (1 - x_{M-1})\big(1 + g(X_M)\big), \\ \vdots \\ f_{M-1}(x) = \frac{1}{2}\, x_1\, (1 - x_2)\big(1 + g(X_M)\big), \\ f_M(x) = \frac{1}{2}\, (1 - x_1)\big(1 + g(X_M)\big), \end{cases}$$

$$where \quad g(X_M) = 100\left[|X_M| + \sum_{x_i \in X_M} (x_i - 0.5)^2 - \cos\left(20\pi\,(x_i - 0.5)\right)\right],$$

$$for\ 0 \le x_i \le 1\ and\ 1 \le i \le n$$

Deb-Theile-Laumanns- Zitzler's function N.2 (DTLZ2) problem:

$$Min = \begin{cases} f_1(x) = \big(1 + g(X_M)\big)\cos\left(x_1\frac{\pi}{2}\right)\cos\left(x_2\frac{\pi}{2}\right) \ldots .\cos\left(x_{M-2}\frac{\pi}{2}\right)\cos\left(x_{M-1}\frac{\pi}{2}\right), \\ f_2(x) = \big(1 + g(X_M)\big)\cos\left(x_1\frac{\pi}{2}\right)\cos\left(x_2\frac{\pi}{2}\right) \ldots .\cos\left(x_{M-2}\frac{\pi}{2}\right) sin\left(x_{M-1}\frac{\pi}{2}\right), \\ f_3(x) = \big(1 + g(X_M)\big)\cos\left(x_1\frac{\pi}{2}\right)\cos\left(x_2\frac{\pi}{2}\right) \ldots .\, sin\left(x_{M-2}\frac{\pi}{2}\right), \\ \vdots \\ f_{M-1}(x) = \big(1 + g(X_M)\big)\cos\left(x_1\frac{\pi}{2}\right) sin\left(x_2\frac{\pi}{2}\right), \\ f_M(x) = \big(1 + g(X_M)\big) sin\left(x_1\frac{\pi}{2}\right), \end{cases}$$

$$where \quad g(X_M) = \textstyle\sum_{x_i \in X_M}(x_i - 0.5)^2$$

$$for\ 0 \le x_i \le 1\ and\ 1 \le i \le n$$

Deb-Theile-Laumanns- Zitzler's function N.3 (DTLZ3) problem:

$$Min = \begin{cases} f_1(x) = \big(1 + g(X_M)\big)\cos\left(x_1\frac{\pi}{2}\right)\cos\left(x_2\frac{\pi}{2}\right) \ldots .\cos\left(x_{M-2}\frac{\pi}{2}\right)\cos\left(x_{M-1}\frac{\pi}{2}\right), \\ f_2(x) = \big(1 + g(X_M)\big)\cos\left(x_1\frac{\pi}{2}\right)\cos\left(x_2\frac{\pi}{2}\right) \ldots .\cos\left(x_{M-2}\frac{\pi}{2}\right) sin\left(x_{M-1}\frac{\pi}{2}\right), \\ f_3(x) = \big(1 + g(X_M)\big)\cos\left(x_1\frac{\pi}{2}\right)\cos\left(x_2\frac{\pi}{2}\right) \ldots .\, sin\left(x_{M-2}\frac{\pi}{2}\right), \\ \vdots \\ f_{M-1}(x) = \big(1 + g(X_M)\big)\cos\left(x_1\frac{\pi}{2}\right) sin\left(x_2\frac{\pi}{2}\right), \\ f_M(x) = \big(1 + g(X_M)\big) sin\left(x_1\frac{\pi}{2}\right), \end{cases}$$

$$where \quad g(X_M) = 100\left[|X_M| + \textstyle\sum_{x_i \in X_M}(x_i - 0.5)^2 - \cos\left(20\pi\,(x_i - 0.5)\right)\right]$$

$$for\ 0 \le x_i \le 1\ and\ 1 \le i \le n$$

Deb-Theile-Laumanns- Zitzler's function N.4 (DTLZ4) problem:

$$Min = \begin{cases} f_1(x) = \left(1+ g(X_M)\right)\cos\left(x_1^a\frac{\pi}{2}\right)\cos\left(x_2^a\frac{\pi}{2}\right)\dots.\cos\left(x_{M-2}^a\frac{\pi}{2}\right)\cos\left(x_{M-1}^a\frac{\pi}{2}\right), \\ f_2(x) = \left(1+ g(X_M)\right)\cos\left(x_1^a\frac{\pi}{2}\right)\cos\left(x_2^a\frac{\pi}{2}\right)\dots.\cos\left(x_{M-2}^a\frac{\pi}{2}\right)sin\left(x_{M-1}^a\frac{\pi}{2}\right), \\ f_3(x) = \left(1+ g(X_M)\right)\cos\left(x_1^a\frac{\pi}{2}\right)\cos\left(x_2^a\frac{\pi}{2}\right)\dots.sin\left(x_{M-2}^a\frac{\pi}{2}\right), \\ \vdots \\ f_{M-1}(x) = \left(1+ g(X_M)\right)\cos\left(x_1^a\frac{\pi}{2}\right)sin\left(x_2^a\frac{\pi}{2}\right), \\ f_M(x) = \left(1+ g(X_M)\right)sin\left(x_1^a\frac{\pi}{2}\right), \end{cases}$$

$$where \quad g(X_M) = \textstyle\sum_{x_i\in X_M}(x_i - 0.5)^2$$

$$for\ 0 \le x_i \le 1\ and\ 1 \le i \le n$$

Deb-Theile-Laumanns- Zitzler's function N.5 (DTLZ5) problem:

$$Min = \begin{cases} f_1(x) = \left(1+ g(X_M)\right)\cos\left(\theta_1\frac{\pi}{2}\right)\cos\left(\theta_2\frac{\pi}{2}\right)\dots.\cos\left(\theta_{M-2}\frac{\pi}{2}\right)\cos\left(\theta_{M-1}\frac{\pi}{2}\right), \\ f_2(x) = \left(1+ g(X_M)\right)\cos\left(\theta_1\frac{\pi}{2}\right)\cos\left(\theta_2\frac{\pi}{2}\right)\dots.\cos\left(\theta_{M-2}\frac{\pi}{2}\right)sin\left(\theta_{M-1}\frac{\pi}{2}\right), \\ f_3(x) = \left(1+ g(X_M)\right)\cos\left(\theta_1\frac{\pi}{2}\right)\cos\left(\theta_2\frac{\pi}{2}\right)\dots.sin\left(\theta_{M-2}\frac{\pi}{2}\right), \\ \vdots \\ f_{M-1}(x) = \left(1+ g(X_M)\right)\cos\left(\theta_1\frac{\pi}{2}\right)sin\left(\theta_2\frac{\pi}{2}\right), \\ f_M(x) = \left(1+ g(X_M)\right)sin\left(\theta_1\frac{\pi}{2}\right), \end{cases}$$

$$where \quad \theta_i = \frac{\pi}{4(1+g(X_M))}(1+2g(X_M)x_i) \quad for\ i = 2, 3, \dots, (M-1),$$

$$g(X_M) = \textstyle\sum_{x_i\in X_M}(x_i - 0.5)^2 \quad for\ 0 \le x_i \le 1\ and\ 1 \le i \le n$$

Deb-Theile-Laumanns- Zitzler's function N.6 (DTLZ6) problem:

$$Min = \begin{cases} f_1(x) = \left(1+ g(X_M)\right)\cos\left(\theta_1\frac{\pi}{2}\right)\cos\left(\theta_2\frac{\pi}{2}\right)\dots.\cos\left(\theta_{M-2}\frac{\pi}{2}\right)\cos\left(\theta_{M-1}\frac{\pi}{2}\right), \\ f_2(x) = \left(1+ g(X_M)\right)\cos\left(\theta_1\frac{\pi}{2}\right)\cos\left(\theta_2\frac{\pi}{2}\right)\dots.\cos\left(\theta_{M-2}\frac{\pi}{2}\right)sin\left(\theta_{M-1}\frac{\pi}{2}\right), \\ f_3(x) = \left(1+ g(X_M)\right)\cos\left(\theta_1\frac{\pi}{2}\right)\cos\left(\theta_2\frac{\pi}{2}\right)\dots.sin\left(\theta_{M-2}\frac{\pi}{2}\right), \\ \vdots \\ f_{M-1}(x) = \left(1+ g(X_M)\right)\cos\left(\theta_1\frac{\pi}{2}\right)sin\left(\theta_2\frac{\pi}{2}\right), \\ f_M(x) = \left(1+ g(X_M)\right)sin\left(\theta_1\frac{\pi}{2}\right), \end{cases}$$

$$where \quad \theta_i = \frac{\pi}{4(1+g(X_M))}(1+2g(X_M)x_i) \quad for\ i = 2, 3, \dots, (M-1),$$

$$g(X_M) = \textstyle\sum_{x_i\in X_M} x_i^{0.1} \quad for\ 0 \le x_i \le 1\ and\ 1 \le i \le n$$

Deb-Theile-Laumanns- Zitzler's function N.7 (DTLZ7) problem:

$$Min = \begin{cases} f_1(X_1) = x_1, \\ f_2(X_2) = x_2, \\ \vdots \\ f_{M-1}(X_{M-1}) = x_{M-1}, \\ f_M(X) = \big(1 + g(X_M)\big)\, h(f_1, f_2, \dots, f_{M-1}, g), \end{cases}$$

$$where \quad g(X_M) = 1 + \frac{9}{|X_M|} \sum_{x_i \in X_M} x_i$$

$$h(f_1, f_2, \dots, f_{M-1}, g) = M - \sum_{i=1}^{M-1} \left[\frac{f_i}{1+g}\, (1 + \sin(3\pi f_i))\right]$$

$$for\, 0 \le x_i \le 1\, and\, 1 \le i \le n$$

Table 7.1 presents the results of DTLZ1-7 test functions with 3 objectives. Specifically, it presents the mean, standard deviation (STD), median and interquartile range (IQR) of all the independent runs carried out for Hypervolume (HV), Inverted Generalization Distance (IGD) and Epsilon indicator respectively.

Clearly, regarding the HV [94], [228] indicator the higher the value (i.e. the greater the hypervolume) the better the computed front. HV is able of capturing in a single number both the closeness of the solutions to the optimal set and to a certain degree, the spread of the solutions across the objective space [221]. The second indicator the IGD [234] measure how far the elements are in the Pareto optimal set from those in the set of non-dominated vectors found. The smaller the value of this indicator, the better the distribution of the solutions. A zero value of IGD indicates that all the generated elements are in the Pareto front and they cover all the extension of the Pareto front.

The third indicator, the Epsilon [234] is a measure of the smaller distance that a solution set *A*, needs to be changed in such a way that it dominates the optimal Pareto front of this problem. Obviously the smaller the value of this indicator, the better the derived solution set.

Table 7.2 use boxplots to present graphically the performance of the NSGAII, the SPEA2 and the MOCELL under two different configurations, the TSX and the SBX respectively, for HV IGD and Epsilon performance indicators. Boxplot is a convenient way of graphically depicting groups of numerical data through their quartiles.

Table 7.1. Mean, Std, Median and Iqr for Hv, IGD and Epsilon: TSX vs SBX - DTLZ with 3 objectives

Problem: DTLZ1	NSGAII		SPEA2		MOCELL	
	TSX	SBX	TSX	SBX	TSX	SBX
HV. Mean and Std	7.64e-01$_{4.8e-03}$	7.62e-01$_{6.5e-03}$	7.88e-01$_{4.3e-04}$	7.87e-01$_{5.7e-04}$	7.68e-01$_{2.6e-03}$	7.61e-01$_{9.3e-03}$
HV. Median and IQR	7.65e-01$_{6.4e-03}$	7.62e-01$_{6.8e-03}$	7.88e-01$_{5.4e-04}$	7.87e-01$_{8.6e-04}$	7.67e-01$_{4.2e-03}$	7.61e-01$_{1.2e-02}$
IGD. Mean and Std	6.33e-04$_{4.2e-05}$	5.99e-04$_{4.8e-05}$	4.37e-04$_{7.3e-06}$	4.39e-04$_{9.7e-06}$	5.95e-04$_{3.6e-05}$	5.90e-04$_{1.0e-04}$
IGD. Median and IQR	6.35e-04$_{5.6e-05}$	5.93e-04$_{3.7e-05}$	4.37e-04$_{8.5e-06}$	4.37e-04$_{1.2e-05}$	5.86e-04$_{5.0e-05}$	5.56e-04$_{4.8e-05}$
EPSILON. Mean and Std	5.38e-02$_{5.9e-03}$	5.37e-02$_{6.3e-03}$	3.38e-02$_{1.7e-03}$	3.37e-02$_{2.1e-03}$	4.86e-02$_{4.8e-03}$	5.60e-02$_{9.9e-03}$
EPSILON. Median and IQR	5.37e-02$_{8.0e-03}$	5.28e-02$_{7.6e-03}$	3.34e-02$_{2.1e-03}$	3.34e-02$_{2.5e-03}$	4.74e-02$_{7.3e-03}$	5.47e-02$_{1.1e-02}$

Problem: DTLZ2	NSGAII		SPEA2		MOCELL	
	TSX	SBX	TSX	SBX	TSX	SBX
HV. Mean and Std	3.83e-01$_{5.0e-03}$	3.75e-01$_{6.7e-03}$	4.11e-01$_{1.7e-03}$	4.05e-01$_{1.7e-03}$	3.86e-01$_{4.9e-03}$	3.82e-01$_{3.9e-03}$
HV. Median and IQR	3.83e-01$_{5.0e-03}$	3.77e-01$_{9.1e-03}$	4.11e-01$_{1.4e-03}$	4.05e-01$_{1.9e-03}$	3.86e-01$_{5.6e-03}$	3.82e-01$_{4.0e-03}$
IGD. Mean and Std	7.95e-04$_{4.9e-05}$	7.76e-04$_{4.7e-05}$	5.77e-04$_{7.7e-06}$	5.89e-04$_{1.2e-05}$	7.51e-04$_{4.4e-05}$	7.44e-04$_{3.2e-05}$
IGD. Median and IQR	7.83e-04$_{6.5e-05}$	7.71e-04$_{6.0e-05}$	5.77e-04$_{1.1e-05}$	5.87e-04$_{1.5e-05}$	7.44e-04$_{5.1e-05}$	7.42e-04$_{3.4e-05}$
EPSILON. Mean and Std	1.20e-01$_{1.7e-02}$	1.25e-01$_{1.4e-02}$	7.88e-02$_{7.6e-03}$	8.16e-02$_{7.9e-03}$	1.20e-01$_{1.4e-02}$	1.24e-01$_{1.2e-02}$
EPSILON. Median and IQR	1.21e-01$_{2.2e-02}$	1.25e-01$_{1.3e-02}$	7.67e-02$_{8.9e-03}$	8.05e-02$_{1.2e-02}$	1.20e-01$_{1.9e-02}$	1.26e-01$_{1.5e-02}$

Problem: DTLZ3	NSGAII		SPEA2		MOCELL	
	TSX	SBX	TSX	SBX	TSX	SBX
HV. Mean and Std	3.82e-01$_{7.7e-03}$	3.79e-01$_{7.4e-03}$	4.13e-01$_{3.1e-03}$	4.14e-01$_{1.3e-03}$	3.87e-01$_{5.0e-03}$	3.88e-01$_{4.7e-03}$
HV. Median and IQR	3.84e-01$_{8.9e-03}$	3.79e-01$_{8.7e-03}$	4.14e-01$_{3.7e-03}$	4.14e-01$_{1.7e-03}$	3.86e-01$_{6.7e-03}$	3.89e-01$_{6.7e-03}$
IGD. Mean and Std	1.28e-03$_{7.2e-05}$	1.23e-03$_{5.8e-05}$	9.14e-04$_{1.1e-05}$	9.11e-04$_{1.5e-05}$	1.19e-03$_{6.4e-05}$	1.18e-03$_{5.7e-05}$
IGD. Median and IQR	1.27e-03$_{8.5e-05}$	1.23e-03$_{6.8e-05}$	9.14e-04$_{1.5e-05}$	9.09e-04$_{2.0e-05}$	1.18e-03$_{8.1e-05}$	1.18e-03$_{8.5e-05}$
EPSILON. Mean and Std	1.28e-01$_{1.9e-02}$	1.33e-01$_{2.1e-02}$	7.29e-02$_{6.6e-03}$	7.82e-02$_{1.3e-02}$	1.21e-01$_{1.8e-02}$	1.18e-01$_{1.8e-02}$
EPSILON. Median and IQR	1.24e-01$_{2.4e-02}$	1.32e-01$_{2.7e-02}$	7.14e-02$_{7.5e-03}$	7.46e-02$_{1.1e-02}$	1.20e-01$_{2.5e-02}$	1.12e-01$_{2.5e-02}$

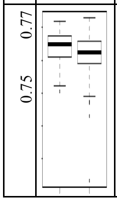
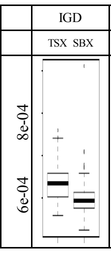
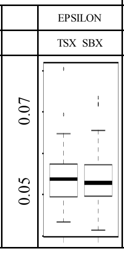
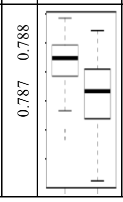
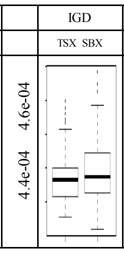
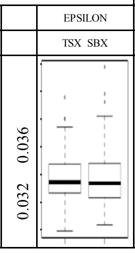
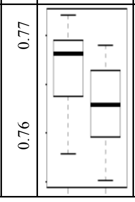
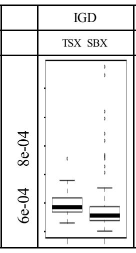
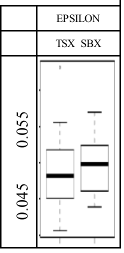
Problem: DTLZ4	NSGAII		SPEA2		MOCELL	
	TSX	SBX	TSX	SBX	TSX	SBX
HV. Mean and Std	3.81e-01$_{4.9e-03}$	3.76e-01$_{4.7e-03}$	3.95e-01$_{4.3e-02}$	3.31e-01$_{1.1e-01}$	2.31e-01$_{1.9e-01}$	2.68e-01$_{1.8e-01}$
HV. Median and IQR	3.83e-01$_{9.1e-03}$	3.76e-01$_{6.3e-03}$	4.05e-01$_{2.7e-03}$	3.98e-01$_{1.9e-01}$	3.81e-01$_{3.9e-01}$	3.80e-01$_{3.8e-01}$
IGD. Mean and Std	1.26e-03$_{1.1e-03}$	1.28e-03$_{8.2e-04}$	1.54e-03$_{2.3e-03}$	3.03e-03$_{3.0e-03}$	3.18e-03$_{3.6e-03}$	3.32e-03$_{3.6e-03}$
IGD. Median and IQR	1.05e-03$_{1.5e-04}$	1.18e-03$_{1.6e-04}$	7.23e-04$_{4.7e-05}$	8.53e-04$_{4.6e-03}$	1.16e-03$_{4.3e-03}$	1.22e-03$_{8.2e-03}$
EPSILON. Mean and Std	1.08e-01$_{1.2e-02}$	1.11e-01$_{1.6e-02}$	9.69e-02$_{1.2e-01}$	2.58e-01$_{2.9e-01}$	4.54e-01$_{4.3e-01}$	3.65e-01$_{4.0e-01}$
EPSILON. Median and IQR	1.12e-01$_{1.9e-02}$	1.06e-01$_{2.9e-02}$	6.82e-02$_{8.4e-03}$	7.52e-02$_{5.6e-01}$	1.23e-01$_{8.7e-01}$	1.05e-01$_{8.8e-01}$

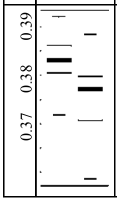
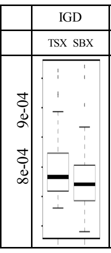
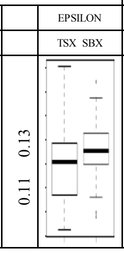
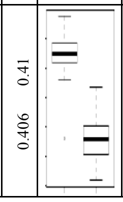
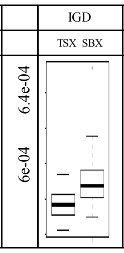
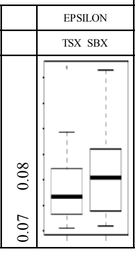
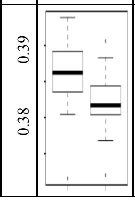
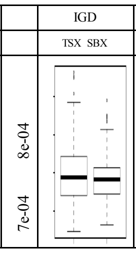
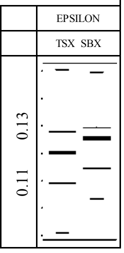
Problem: DTLZ5	NSGAII		SPEA2		MOCELL	
	TSX	SBX	TSX	SBX	TSX	SBX
HV. Mean and Std	9.31e-02$_{1.7e-04}$	9.28e-02$_{2.0e-04}$	9.35e-02$_{1.2e-04}$	9.34e-02$_{1.0e-04}$	9.41e-02$_{2.2e-05}$	9.41e-02$_{2.0e-05}$
HV. Median and IQR	9.31e-02$_{2.1e-04}$	9.28e-02$_{3.1e-04}$	9.35e-02$_{1.9e-04}$	9.34e-02$_{1.4e-04}$	9.41e-02$_{3.1e-05}$	9.41e-02$_{2.8e-05}$
IGD. Mean and Std	1.94e-05$_{1.1e-06}$	2.03e-05$_{1.0e-06}$	1.48e-05$_{3.5e-07}$	1.55e-05$_{4.0e-07}$	1.42e-05$_{3.5e-07}$	1.42e-05$_{3.2e-07}$
IGD.Median and IQR	1.93e-05$_{1.1e-06}$	2.04e-05$_{1.3e-06}$	1.48e-05$_{4.8e-07}$	1.54e-05$_{4.9e-07}$	1.42e-05$_{4.5e-07}$	1.41e-05$_{4.1e-07}$
EPSILON. Mean and Std	9.79e-03$_{1.2e-03}$	1.10e-02$_{2.0e-03}$	7.03e-03$_{1.2e-03}$	7.49e-03$_{1.1e-03}$	4.60e-03$_{5.9e-04}$	4.40e-03$_{5.4e-04}$
EPSILON. Median and IQR	9.78e-03$_{2.1e-03}$	1.01e-02$_{1.8e-03}$	7.02e-03$_{1.5e-03}$	7.25e-03$_{1.2e-03}$	4.37e-03$_{6.2e-04}$	4.28e-03$_{1.6e-04}$

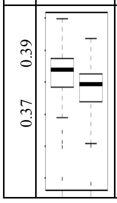
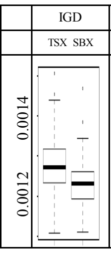
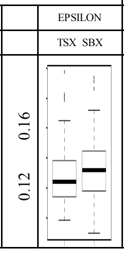
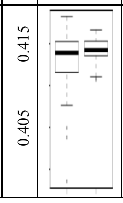
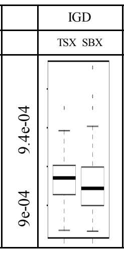
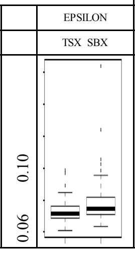
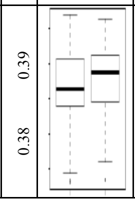
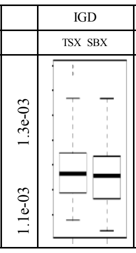
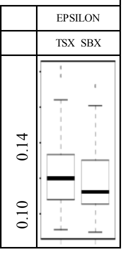
Problem: DTLZ6	NSGAII		SPEA2		MOCELL	
	TSX	SBX	TSX	SBX	TSX	SBX
HV. Mean and Std	8.05e-02$_{1.1e-02}$	8.35e-02$_{1.1e-02}$	9.16e-02$_{5.5e-03}$	8.96e-02$_{7.7e-03}$	9.40e-02$_{3.6e-03}$	9.01e-02$_{8.4e-03}$
HV. Median and IQR	7.83e-02$_{2.2e-02}$	8.65e-02$_{1.8e-02}$	9.46e-02$_{3.5e-03}$	9.46e-02$_{1.5e-02}$	9.50e-02$_{5.0e-05}$	9.50e-02$_{1.4e-02}$
IGD. Mean and Std	1.40e-04$_{9.0e-05}$	1.36e-04$_{9.7e-05}$	1.06e-04$_{7.6e-05}$	7.09e-05$_{6.2e-05}$	3.50e-05$_{8.3e-06}$	6.90e-05$_{7.4e-05}$
IGD. Median and IQR	1.06e-04$_{1.4e-04}$	1.15e-04$_{1.3e-04}$	8.62e-05$_{1.3e-04}$	3.67e-05$_{1.0e-04}$	3.42e-05$_{1.2e-06}$	3.40e-05$_{1.9e-06}$
EPSILON. Mean and Std	2.63e-02$_{1.3e-02}$	2.25e-02$_{1.3e-02}$	8.28e-03$_{5.7e-03}$	1.12e-02$_{8.7e-03}$	5.29e-03$_{4.0e-03}$	9.74e-03$_{9.9e-03}$
EPSILON. Median and IQR	2.72e-02$_{2.3e-02}$	1.98e-02$_{1.7e-02}$	5.30e-03$_{3.8e-03}$	5.77e-03$_{1.6e-02}$	4.24e-03$_{3.6e-04}$	4.16e-03$_{1.5e-02}$

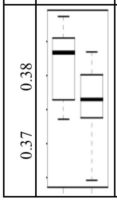
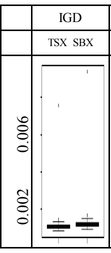
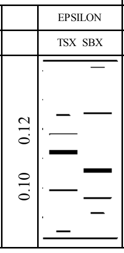
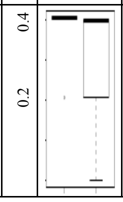
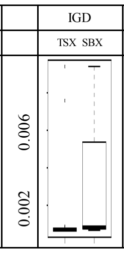
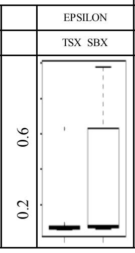
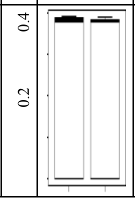
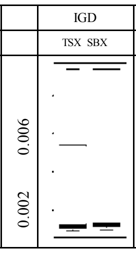
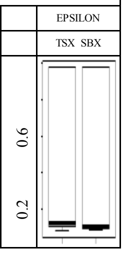
Problem: DTLZ7	NSGAII		SPEA2		MOCELL	
	TSX	SBX	TSX	SBX	TSX	SBX
HV. Mean and Std	2.86e-01$_{3.8e-03}$	2.84e-01$_{3.7e-03}$	3.01e-01$_{1.2e-03}$	2.96e-01$_{7.9e-03}$	2.65e-01$_{3.2e-02}$	2.68e-01$_{2.3e-02}$
HV. Median and IQR	2.86e-01$_{7.2e-03}$	2.85e-01$_{4.2e-03}$	3.01e-01$_{1.4e-03}$	2.98e-01$_{3.6e-03}$	2.86e-01$_{5.2e-02}$	2.70e-01$_{3.0e-02}$
IGD. Mean and Std	2.19e-03$_{1.5e-04}$	2.37e-03$_{1.4e-03}$	1.75e-03$_{4.9e-05}$	2.55e-03$_{3.1e-03}$	1.11e-02$_{9.0e-03}$	8.12e-03$_{7.6e-03}$
IGD. Median and IQR	2.16e-03$_{2.0e-04}$	2.23e-03$_{1.9e-04}$	1.75e-03$_{6.2e-05}$	1.84e-03$_{9.7e-05}$	1.52e-02$_{1.4e-02}$	2.45e-03$_{1.3e-02}$
EPSILON. Mean and Std	1.36e-01$_{4.2e-02}$	1.29e-01$_{3.0e-02}$	8.91e-02$_{9.2e-03}$	1.51e-01$_{2.6e-01}$	9.02e-01$_{1.0e+00}$	8.22e-01$_{7.8e-01}$
EPSILON. Median and IQR	1.25e-01$_{4.9e-02}$	1.20e-01$_{3.7e-02}$	8.75e-02$_{1.0e-02}$	9.10e-02$_{1.2e-02}$	1.40e-01$_{1.8e+00}$	7.47e-01$_{1.2e+00}$

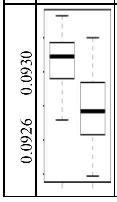
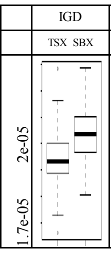
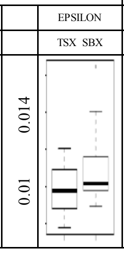
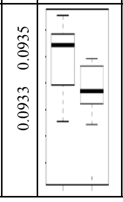
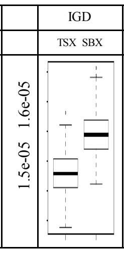
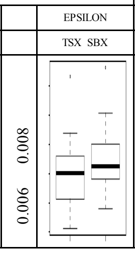
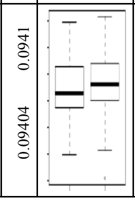
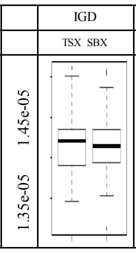
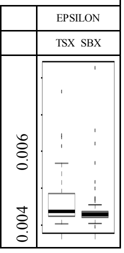
Table 7.2. Boxplots for Hv, IGD and Epsilon: TSX vs SBX - DTLZ with 3 objectives

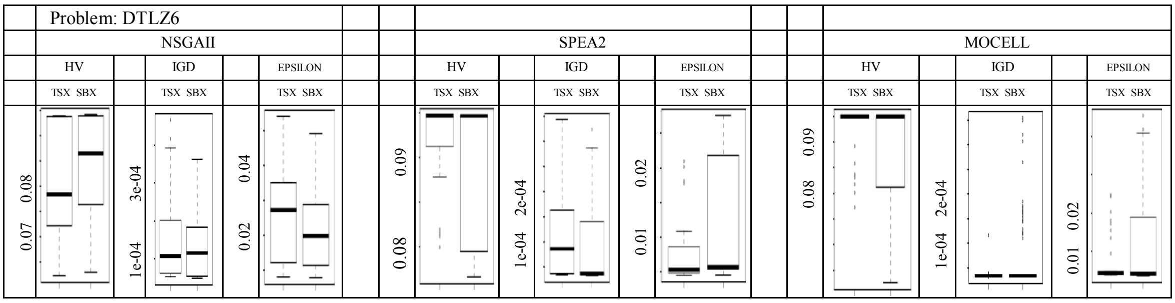
	Problem: DTLZ1																		
	NSGAII								SPEA2								MOCELL		
	HV		IGD		EPSILON				HV		IGD		EPSILON				HV		IGD		EPSILON
	TSX SBX		TSX SBX		TSX SBX				TSX SBX		TSX SBX		TSX SBX				TSX SBX		TSX SBX		TSX SBX
																			

	Problem: DTLZ2																		
	NSGAII								SPEA2								MOCELL		
	HV		IGD		EPSILON				HV		IGD		EPSILON				HV		IGD		EPSILON
	TSX SBX		TSX SBX		TSX SBX				TSX SBX		TSX SBX		TSX SBX				TSX SBX		TSX SBX		TSX SBX
																			

	Problem: DTLZ3																		
	NSGAII								SPEA2								MOCELL		
	HV		IGD		EPSILON				HV		IGD		EPSILON				HV		IGD		EPSILON
	TSX SBX		TSX SBX		TSX SBX				TSX SBX		TSX SBX		TSX SBX				TSX SBX		TSX SBX		TSX SBX
																			

	Problem: DTLZ4																		
	NSGAII								SPEA2								MOCELL		
	HV		IGD		EPSILON				HV		IGD		EPSILON				HV		IGD		EPSILON
	TSX SBX		TSX SBX		TSX SBX				TSX SBX		TSX SBX		TSX SBX				TSX SBX		TSX SBX		TSX SBX
																			

	Problem: DTLZ5																		
	NSGAII								SPEA2								MOCELL		
	HV		IGD		EPSILON				HV		IGD		EPSILON				HV		IGD		EPSILON
	TSX SBX		TSX SBX		TSX SBX				TSX SBX		TSX SBX		TSX SBX				TSX SBX		TSX SBX		TSX SBX
																			



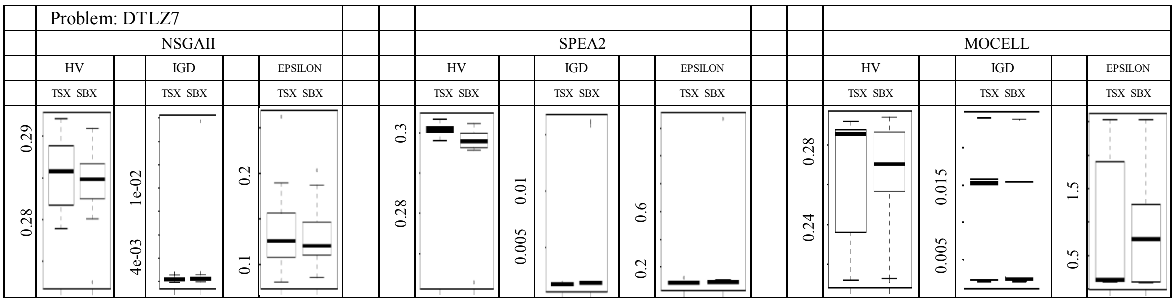


Table 7.3, makes use of a nonparametric statistical test called Wilcoxon’s rank-sum test for independent samples [66] at the 5% significance level in order to judge whether the results obtained with the best performing algorithm differ from the final results of rest of the competitors in a statistically significant way. In particular in Table 7.3 we present if the results of NSGAII, SPEA2 and MOCELL that are derived under the two different configurations, namely the TSX and the SBX, are statistically significant or not. In Table 7.3, three different symbols are used. In particular “ - ” indicates that there is not statistical significance between the algorithms. “↑” means that the algorithm in the row has yielded better results than the algorithm in the column with confidence and “ ↓ ” is used when the algorithm in the column is statistically better than the algorithm in the row.

Table 7.3. Wilcoxon Test for Hv, IGD and Epsilon: TSX vs SBX - DTLZ with 3 objectives

Problem			NSGAII with SBX		SPEA2 with SBX		MOCELL with SBX
DTLZ1	HV. Mean and Std	NSGAII with TSX	↑	SPEA2 with TSX	–	MOCELL with TSX	–
	HV. Median and IQR		↑		–		–
	IGD. Mean and Std		↓		–		–
	IGD. Median and IQR		↓		–		–
	EPSILON. Mean and Std		↑		↑		–
	EPSILON. Median and IQR		↑		↑		–

Problem			NSGAII with SBX		SPEA2 with SBX		MOCELL with SBX
DTLZ2	HV. Mean and Std	NSGAII with TSX	↑	SPEA2 with TSX	↑	MOCELL with TSX	↑
	HV. Median and IQR		↑		↑		↑
	IGD. Mean and Std		–		↑		–
	IGD. Median and IQR		–		↑		–
	EPSILON. Mean and Std		↑		–		↑
	EPSILON. Median and IQR		↑		–		↑

Problem			NSGAII with SBX		SPEA2 with SBX		MOCELL with SBX
DTLZ3	HV. Mean and Std	NSGAII with TSX	↑	SPEA2 with TSX	–	MOCELL with TSX	↑
	HV. Median and IQR		↑		–		↑
	IGD. Mean and Std		↓		–		–
	IGD. Median and IQR		↓		–		–
	EPSILON. Mean and Std		–		–		–
	EPSILON. Median and IQR		–		–		–

Problem			NSGAII with SBX		SPEA2 with SBX		MOCELL with SBX
DTLZ4	HV. Mean and Std	NSGAII with TSX	↑	SPEA2 with TSX	–	MOCELL with TSX	–
	HV. Median and IQR		↑		–		–
	IGD.Mean and Std		↑		↑		–
	IGD. Median and IQR		↑		↑		–
	EPSILON. Mean and Std		–		–		–
	EPSILON. Median and IQR		–		–		–

Problem			NSGAII with SBX		SPEA2 with SBX		MOCELL with SBX
DTLZ5	HV. Mean and Std	NSGAII with TSX	↑	SPEA2 with TSX	–	MOCELL with TSX	↑
	HV. Median and IQR		↑		–		↑
	IGD. Mean and Std		↑		↑		–
	IGD. Median and IQR		↑		↑		–
	EPSILON. Mean and Std		–		–		–
	EPSILON. Median and IQR		–		–		–

Problem			NSGAII with SBX		SPEA2 with SBX		MOCELL with SBX
DTLZ6	HV. Mean and Std	NSGAII with TSX	↓	SPEA2 with TSX	–	MOCELL with TSX	–
	HV. Median and IQR		↓		–		–
	IGD. Mean and Std		–		–		–
	IGD. Median and IQR		–		–		–
	EPSILON. Mean and Std		↑		↑		↑
	EPSILON. Median and IQR		↑		↑		↑

Problem			NSGAII with SBX		SPEA2 with SBX		MOCELL with SBX
DTLZ7	HV. Mean and Std	NSGAII with TSX	—	SPEA2 with TSX	—	MOCELL with TSX	—
	HV. Median and IQR		—		—		—
	IGD. Mean and Std		—		—		—
	IGD. Median and IQR		—		—		—
	EPSILON. Mean and Std		↑		—		↑
	EPSILON. Median and IQR		↑		—		↑

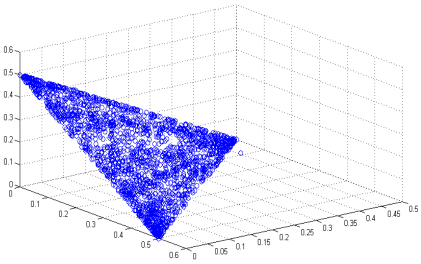


Fig. 7.3. Problem: DTLZ1. MOCELL + TSX operator

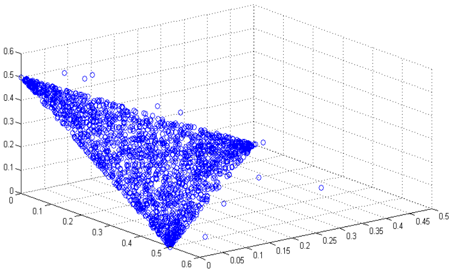


Fig. 7.4. Problem: DTLZ1. MOCELL + SBX operator

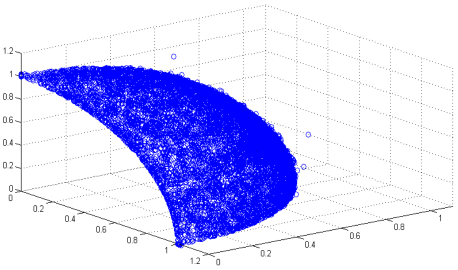


Fig. 7.5. Problem: DTLZ3. NSGAII + TSX operator

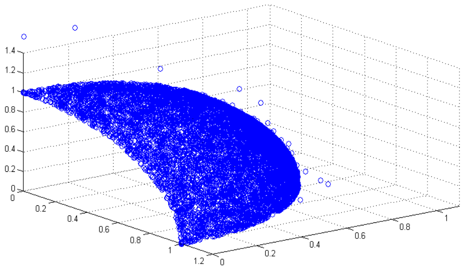


Fig. 7.6. Problem: DTLZ3. NSGAII + SBX operator

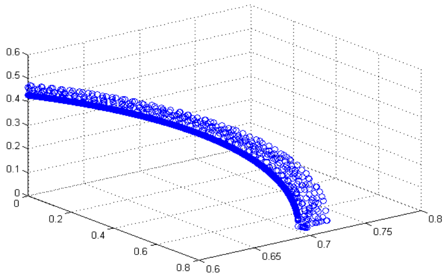


Fig. 7.7. Problem: DTLZ6. MOCELL + TSX operator

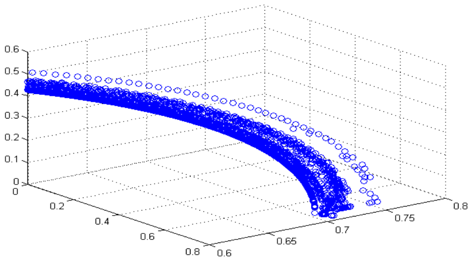
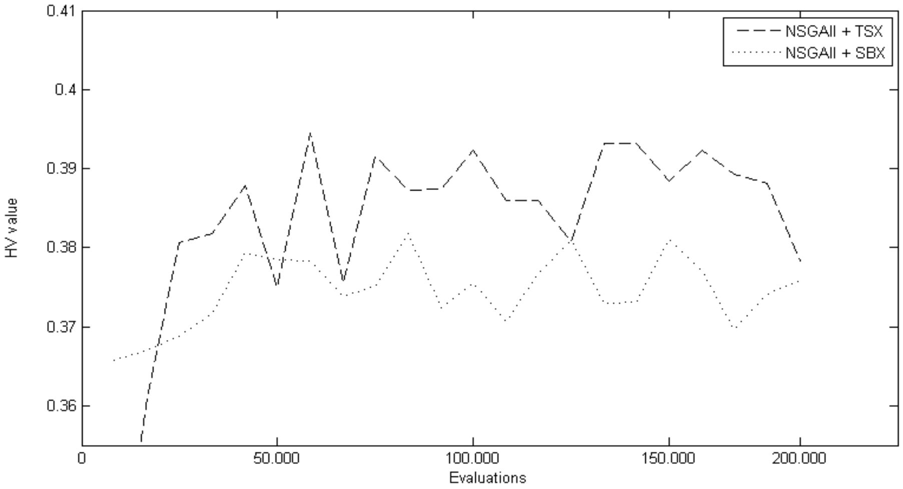


Fig. 7.8. Problem: DTLZ6. MOCELL + SBX operator

Fig. 7.9. DTLZ2: Evolution of HV: NSGAII+TSX and NSGAII+SBX



Below, we analyze the results obtained by applying the proposed Two Stage Crossover (TSX) and the Simulated Binary Crossover (SBX), to the NSGAII, SPEA2 and MOCELL for solving the DTLZ 1-7 [60] set of test functions with 3 objectives. The assessment of the performance of the proposed crossover operator is done with the assistance of three well known performance indicators, namely the *Hypervolume*, *IGD* and *Epsilon* indicator.

Examining the results of the first indicator, the *HV*, we notice that the TSX operator performs better than the classical SBX operator when is applied to the NSGAII, SPEA2 and MOCELL. Figure 10 reveals the relevant results for the performance of TSX operator regarding the *HV* indicator. The Wilcoxon rank-sum test (see Table 7.3) validates that the observed difference in TSX and SBX performance is statistically significant with 95% confidence. Analytically, in 43% of the cases the TSX yields better results with confidence than the conventional configuration of the NSGAII, SPEA2 and MOCELL with the SBX operator. Also, in 52% of the cases there was not statistical significance between the TSX and SBX. Finally, in 5% of the cases the SBX outperformed the TSX operator regarding the *HV metric*.

Regarding the *IGD indicator* (see Fig. 7.11) the same methodology is applied. In particular the Wilcoxon rank-sum test helped us to identify that in 24% of the cases the TSX yields better results with confidence than the conventional configuration of the NSGAII, SPEA2 and MOCELL with the SBX operator. In 67% of the cases there was not statistical significance between the TSX and SBX regarding the *IGD* *indicator* and in 9% of the examined cases the classical SBX operator generated better results with confidence than the proposed methodology.

Figure 12 reveals the relevant results for the performance of TSX regarding the *Epsilon* indicator. In particular the Wilcoxon rank-sum test helps us to identify that in 43% of the cases the TSX yields better results with confidence than the conventional configuration of the NSGAII, SPEA2 and MOCELL with the SBX operator. Also, in 57% of the cases there was not statistical significance between the TSX and the SBX regarding the *Epsilon metric.* We should highlight that the SBX in none of the examined test functions outperformed the TSX operator regarding the *Epsilon metric*. Please note that the boxplots (Table 7.2) provide graphical representation of the relevant results for the three metrics examined, namely *HV*, *IGD* and *Epsilon*. Fig.7.3-7.8 provide trade-off fronts over 100 runs for DTLZ1, 3 and 6 test problems using MOEAs with the TSX operator and MOEAs with classical SBX operator. Finally Fig. 7.9 presents the evolutionary process of HV metric for the DTLZ2 test function for the configuration of the NSGAII with the TSX operator and the SBX operator respectively.

Figure 7.10: Hypervolume
TSX performance compared with SBX

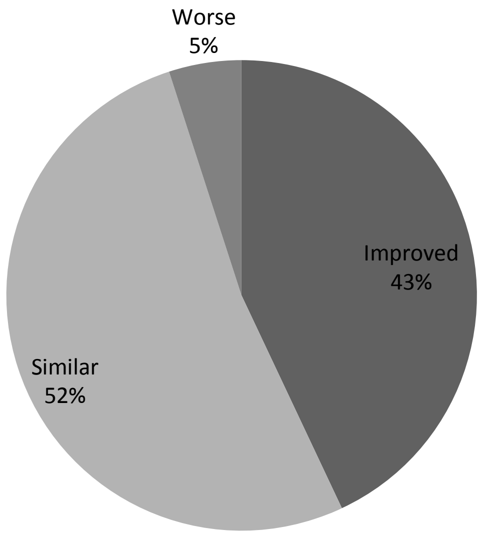


Figure 7.11: IGD
TSX performance compared with SBX

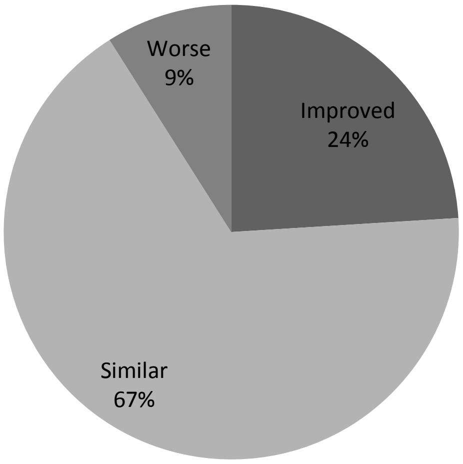
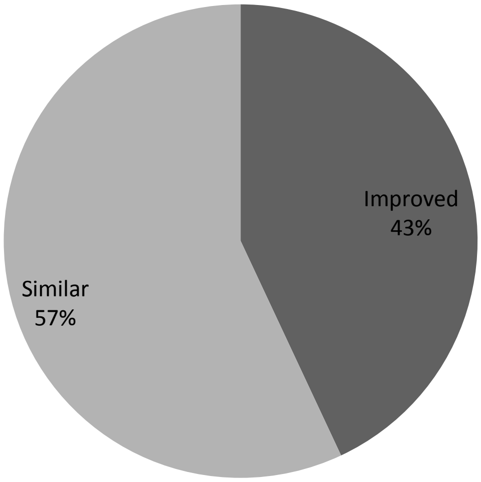


Figure 7.12: Epsilon
TSX performance compared with SBX



Based on the analysis of the relevant experimental results for the DTLZ1-7 set of test functions, the proposed TSX operator outperforms with confidence the performance of classical SBX when applied to NSGAII, SPEA2 and MOCELL respectively for all three performance metrics. Please, notice that the comparison of the TSX and the SBX operator was carried out on the same terms as we used higher number of nominal fitness function evaluations for the configuration of the algorithm with the SBX (*200,000 nominal evaluations*) compared to the configuration of the algorithm with the TSX (*76,805 nominal evaluations*) to count for the extra fitness functions evaluations introduced by the proposed methodology. For more details please refer to section 4.

By using different numbers of nominal fitness evaluations for the configuration of the algorithm with the TSX operator and the configuration of the algorithm with the SBX operator, in reality we secured equal number of real fitness functions evaluations (*200,000*) for the one and the other configuration of the algorithm. However, as expected the configuration of the algorithm with the SBX will require a higher number of generational cycles in order to execute the same number of evaluations compared to the configuration of the algorithm with the TSX operator, as each generational cycle of the algorithm with the TSX operator involves the additional fitness functions evaluations introduced by the proposed methodology.

As a result of the higher number of generational cycles for the configuration of the algorithm with the SBX, for the same number of real fitness functions evaluations (*200,000*) we should expect a computational overhead for the configuration of the algorithm with the SBX operator compared to the configuration of the algorithm with the TSX operator. Indeed, as shown in Table 7.4 that provides the average computational times for the configuration of the algorithm with the TSX and SBX respectively, there is an average computational overhead for the configuration of the algorithm with the SBX of 56.66% compared to the configuration of the algorithm with the TSX operator for the same number of real fitness functions evaluations.

Table 7.4. MEAN TOTAL CPU TIMES (in seconds) FOR THE TEST PROBLEMS FOR 200,000 (REAL) FUNCTIONS EVALUATIONS, FOR 100 INDEPENDENT RUNS

Problem	NSGAII		
	TSX	SBX	Overhead of SBX
DTLZ1	6.365e+00	6.583e+00	3.42%
DTLZ2	5.507e+00	8.315e+00	50.99%
DTLZ3	6.162e+00	8.206e+00	33.17%
DTLZ4	7.816e+00	8.003e+00	2.39%
DTLZ5	5.678e+00	8.081e+00	42.32%
DTLZ6	5.398e+00	5.513e+00	2.13%
DTLZ7	7.410e+00	9.937e+00	34.10%
		Average:	**24.07%**

Problem	SPEA2		
	TSX	SBX	Overhead of SBX
DTLZ1	28.159e+00	63.914e+00	126.98%
DTLZ2	36.693e+00	76.550e+00	108.62%
DTLZ3	31.091e+00	65.785e+00	111.59%
DTLZ4	40.358e+00	72.135e+00	78.74%
DTLZ5	28.237e+00	62.759e+00	122.26%
DTLZ6	36.629e+00	86.909e+00	137.27%
DTLZ7	41.855e+00	88.531e+00	111.52%
		Average:	**113.85%**

Problem	MOCELL		
	TSX	SBX	Overhead of SBX
DTLZ1	9.890e+00	12.340e+00	24.77%
DTLZ2	13.182e+00	19.251e+00	46.04%
DTLZ3	11.669e+00	12.683e+00	8.69%
DTLZ4	16.583e+00	16.916e+00	2.01%
DTLZ5	9.875e+00	14.196e+00	43.76%
DTLZ6	8.502e+00	12.527e+00	47.34%
DTLZ7	13.338e+00	20.264e+00	51.93%
		Average:	**32.07%**

To recap, the proposed methodology when compared to the conventional SBX operator for the same number of real fitness functions evaluations manages to generate better results with confidence for all examined performance metrics, namely HV, IGD and Epsilon indicator. On top of that the TSX operator requires a lower number of generational cycles for executing the same number of real fitness functions evaluations compared to the classical SBX operator. As a result of the aforementioned discrepancy in generational cycles between the two methodologies, we found out that the SBX requires an average computational overhead of 56.66% for executing the same number of real fitness function evaluations compared to the TSX operator. Last but not least, the proposed TSX operator in reality consist an improvement of the already good SBX operator, reinforced with an efficient fitness function evaluation mechanism. We believe that the resemblance of the proposed TSX with the well-known SBX operator will facilitate and encourage other researchers to adopt the proposed methodology and experiment with it in their areas of expertise.

7.6.2 Experimental results of the TSX operator against the SBX with the assistance of DTLZ test instances with 5 objectives

According to the Deb and Saxena [58] the vast majority of the studies in the field of MOEAs examine problems having less than 5 objectives. Also, according to the same research paper, studies using more than 10 objectives often employed a single-objective optimization method by converting the multiobjective optimization problem into single objective.

Below, we examine the efficacy of the proposed crossover operator in handling problems having 5 objectives. For that reason, we utilize the DTLZ test functions family with 5 objectives. The performance of the proposed TSX operator is assessed in comparison with the Simulated Binary Crossover (SBX) operator with the assistance of the NSGAII for the solution of the DTLZ1-7 set of test functions, with 5 objectives.

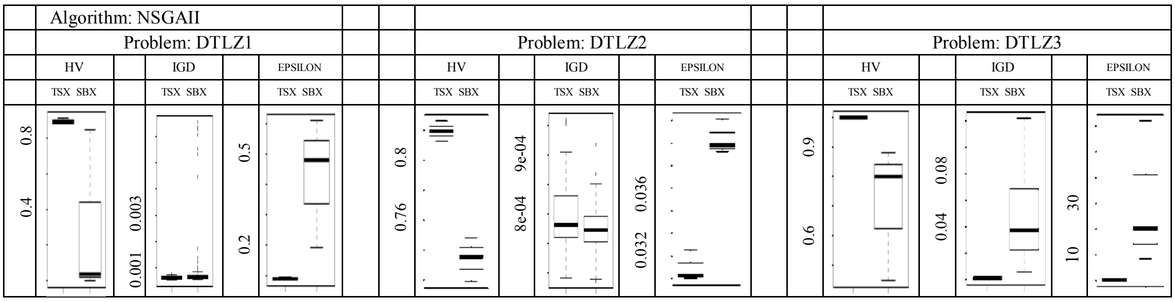
Table 7.5. Mean, Std, Median and Iqr for Hv, IGD and Epsilon: TSX vs SBX - DTLZ with 5 objectives

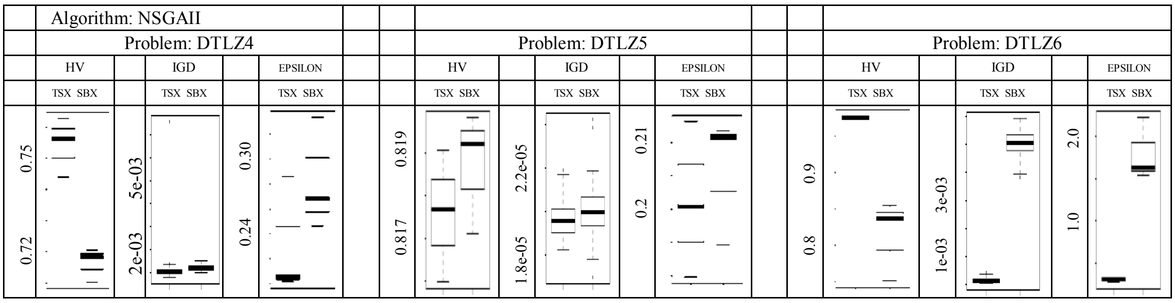
Algorithm: NSGAII	Problem: DTLZ1		Problem: DTLZ2		Problem: DTLZ3	
	TSX	SBX	TSX	SBX	TSX	SBX
HV. Mean and Std	8.97e-01$_{1.1e-02}$	2.94e-01$_{3.9e-01}$	8.19e-01$_{5.5e-03}$	7.34e-01$_{1.2e-02}$	9.99e-01$_{6.8e-04}$	7.09e-01$_{1.9e-01}$
HV. Median and IQR	8.90e-01$_{2.4e-02}$	3.82e-02$_{8.4e-01}$	8.19e-01$_{1.3e-02}$	7.36e-01$_{2.9e-02}$	9.99e-01$_{1.6e-03}$	8.00e-01$_{4.3e-01}$
IGD. Mean and Std	6.26e-04$_{4.2e-05}$	1.04e-03$_{1.3e-03}$	7.97e-04$_{5.2e-05}$	7.74e-04$_{3.6e-05}$	1.27e-03$_{7.6e-05}$	4.68e-02$_{2.8e-02}$
IGD. Median and IQR	6.24e-04$_{5.8e-05}$	6.40e-04$_{1.2e-04}$	7.82e-04$_{7.0e-05}$	7.73e-04$_{4.3e-05}$	1.26e-03$_{9.7e-05}$	3.76e-02$_{4.6e-02}$
EPSILON. Mean and Std	8.99e-02$_{2.8e-03}$	4.28e-01$_{1.8e-01}$	2.29e-01$_{1.1e-02}$	3.44e-01$_{1.2e-02}$	3.39e-01$_{6.7e-02}$	3.02e+01$_{2.3+01}$
EPSILON. Median and IQR	8.89e-02$_{6.6e-03}$	4.80e-01$_{4.2e-01}$	2.23e-01$_{2.5e-02}$	3.38e-01$_{2.8e-02}$	3.78e-01$_{1.5e-01}$	2.01e+01$_{5.4+01}$

Algorithm: NSGAII	Problem: DTLZ4		Problem: DTLZ5		Problem: DTLZ6	
	TSX	SBX	TSX	SBX	TSX	SBX
HV. Mean and Std	7.54e-01$_{7.8e-03}$	7.16e-01$_{4.3e-03}$	8.18e-01$_{1.3e-03}$	8.19e-01$_{1.2e-03}$	9.76e-01$_{1.2e-03}$	8.14e-01$_{4.5e-02}$
HV. Median and IQR	7.56e-01$_{1.9e-02}$	7.18e-01$_{1.0e-02}$	8.18e-01$_{3.1e-03}$	8.19e-01$_{2.7e-03}$	9.76e-01$_{2.9e-03}$	8.36e-01$_{1.0e-01}$
IGD. Mean and Std	1.17e-03$_{9.2e-04}$	1.19e-03$_{1.1e-04}$	1.97e-05$_{8.5e-07}$	2.00e-05$_{1.1e-06}$	1.54e-04$_{8.5e-05}$	5.03e-03$_{4.6e-04}$
IGD. Median and IQR	1.03e-03$_{1.6e-04}$	1.17e-03$_{1.6e-04}$	1.96e-05$_{1.1e-06}$	2.00e-05$_{1.3e-06}$	1.38e-04$_{1.2e-04}$	5.06e-03$_{5.7e-04}$
EPSILON. Mean and Std	2.31e-01$_{3.9e-02}$	2.83e-01$_{3.7e-02}$	2.01e-01$_{9.6e-03}$	2.06e-01$_{7.8e-03}$	2.99e-01$_{2.1e-02}$	1.80e+00$_{3.0e-01}$
EPSILON. Median and IQR	2.06e-01$_{8.5e-02}$	2.68e-01$_{8.7e-02}$	2.00e-01$_{2.3e-02}$	2.11e-01$_{1.7e-02}$	3.07e-01$_{4.9e-02}$	1.63e+00$_{6.8e-01}$

Algorithm: NSGAII	Problem: DTLZ7	
	TSX	SBX
HV. Mean and Std	2.87e-01$_{6.0e-04}$	2.85e-01$_{3.5e-03}$
HV. Median and IQR	2.87e-01$_{1.2e-03}$	2.85e-01$_{7.1e-03}$
IGD. Mean and Std	2.18e-03$_{1.1e-04}$	2.30e-03$_{2.0e-04}$
IGD. Median and IQR	2.18e-03$_{1.6e-04}$	2.28e-03$_{2.2e-04}$
EPSILON. Mean and Std	1.50e-01$_{1.4e-03}$	1.55e-01$_{1.9e-02}$
EPSILON. Median and IQR	1.50e-01$_{2.8e-03}$	1.55e-01$_{3.8e-02}$

Table 7.6. Boxplots for Hv, IGD and Epsilon: TSX vs SBX - DTLZ with 5 objectives





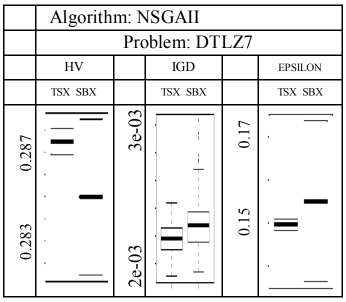


Table 7.7. Wilcoxon Test for Hv, IGD and Epsilon: TSX vs SBX - DTLZ with 5 objectives

Problem		DTLZ1	DTLZ2	DTLZ3	DTLZ4	DTLZ5	DTLZ6	DTLZ7
		NSGAII with SBX	NSGAII with SBX	NSGAII with SBX	NSGAII with SBX	NSGAII with SBX	NSGAII with SBX	NSGAII with SBX
HV. Mean and Std	NSGAII with TSX	↑	↑	↑	↑	↓	↑	–
HV. Median and IQR		↑	↑	↑	↑	↓	↑	–
IGD. Mean and Std		–	–	↑	–	↑	↑	↑
IGD. Median and IQR		–	–	↑	–	↑	↑	↑
EPSILON. Mean and Std		↑	↑	↑	↑	↑	↑	–
EPSILON. Median and IQR		↑	↑	↑	↑	↑	↑	–

As we explained in section 3 that we presented the TSX operator, *whenever, there is not a clear winner i.e. strong or weak dominance, between the* $c_L^{(1)}$*and* $c_R^{(1)}$ *, or respectively between the* $c_L^{(2)}$*and* $c_R^{(2)}$ *the generation of a random number allows the random choice of one of the two alternative child solutions.*

Someone may argue that as the number of objectives increases the random choice of one of the two alternative child solutions will apply more frequently, as it becomes more likely the solutions to be non-dominated to each other. Clearly, as the number of random choices of one of the two alternative child solutions increases due to the non-domination between the solutions, the deterministic nature of the selection process of the TSX operator becomes partially probabilistic. We define the probabilistic indicator of TSX operator as the ratio between the random choices of one of the two alternative child solutions and the total evaluations,

$$Probabilistic\ indicator\ of\ TSX\ operator = \frac{Random\ choices\ (i.e.\ child\ \ solutions\ non{-}dominated\ )}{Total\ evaluations}$$

Below, we provide the probabilistic indicator of the TSX operator for the DTLZ1-7 set of test functions for 3 and 5 objectives respectively.

Table 7.8. Probabilistic Indicator of TSX operator with 3 objectives

DTLZ1-7 set of test functions with **3 objectives**	DTLZ1	DTLZ2	DTLZ3	DTLZ4	DTLZ5	DTLZ6	DTLZ7
Probabilistic indicator of TSX operator	22.5%	9.30%	10.2%	9.7%	8.2%	6.8%	12.7%

Table 7.8 shows the probabilistic indicator of TSX operator for the DTLZ family of test functions with 3 objectives. This configuration corresponds to the results of section 6.1. Although, a considerable part that varies between 8.2% for the DTLZ5 and up to 22.5% for the DTLZ1 of the selection process of the TSX operator is probabilistic, the remaining deterministic part (i.e. strong or weak domination between the child solutions) of the TSX operator is adequate to generate superior results than the SBX operator as shown in section 6.1.

Table 7.9. Probabilistic Indicator of TSX operator with 5 objectives

DTLZ1-7 set of test functions with **5 objectives**	DTLZ1	DTLZ2	DTLZ3	DTLZ4	DTLZ5	DTLZ6	DTLZ7
Probabilistic indicator of TSX operator	28.7%	12.50%	12.4%	11.9%	11.4%	9.7%	17.1%

Table 7.9 shows the probabilistic indicator of TSX operator for the DTLZ family of test functions with 5 objectives. This configuration corresponds to the results of section 7.6.2. As expected the higher number of objectives, leads to an increase of the probabilistic indicator of the TSX operator. However, despite the higher level of the

probabilistic indicator for the configuration of the DTLZ family of test functions with 5 objectives, the TSX operator still generates superior quality results than the SBX operator as shown by the Tables 7.5, 7.6 and 7.7 respectively. This can be attributed to the ability of the TSX operator to locate higher fitness regions in complicated search spaces, thanks to its exploratory mechanism.

Below, we analyze the results obtained by applying the proposed Two Stage Crossover (TSX) and the Simulated Binary Crossover (SBX), to the NSGAII for solving the DTLZ 1-7 set of test functions with 5 objectives. Please notice that for the configuration of the algorithm with the TSX operator and the SBX operator alike, we secured equal number of real fitness functions evaluations (*200,000*).

Examining the results of the first indicator, the *HV*, we notice that the TSX operator performs better than the classical SBX operator when is applied to the NSGAII for solving the DTLZ 1-7 set of test functions with 5 objectives. Figure 13 reveals the relevant results for the performance of TSX operator regarding the *HV* indicator. The Wilcoxon rank-sum test (see Table 7.7) validates that the observed difference in TSX and SBX performance is statistically significant with 95% confidence. In 72% of the cases the TSX yields better results with confidence than the conventional configuration of the NSGAII with the SBX operator. Also, in 14% of the cases there was not statistical significance between the TSX and SBX. Finally, in 14% of the cases the SBX outperformed the TSX operator regarding the *HV metric*.

Regarding the *IGD indicator* (see Fig. 7.14) in 57% of the cases the TSX yields better results with confidence than the conventional configuration of the NSGAII with the SBX operator. In 43% of the cases there was not statistical significance between the TSX and the SBX regarding the *IGD indicator*.

Figure 15 reveals the relevant results for the performance of TSX regarding the *Epsilon* indicator. The Wilcoxon rank-sum test helps us to identify that in 86% of the cases the TSX yields better results with confidence than the conventional configuration of the NSGAII with the SBX operator. Also, in 14% of the cases there was not statistical significance between the TSX and the SBX regarding the *Epsilon metric.* We should highlight that the SBX in none of the examined test functions outperformed the TSX operator regarding the *Epsilon metric*.

Figure 7.13: Hypervolume
TSX performance compared to SBX
DTLZ with 5 objectives

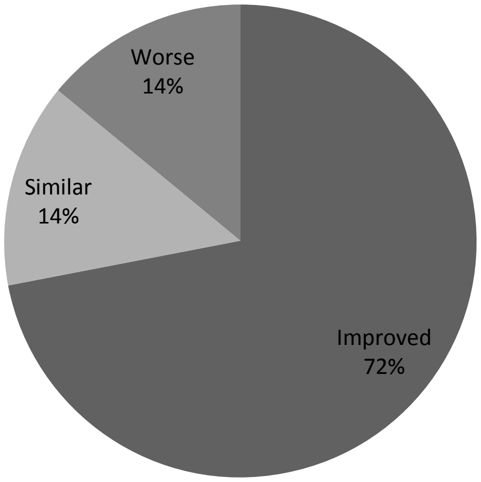


Figure 7.14: IGD
TSX performance compared to SBX
DTLZ with 5 objectives

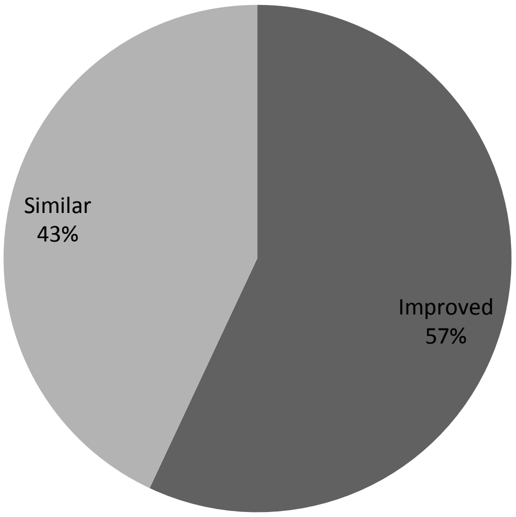
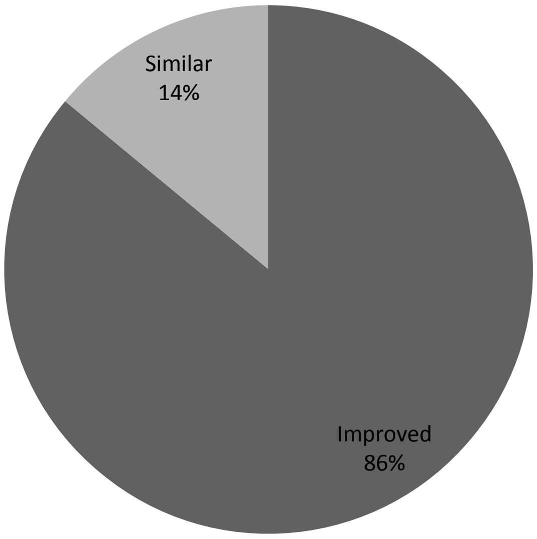


**Figure 7.15: Epsilon
TSX performance compared to SBX
DTLZ with 5 objectives**



7.6.3 Experimental results of the TSX operator against the Differential Evolution (DE) with the assistance of DTLZ test suite with 3 objectives

Below, we examine the efficacy of the proposed TSX operator when compared to the Differential Evolution (DE) [138] with the assistance of the Deb-Theile-Laumanns-Zitzler (DTLZ) [60] family of test functions with 3 objectives. In particular, the performance of the proposed TSX operator is assessed in comparison with the Differential Evolution (DE) operator with the assistance of the Multiobjective Evolutionary Algorithm based on Decomposition (MOEA/D) for the solution of the DTLZ1-7 set of test functions. Table 7.10, 7.11 and 7.12 reveal the relevant results.

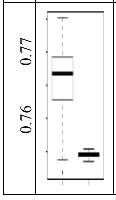
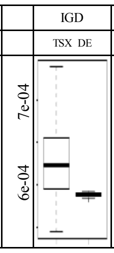
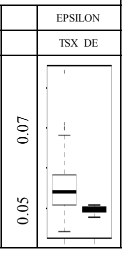
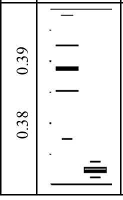
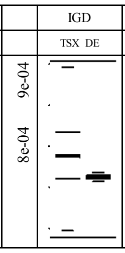
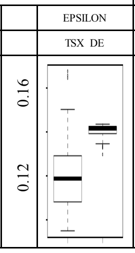
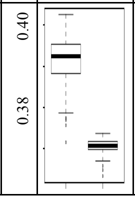
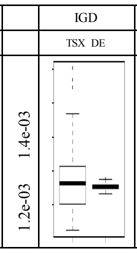
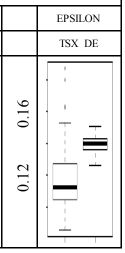
Table 7.10. Mean, Std, Median and Iqr for Hv, IGD and Epsilon – TSX vs DE

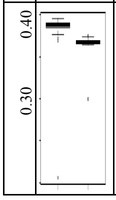
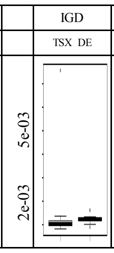
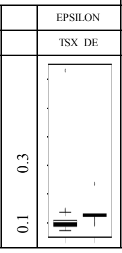
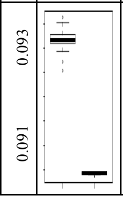
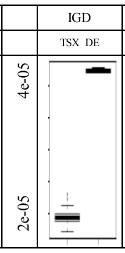
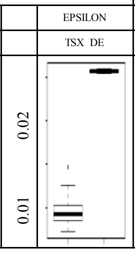
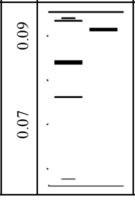
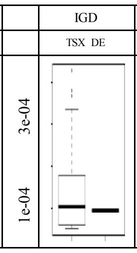
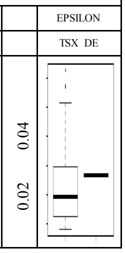
3 objectives	Problem: DTLZ1		Problem: DTLZ2		Problem: DTLZ3	
	MOEAD+TSX	MOEAD+DE	MOEAD+TSX	MOEAD+DE	MOEAD+TSX	MOEAD+DE
HV. Mean and Std	7.66e-01$_{5.1e-03}$	7.54e-01$_{3.9e-04}$	3.89e-01$_{4.9e-03}$	3.72e-01$_{5.4e-04}$	3.92e-01$_{5.9e-03}$	3.71e-01$_{1.6e-03}$
HV. Median and IQR	7.67e-01$_{6.5e-03}$	7.54e-01$_{5.2e-04}$	3.89e-01$_{7.4e-03}$	3.72e-01$_{6.9e-04}$	3.92e-01$_{7.1e-03}$	3.71e-01$_{2.0e-03}$
IGD. Mean and Std	6.27e-04$_{4.4e-05}$	5.89e-04$_{1.8e-06}$	7.92e-04$_{4.3e-05}$	7.62e-04$_{2.8e-06}$	1.27e-03$_{9.0e-05}$	1.25e-03$_{8.5e-06}$
IGD. Median and IQR	6.23e-04$_{6.1e-05}$	5.89e-04$_{2.5e-06}$	7.88e-04$_{5.6e-05}$	7.63e-04$_{4.3e-06}$	1.26e-03$_{1.1e-04}$	1.25e-03$_{1.1e-05}$
EPSILON. Mean and Std	5.49e-02$_{6.4e-03}$	4.96e-02$_{7.2e-04}$	1.20e-01$_{1.5e-02}$	1.40e-01$_{3.1e-03}$	1.17e-01$_{2.0e-02}$	1.39e-01$_{4.6e-03}$
EPSILON. Median and IQR	5.41e-02$_{7.5e-03}$	4.97e-02$_{1.2e-03}$	1.19e-01$_{2.1e-02}$	1.42e-01$_{3.3e-03}$	1.12e-01$_{2.3e-02}$	1.40e-01$_{6.8e-03}$

3 objectives	Problem: DTLZ4		Problem: DTLZ5		Problem: DTLZ6	
	MOEAD+TSX	MOEAD+DE	MOEAD+TSX	MOEAD+DE	MOEAD+TSX	MOEAD+DE
HV. Mean and Std	3.85e-01$_{1.9e-02}$	3.56e-01$_{2.6e-02}$	9.32e-02$_{1.6e-04}$	9.04e-02$_{1.5e-05}$	8.28e-02$_{9.3e-03}$	9.13e-02$_{2.3e-06}$
HV. Median and IQR	3.88e-01$_{5.7e-03}$	3.67e-01$_{3.3e-03}$	9.32e-02$_{1.8e-04}$	9.04e-02$_{1.7e-05}$	8.39e-02$_{1.7e-02}$	9.13e-02$_{3.2e-06}$
IGD. Mean and Std	1.14e-03$_{6.6e-04}$	1.29e-03$_{1.8e-04}$	1.94e-05$_{9.5e-07}$	4.23e-05$_{2.3e-07}$	1.34e-04$_{8.3e-05}$	9.58e-05$_{3.9e-08}$
IGD. Median and IQR	1.05e-03$_{2.0e-04}$	1.25e-03$_{1.3e-04}$	1.94e-05$_{1.2e-06}$	4.23e-05$_{4.2e-07}$	1.04e-04$_{1.2e-04}$	9.58e-05$_{5.8e-08}$
EPSILON. Mean and Std	1.05e-01$_{5.5e-02}$	1.44e-01$_{4.3e-02}$	9.53e-03$_{1.4e-03}$	2.57e-02$_{9.8e-05}$	2.24e-02$_{1.2e-02}$	2.66e-02$_{7.6e-06}$
EPSILON. Median and IQR	9.53e-02$_{2.0e-02}$	1.26e-01$_{8.4e-04}$	9.26e-03$_{1.8e-03}$	2.57e-02$_{1.4e-04}$	1.93e-02$_{1.7e-02}$	2.66e-02$_{5.7e-06}$

3 objectives	Problem: DTLZ7	
	MOEAD+TSX	MOEAD+DE
HV. Mean and Std	2.88e-01$_{3.3e-03}$	2.38e-01$_{6.7e-03}$
HV. Median and IQR	2.88e-01$_{5.3e-03}$	2.39e-01$_{1.5e-03}$
IGD. Mean and Std	2.18e-03$_{1.5e-04}$	6.50e-03$_{5.4e-03}$
IGD. Median and IQR	2.16e-03$_{1.8e-04}$	4.44e-03$_{3.5e-04}$
EPSILON. Mean and Std	1.38e-01$_{4.4e-02}$	3.89e-01$_{6.1e-01}$
EPSILON. Median and IQR	1.26e-01$_{4.7e-02}$	1.70e-01$_{1.1e-03}$

Table 7.11. Boxplots for Hv, IGD and Epsilon: TSX vs DE

	MOEA/D +TSX vs MOEA/D+DE																		
	Problem: DTLZ1							Problem: DTLZ2							Problem: DTLZ3				
	HV		IGD		EPSILON			HV		IGD		EPSILON			HV		IGD		EPSILON
	TSX DE		TSX DE		TSX DE			TSX DE		TSX DE		TSX DE			TSX DE		TSX DE		TSX DE
																			

	MOEA/D +TSX vs MOEA/D+DE																		
	Problem: DTLZ4							Problem: DTLZ5							Problem: DTLZ6				
	HV		IGD		EPSILON			HV		IGD		EPSILON			HV		IGD		EPSILON
	TSX DE		TSX DE		TSX DE			TSX DE		TSX DE		TSX DE			TSX DE		TSX DE		TSX DE
																			

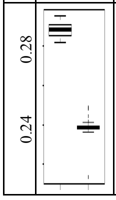
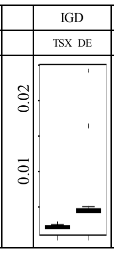
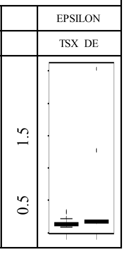
	MOEA/D +TSX vs MOEA/D+DE				
	Problem: DTLZ7				
	HV		IGD		EPSILON
	TSX DE		TSX DE		TSX DE
					

Table 7.12. Wilcoxon Test for Hv, Spread and Epsilon: TSX vs DE

Problem		DTLZ1	DTLZ2	DTLZ3	DTLZ4	DTLZ5	DTLZ6	DTLZ7
		MOEA/D with DE	MOEA/D with DE	MOEA/D with DE	MOEA/D with DE	MOEA/D with DE	MOEA/D with DE	MOEA/D with DE
HV. Mean and Std	MOEA/D with TSX	↑	↑	↑	↑	↑	↓	↑
HV. Median and IQR		↑	↑	↑	↑	↑	↓	↑
IGD. Mean and Std		↓	↓	–	↑	↑	–	↑
IGD. Median and IQR		↓	↓	–	↑	↑	–	↑
EPSILON. Mean and Std		↓	↑	↑	↑	↑	↑	↑
EPSILON. Median and IQR		↓	↑	↑	↑	↑	↑	↑

Below, we analyze the results obtained by applying the proposed Two Stage Crossover (TSX) and the Differential Evolution (DE), for solving the DTLZ 1-7 [60] set of test functions with 3 objectives. Please notice that for the configuration of the algorithm with the TSX operator and the DE alike, we secured equal number of real fitness functions evaluations (*200,000*).

Examining the results of the first indicator, the *HV*, we notice that the MOEAD+TSX operator performs better than the MOEAD+DE for solving the DTLZ 1-7 set of test functions with 3 objectives. Figure 18 reveals the relevant results for the performance of TSX operator regarding the *HV* indicator. The Wilcoxon rank-sum test (see Table 7.12) validates that the observed difference in TSX and DE performance is statistically significant with 95% confidence. In 86% of the cases the MOEAD+TSX yields better results with confidence than the MOEAD with the DE operator. Also, in 14% of the cases the DE outperformed the TSX operator regarding the *HV metric*.

Regarding the *IGD indicator* (see Fig. 7.19) in 44% of the cases the MOEAD+TSX yields better results with confidence than the MOEAD with the DE operator. In 28% of the examined cases the DE operator generated better results with confidence than the proposed methodology. Also in 28% of the cases there was not statistical significance between the TSX and the DE regarding the *IGD indicator.*

Figure 20 reveals the relevant results for the performance of TSX regarding the *Epsilon* indicator. The Wilcoxon rank-sum test helps us to identify that in 86% of the cases the TSX yields better results with confidence than the MOEAD with the DE. Also, in 14% of the cases the DE outperformed the TSX operator regarding the *Epsilon metric.*

Moreover, Fig.7.21-7.24 provide trade-off fronts for DTLZ1 and 5 test problems using the MOEA/D with the TSX operator and the MOEA/D with the DE. Finally, Fig. 7.16-7.17 present the evolutionary process of IGD and Epsilon indicator for the DTLZ5 test function for the configuration of the MOEA/D with the TSX operator and the DE respectively.

Fig. 7.16. DTLZ5: Evolution of Epsilon: MOEA/D+TSX and MOEA/D+DE

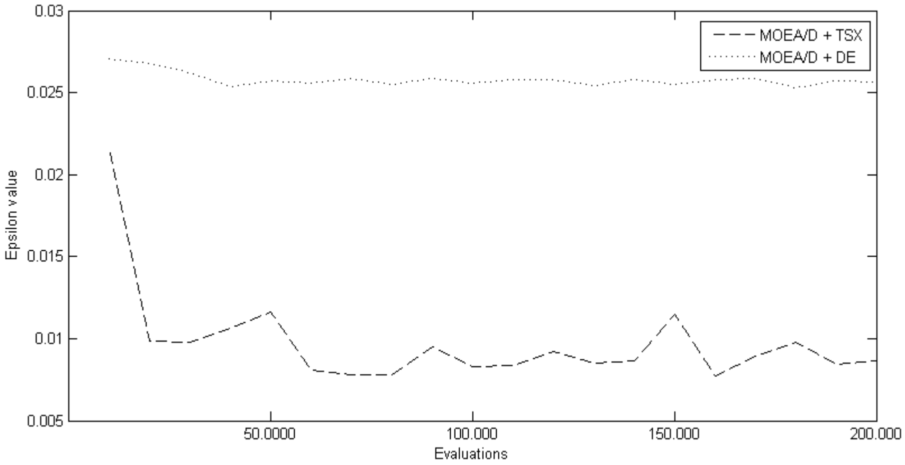


Fig. 7.17. DTLZ5: Evolution of IGD: MOEA/D+TSX and MOEA/D+DE

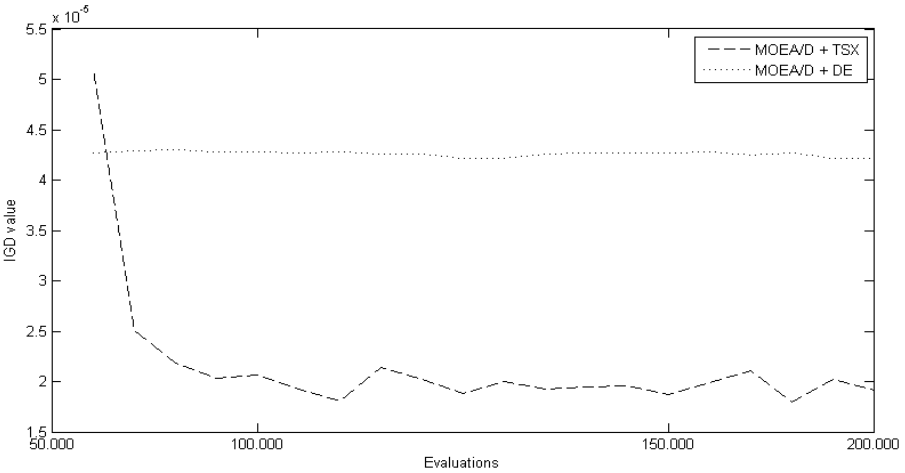


Figure 7.18: Hypervolume
TSX performance compared to DE

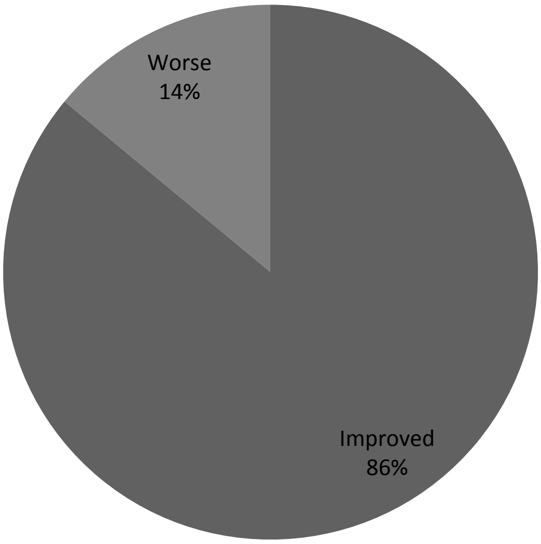


Figure 7.19: IGD
TSX performance compared to DE

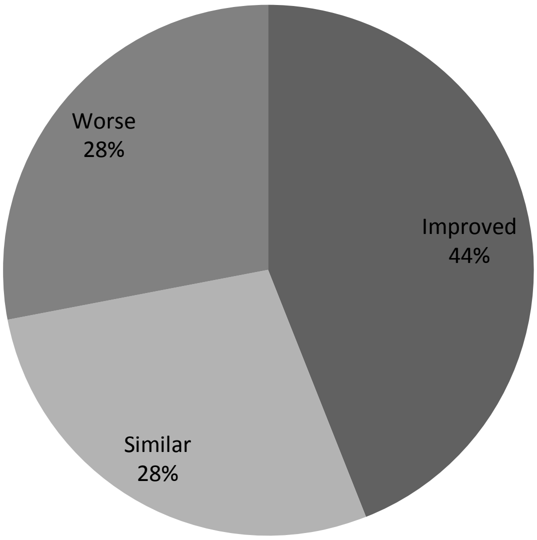
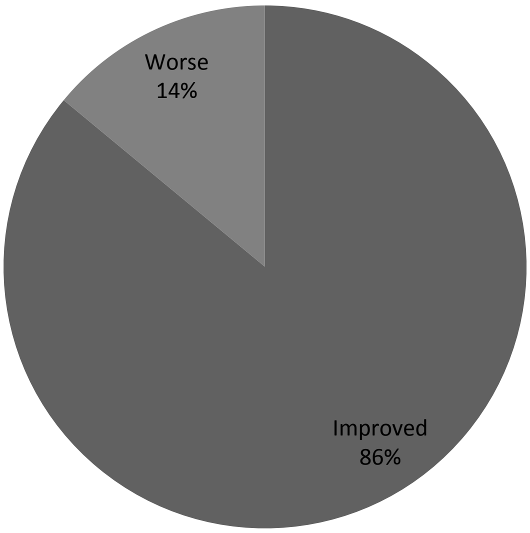


Figure 7.20: Epsilon
TSX performance compared to DE



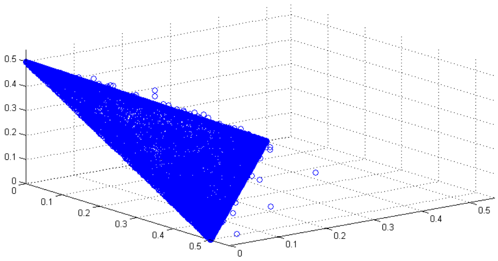


Fig. 7.21. Problem: DTLZ1. MOAE/D + TSX operator

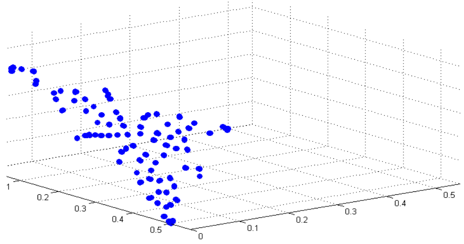


Fig. 7.22. Problem: DTLZ1. MOAE/D + DE operator

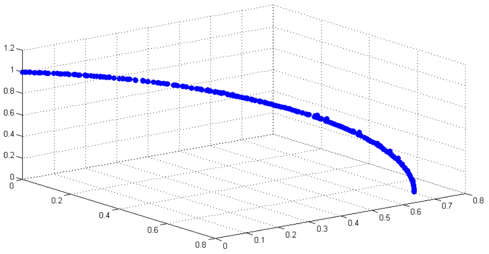


Fig. 7.23. Problem: DTLZ5. MOAE/D + TSX operator

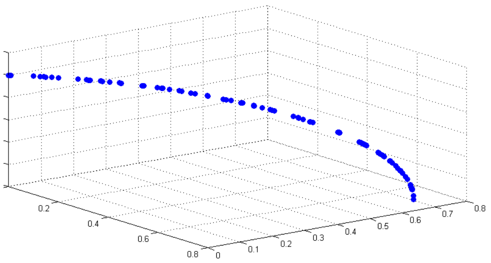


Fig. 7.24. Problem: DTLZ5. MOAE/D + DE operator

7.6.4 Experimental results of the TSX operator against the Particle Swarm Optimization (PSO) with the assistance of DTLZ test suite with 3 objectives

Below, we examine the efficacy of the proposed TSX operator when compared to the Particle Swarm Optimization (PSO) with the assistance of the Deb-Theile-Laumanns-Zitzler (DTLZ) [60] family of test functions with 3 objectives. Specifically, the performance of the NSGAII+TSX is compared with the performance of the Speed-constrained Multi-objective PSO (SMPSO) [165] for the DTLZ1-7 set of test functions. The SMPSO incorporates a constriction mechanism in order to limit the maximum velocity of particles and enhance the search capability of the algorithm.

There are numerous studies in the field that provide an experimental comparison between the SMPSO and the NSGAII, to refer some of these studies: [163], [165], [191] and [162]. According to all aforementioned studies ([163], [165], [191] and [162]) the SMPSO clearly outperforms the NSGAII based on a variety of performance metrics like the HV, Epsilon and Spread indicator for the DTLZ family of test functions. For comparison reasons in our study we make use of the same family of test functions (i.e. DTLZ1-7), however in our experiments we use a configuration of the NSGAII with the proposed TSX operator instead of the typical configuration of the NSGAII with the SBX operator. Potential gain in performance of the NSGAII configured with the TSX operator when compared with the SMPSO for the DTLZ1-7 test problems will be a clear indication of the efficacy of the proposed methodology. Table 7.13, 7.14 and 7.15 reveal the relevant results.

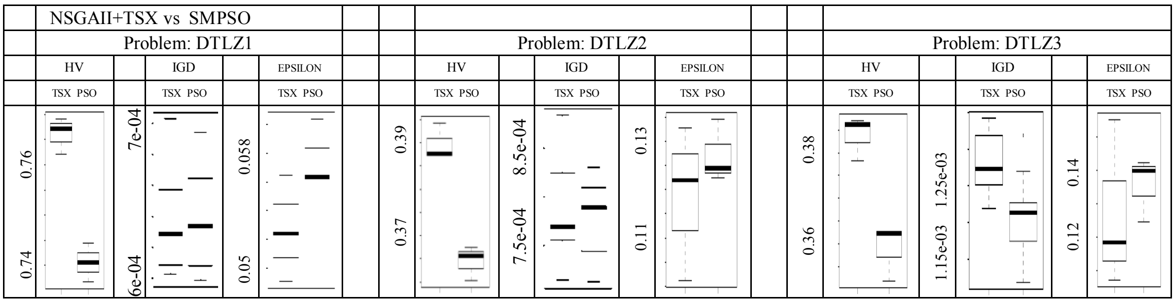
Table 7.13. Mean, Std, Median and Iqr for Hv, IGD and Epsilon – TSX vs PSO

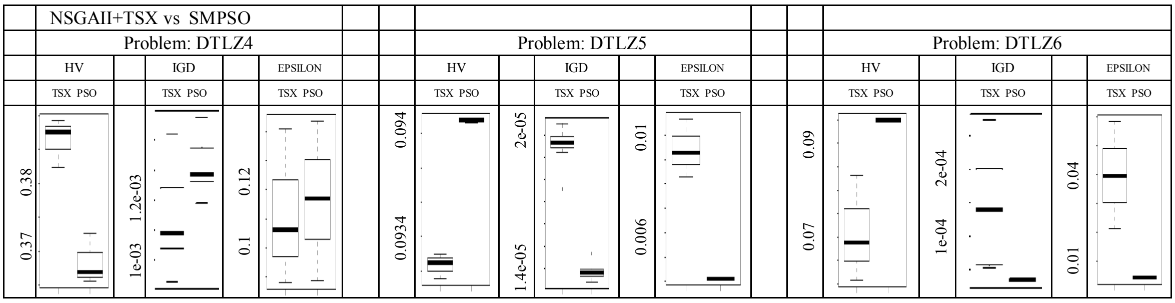
3 objectives	Problem: DTLZ1		Problem: DTLZ2		Problem: DTLZ3	
	NSGAII+TSX	SMPSO	NSGAII+TSX	SMPSO	NSGAII+TSX	SMPSO
HV. Mean and Std	7.66e-01$_{3.0e-03}$	7.41e-01$_{3.1e-03}$	3.90e-01$_{3.3e-03}$	3.64e-01$_{3.0e-03}$	3.84e-01$_{3.9e-03}$	3.59e-01$_{5.1e-03}$
HV. Median and IQR	7.67e-01$_{7.0e-03}$	7.41e-01$_{7.7e-03}$	3.88e-01$_{7.1e-03}$	3.66e-01$_{7.1e-03}$	3.86e-01$_{8.8e-03}$	3.62e-01$_{1.1e-02}$
IGD. Mean and Std	6.34e-04$_{3.0e-05}$	6.35e-04$_{3.1e-05}$	7.89e-04$_{5.3e-05}$	7.78e-04$_{3.8e-05}$	1.28e-03$_{3.7e-05}$	1.21e-03$_{5.5e-05}$
IGD. Median and IQR	6.29e-04$_{4.9e-05}$	6.33e-04$_{5.7e-05}$	7.69e-04$_{7.0e-05}$	7.89e-04$_{6.5e-05}$	1.27e-03$_{6.7e-05}$	1.21e-03$_{5.2e-05}$
EPSILON. Mean and Std	5.26e-02$_{3.0e-03}$	5.77e-02$_{1.9e-03}$	1.19e-01$_{1.3e-02}$	1.27e-01$_{5.3e-03}$	1.27e-01$_{2.1e-02}$	1.36e-01$_{7.8e-03}$
EPSILON. Median and IQR	5.24e-02$_{7.4e-03}$	5.64e-02$_{4.2e-03}$	1.22e-01$_{3.2e-02}$	1.24e-01$_{1.2e-02}$	1.18e-01$_{4.8e-02}$	1.40e-01$_{1.8e-02}$

3 objectives	Problem: DTLZ4		Problem: DTLZ5		Problem: DTLZ6	
	NSGAII+TSX	SMPSO	NSGAII+TSX	SMPSO	NSGAII+TSX	SMPSO
HV. Mean and Std	3.86e-01$_{2.9e-03}$	3.68e-01$_{3.1e-03}$	9.32e-02$_{6.1e-05}$	9.41e-02$_{7.3e-06}$	7.09e-02$_{9.3e-03}$	9.49e-02$_{7.0e-06}$
HV. Median and IQR	3.88e-01$_{6.9e-03}$	3.67e-01$_{7.2e-03}$	9.32e-02$_{1.5e-04}$	9.41e-02$_{1.6e-05}$	6.88e-02$_{2.2e-02}$	9.49e-02$_{1.6e-05}$
IGD. Mean and Std	1.14e-03$_{1.4e-04}$	1.31e-03$_{7.8e-05}$	1.95e-05$_{7.6e-07}$	1.38e-05$_{3.5e-07}$	1.46e-04$_{7.7e-05}$	3.46e-05$_{1.0e-06}$
IGD. Median and IQR	1.09e-03$_{2.0e-04}$	1.29e-03$_{1.1e-04}$	1.97e-05$_{5.1e-07}$	1.38e-05$_{3.2e-07}$	1.40e-04$_{1.4e-04}$	3.48e-05$_{1.8e-06}$
EPSILON. Mean and Std	1.12e-01$_{2.2e-02}$	1.16e-01$_{2.3e-02}$	9.40e-03$_{9.7e-04}$	4.13e-03$_{2.1e-05}$	3.97e-02$_{1.5e-02}$	4.07e-03$_{1.3e-04}$
EPSILON. Median and IQR	1.06e-01$_{5.3e-02}$	1.17e-01$_{5.5e-02}$	9.27e-03$_{2.4e-03}$	4.12e-03$_{4.6e-05}$	3.95e-02$_{3.7e-02}$	4.15e-03$_{2.7e-04}$

3 objectives	Problem: DTLZ7	
	NSGAII+TSX	SMPSO
HV. Mean and Std	2.86e-01$_{2.1e-03}$	2.76e-01$_{2.6e-03}$
HV. Median and IQR	2.86e-01$_{5.1e-03}$	2.77e-01$_{5.9e-03}$
IGD. Mean and Std	2.27e-03$_{2.0e-04}$	2.77e-03$_{2.0e-04}$
IGD. Median and IQR	2.21e-03$_{1.7e-04}$	2.71e-03$_{2.5e-04}$
EPSILON. Mean and Std	1.33e-01$_{3.5e-02}$	2.00e-01$_{2.0e-02}$
EPSILON. Median and IQR	1.15e-01$_{8.0e-02}$	2.00e-01$_{4.9e-02}$

Table 7.14. Boxplots for Hv, IGD and Epsilon: TSX vs PSO





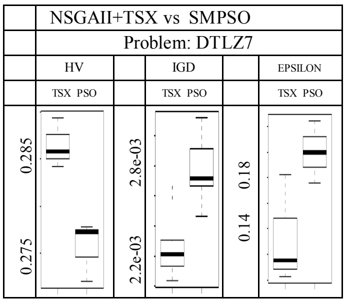


Table 7.15. Wilcoxon Test for Hv, IGD and Epsilon: TSX vs PSO

Problem		DTLZ1	DTLZ2	DTLZ3	DTLZ4	DTLZ5	DTLZ6	DTLZ7
		SMPSO	SMPSO	SMPSO	SMPSO	SMPSO	SMPSO	SMPSO
HV. Mean and Std	NSGAII with TSX	↑	↑	↑	↑	↓	↓	↑
HV. Median and IQR		↑	↑	↑	↑	↓	↓	↑
IGD. Mean and Std		–	↑	↓	↑	↓	↓	↑
IGD. Median and IQR		–	↑	↓	↑	↓	↓	↑
EPSILON. Mean and Std		↑	–	↑	↑	↓	↓	↑
EPSILON. Median and IQR		↑	–	↑	↑	↓	↓	↑

Below, we analyze the results obtained by applying the proposed Two Stage Crossover (TSX) and the Particle Swarm Optimization (PSO), for solving the DTLZ 1-7 [60] set of test functions with 3 objectives. Please notice that for the configuration of the algorithm with the TSX operator and the PSO alike, we secured equal number of real fitness functions evaluations (*200,000*).

Examining the results of the first indicator, the *HV*, we notice that the NSGAII+TSX operator performs better than the SMPSO for solving the DTLZ 1-7 set of test functions with 3 objectives. Figure 25 reveals the relevant results for the performance of TSX operator regarding the *HV* indicator. The Wilcoxon rank-sum test (see Table 7.15) validates that the observed difference in TSX and PSO performance is statistically significant with 95% confidence. In 71% of the cases the TSX yields better results with confidence than the PSO. Also, in 29% of the cases the PSO outperformed the TSX operator regarding the *HV metric*.

Regarding the *IGD indicator* (see Fig. 7.26) in 43% of the cases the TSX yields better results with confidence than the PSO. Also, in 43% of the examined cases the PSO operator generated better results with confidence than the proposed methodology. Finally, in 14% of the cases there was not statistical significance between the TSX and the PSO regarding the *IGD indicator.*

Figure 7.27 reveals the relevant results for the performance of TSX regarding the *Epsilon* indicator. The Wilcoxon rank-sum test helps us to identify that in 57% of the cases the TSX yields better results with confidence than the PSO. Also, in 14% of the cases there was not statistical significance between the TSX and the PSO regarding the *Epsilon metric.* Finally, in 29% of the examined cases the PSO operator generated better results with confidence than the proposed methodology regarding the *Epsilon metric.*

From the relevant results we conclude that the NSGAII + TSX outperforms the SMPSO in HV and Epsilon indicator, while the two algorithms perform similarly regarding the IGD indicator. Please notice that the relevant results in these two performance metrics (i.e. HV and Epsilon indicator) are contrary to the results of the aforementioned [163], [165], [191] and [162] studies and can be attributed to the efficacy of the proposed TSX operator when compared to the SBX operator, as everything else about the configuration of the NSGAII algorithm when compared with the SMPSO remains the same.

Figure 7.25: Hypervolume
TSX performance compared to PSO

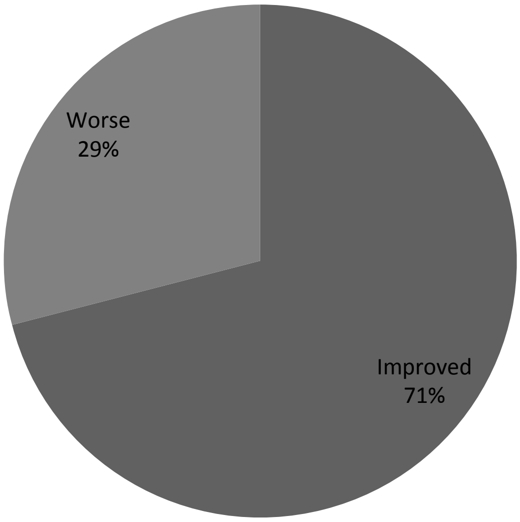


Figure 7.26: IGD
TSX performance compared to PSO

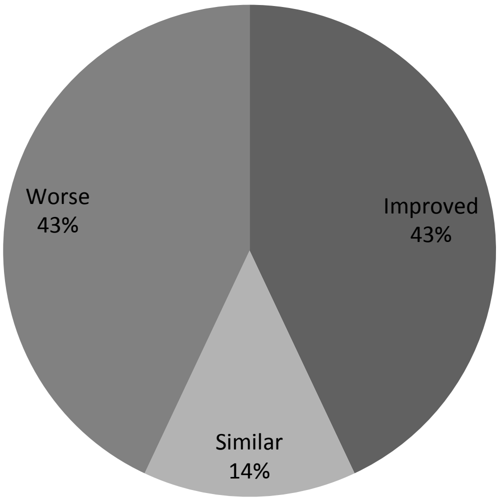
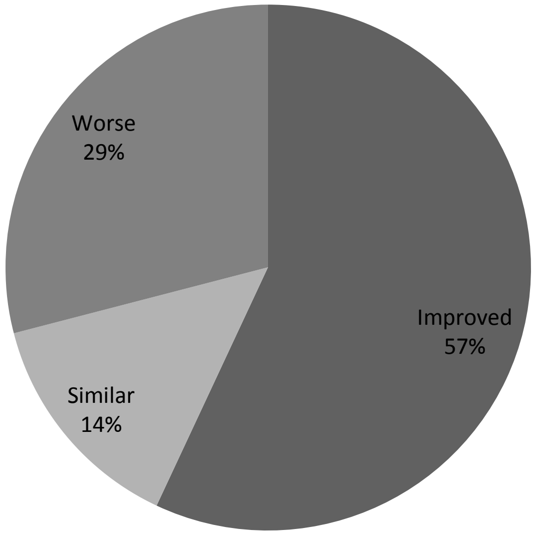


Figure 7.27: Epsilon
TSX performance compared to PSO



7.7 Conclusions

In this chapter, we have developed a new Two Stage Crossover (TSX) operator for real coded evolutionary algorithms. The proposed crossover operator incorporates a fitness function evaluation mechanism that allows the evaluation of the corresponding fitness for the left hand side region and the right hand side region of the parent solution. The selection between the two alternative child solutions is done with the assistance of the Pareto optimality conditions. Thanks to the TSX operator the algorithm is able to move fast towards the higher fitness regions of the search landscape and discover near optimal solutions.

In the experimental part of the study we analyzed the results obtained by applying the proposed Two Stage Crossover (TSX), the Simulated Binary Crossover (SBX), the Differential Evolution (DE) and the Particle Swarm Optimization (PSO) to a number of algorithms for solving the DTLZ 1-7 set of test functions. The assessment of the performance of the proposed crossover operator is done with the assistance of three well known performance indicators, namely the *Hypervolume*, *IGD* and *Epsilon* indicator for equal number of real function evaluations.

In particular in section 6.1 we applied the TSX operator and the SBX operator respectively to the NSGAII, SPEA2 and MOCELL for solving the DTLZ 1-7 set of test functions with 3 objectives. The experimental results showed us that the proposed TSX operator yields better results with confidence than the conventional configuration of the NSGAII, SPEA2 and MOCELL with the SBX operator for all three performance metrics. Also, as shown in Table 7.4 the configuration of the algorithm with the SBX requires an average computational overhead 56.66% compared to the configuration of the algorithm with the TSX operator for the same number of real fitness functions evaluations.

In section 7.6.2 we examined the ability of the proposed TSX operator in handling problems with more than 3 objectives. For that reason, we applied the TSX operator and the SBX operator respectively to the NSGAII for solving the DTLZ 1-7 set of test functions with 5 objectives. The proposed TSX operator generated better results with confidence than the SBX operator for all three performance metrics for solving the DTLZ 1-7 test suite with 5 objectives.

In section 7.6.3 and 7.6.4 we compared the performance of the proposed TSX operator against other popular and well-established reproduction techniques. In particular in section 6.3 we compared the performance of the proposed TSX operator against the Differential Evolution (DE) with the assistance of the MOEA/D for the same number of real function evaluations. The proposed TSX operator generated better results with confidence than the DE for all three performance metrics for solving the DTLZ 1-7 set of test functions with 3 objectives. In section 6.4 we compared the performance of the proposed TSX operator against the Particle Swarm Optimization (PSO) with the assistance of NSGAII and SMPSO respectively for the same number of real function evaluations. The proposed TSX operator generated better results with confidence than the PSO for the HV and Epsilon performance metrics and not worse results for the IGD indicator, for solving the DTLZ 1-7 set of test functions with 3 objectives.

Given the efficacy of the proposed TSX operator as it is demonstrated by the experimental part of the study, we hope that other researchers will adopt the proposed methodology and experiment with it in their areas of expertise.

In our future work, we will attempt to develop a technique that will replace the random boolean value r_b of TSX operator with a variable that will be updated at run-time according to the performance of the algorithm in a number of metrics such as the hypervolume or epsilon indicator.

Chapter 8

A new Efficiently Encoded Multiobjective Algorithm for the Solution of the Constrained Portfolio Optimization Problem

8.1 Introduction

The portfolio optimization problem aims to find the optimal allocation of a limited capital among a finite set of risky assets by simultaneously maximizing the expected return of the portfolio and minimizing its associated risk. The portfolio optimization belongs to the category of the combinatorial optimization problems as it has a highly complex search space due to the large number of selectable assets available, especially for big instances of the problem [46], [142], [158], [155]. The portfolio optimization problem as introduced by Markowitz [151] can be solved by quadratic programming. However, if we apply some real-world nonlinear constraints such as cardinality constraints, which impose a lower and upper limit to the number of assets held in the portfolio the problem becomes NP-hard.

The additional nonlinear constraints imposed to the problem make it difficult to be solved with exact methods [149]. In the last decade several metaheuristic optimization techniques have been developed to address the challenges imposed by the constrained portfolio optimization problem. Due to the intrinsic multiobjective nature of the problem, multiobjective approaches, particularly multiobjective evolutionary algorithms (MOEA) have been proved very useful and effective in handling the difficulties imposed by the constrained portfolio optimization problem in a reasonable time.

MOEA techniques are able to generate a set of solutions approximating the efficient frontier in a single run, in contrast to the multiple runs needed in the case of single objective approaches [139], [140], [141], [156]. There exist many studies which applied MOEAs to solve the constrained portfolio optimization problem.

Liagkouras and Metaxiotis [142] based on the examination of the state-of-the art, provide the best practices for dealing with the complexities of the constrained portfolio optimization problem (CPOP). In particular, rigorous algorithmic and technical treatment is provided for the efficient incorporation of a wide range of real-world constraints into the MOEAs. The authors also address special configuration issues related to the application of MOEAs for solving the CPOP. Finally, by examining the state-of-the-art they identify the most appropriate performance metrics for the evaluation of the relevant results from the implementation of the MOEAs to the solution of the CPOP.

Lwin et al. [147] examine the extended Markowitz's portfolio optimization model with cardinality, quantity, pre-assignment and round lot constraints. The authors propose learning-guided hybrid multi-objective evolutionary algorithm to solve the constrained portfolio optimization problem. The proposed algorithm is compared against four well-known MOEAs, namely Non-dominated Sorting Genetic Algorithm (NSGA-II), Strength Pareto Evolutionary Algorithm (SPEAII), Pareto Envelope-based Selection Algorithm (PESA-II) and Pareto Archived Evolution Strategy (PAES). Experimental results on the constrained portfolio optimization problem demonstrate that the proposed algorithm significantly outperforms the four algorithms with respect to the quality of obtained efficient frontier.

Suksonghong et al. [204] experiment with several MOEAs, namely the nondominated sorting genetic algorithm II (NSGA-II), strength Pareto evolutionary algorithm II (SPEA-II) and compressed objective genetic algorithm II (COGA-II) for solving the portfolio optimization problem for a power generation company (GenCo) faced with different trading choices. The authors introduce an additional objective to enhance the diversification benefit alongside with the three original objectives of the mean–variance–skewness (MVS) portfolio framework. According to the authors the configuration of the algorithm with the additional objective offers better trade-off solutions while promoting investment diversification benefits for power generation companies, compared with the traditional MVS framework.

Liagkouras and Metaxiotis [140] propose a new probe guided version of the polynomial mutation (PLM) operator designed to be used in conjunction with MOEAs. The proposed Probe Guided Mutation (PGM) operator is validated by using data sets from six different stock markets. The performance of the proposed PGM operator is assessed in comparison with the one of the classical PLM with the assistance of the Non-dominated Sorting Genetic Algorithm II (NSGAII) and the Strength Pareto Evolutionary Algorithm 2 (SPEA2). According to the authors, the experimental results reveal that the proposed PGM operator outperforms with confidence the performance of the classical PLM operator for all performance metrics when applied to the solution of the cardinality constrained portfolio optimization problem (CCPOP).

Li and Xu [137] address the multi-objective portfolio selection model with fuzzy random returns. The authors consider three objectives, return, risk and liquidity. They also consider securities historical data, experts' opinions and investors' different attitudes in the portfolio selection process. To ensure the selection of the best solution, the authors propose a compromise approach-based genetic algorithm for solving the proposed model.

Corazza et al. [43] examine a class of measures that uses information contained both in lower and in upper tail of the distribution of the returns. They formulate the problem as a nonlinear mixed-integer portfolio selection model, which takes into account several constraints used in fund management practice. Because the aforementioned problem is NP-hard, they apply the heuristics Particle Swarm Optimization (PSO) to a reformulation of the mixed-integer model, where a standard exact penalty function is introduced.

Metaxiotis and Liagkouras [155] provide a review of the current state of research on portfolio management with the support of MOEAs. The authors present a methodological framework for conducting a comprehensive literature review on the (MOEAs) for the portfolio management. According to the authors this framework can be efficiently used to gain an understanding of the current state of the MOEAs for the portfolio management research field and to identify areas of concern with regard to MOEAs for the Portfolio Management research field.

Smimou [194] examines whether the home asset bias in a portfolio holding is associated with higher political instability risk, and to what extent international diversification among stocks, in the presence of such risk, outperforms domestic stock portfolios. The author experiments with alternative instability risk proxies under a discrete-time version of Markowitz's mean–variance portfolio selection problem.

In this chapter, we propose a multiobjective evolutionary algorithm, which incorporate a novel encoding scheme specially designed for dealing with the complexities of the CPOP alongside a new mutation and recombination operator tailor-made to work well with the new encoding scheme. The proposed encoding scheme has been designed having in mind to satisfy a two-fold objective. First should be able to efficiently facilitate the search process by locating optimum or near optimum solutions. Second, should be able to do that by making optimum use of the available computational resources.

The remainder of the chapter is structured as follows. Section 8.2 starts with a formal introduction of the portfolio optimization model and the various real-world constraints imposed to the model. In section 8.3, we formulate the Constrained Portfolio Optimization problem (CPOP) as a Mixed Integer Quadratic Programming (MIQP) problem. Section 8.4 provides a description of the proposed encoding scheme alongside a comparison with existing encoding schemes and potential benefits from its adoption. The new mutation and recombination operator are presented in section 8.5. Section 8.6 presents the experimental results for both the unconstrained and constrained problem and the relevant results are analyzed. Finally in section 8.7 we draw conclusions and we propose some potential paths for future research.

8.2 Portfolio optimization problem formulation

The Mean-Variance (M-V) portfolio optimization model introduced by Markowitz [151] for creating an optimum portfolio assumes that investors choose assets that maximize their portfolio returns and respectively minimize their portfolio risk.

Thus M-V model is a bi-objective problem as two conflicting objectives should be satisfied at any time. The M-V portfolio optimization model can be stated mathematically as follows:

Let Ω be the search space. Consider 2 objective functions f_1, f_2 where $\mathrm{f_i}: \Omega \rightarrow \mathrm{R^N}$. and $\Omega \subset \mathrm{R^N}$.

Optimize: $$f(w) = (f_1(w), f_2(w)) \qquad (1)$$

Maximize portfolio return: $$f_1(w) = \sum_{i=1}^{N} w_i \bar{r}_i \qquad (2)$$

Minimize portfolio risk: $$f_2(w) = \sum_{i=1}^{N} \sum_{j=1}^{N} w_i\, w_j \sigma_\iota \sigma_j \rho_{ij} \qquad (3)$$

subjected to

Budget constraint: $$\sum_{i=1}^{N} w_i = 1 \qquad (4)$$

All portfolios to have non-negative weights: $0 \leq w_i \leq 1$, $i = 1,2,\ldots, N$ (5)

where N is the number of assets available, w_i is the decision variable denoting the proportion of asset i in the portfolio, r_i is the expected return of asset i, ρ_{ij} is the correlation between asset i and j and $-1 \leq \rho_{ij} \leq 1$, σ_i represent the standard deviation of stock returns i. Equation (5) requires all portfolios to have non-negative weights, i.e. no short selling is allowed.

The M-V model can be extended to include the following real-world constraints that stem from industry regulations and practical concerns:

Cardinality constraint

Cardinality constraint limits the number of assets held in a portfolio. There are both practical and regulatory reasons that justify the imposition of lower and upper limit to the number of assets that compose a portfolio. The lower limit to the number of assets held in a portfolio is usually imposed to avoid excessive exposure to the idiosyncrasies of any particular asset, even though the portfolio's overall risk may appear acceptable. On the other hand the upper limit to the number of assets held in a portfolio is imposed to avoid excessive administrative and monitoring costs.

The cardinality constraint can be expressed as follows:

$$K_{min} \leq \sum_{i=1}^{N} q_i \leq K_{max}$$

where K_{min} is the minimum number of assets that a portfolio can hold, K_{max} is the maximum number of assets that a portfolio can hold, $q_i = 1$, for $w_i > 0$ and $q_i = 0$, for $w_i = 0$.

Floor and ceiling constraints

The justification of floor constraint stem from the fact that very small weightings of any asset will have no real influence on the portfolio's performance, but will add to administrative and monitoring costs. Likewise, ceiling constraint is imposed to avoid excessive exposure to the idiosyncrasies of any particular stock.

The Floor and ceiling constraints can be expressed as follows:

$$l_i \leq w_i \leq u_i, \ \forall\ i = 1,2,\dots, N.$$

where l_i is the minimum weighting that can be held of asset i $(i = 1, \dots, N)$, u_i is the maximum weighting that can be held of asset i $(i = 1, \dots,N)$ and $0 \leq l_i \leq u_i \leq 1, \ \forall\ i = 1,2,\dots, N.$

The incorporation of these constraints in the M-V model transforms the problem from a quadratic programming problem into a mixed-integer quadratic programming problem which is NP-hard [187]. There have been proposed several exact approaches for the cardinality constrained portfolio optimization problem [17], [20], [136], [187] however these approaches run into difficulties to generate solutions within reasonable time for large instances of the problem.

The aforementioned shortcoming of exact methods to generate solutions for the constrained case of the portfolio optimization problem within a reasonable frame of time led to the development of approximate approaches like the multiobjective evolutionary algorithms (MOEAs) that belong to the class of metaheuristics. MOEAs like any other metaheuristic technique cannot guarantee that in all cases will find the optimal solution, however have been proven to be efficient in finding near optimal solutions in a reasonable interval of time. The purpose of this chapter is to introduce an efficiently encoded MOEA that through the more efficient utilization of the available computational resources, such as processing power and genetic operators, manage to achieve good quality solutions within a reasonable frame of time.

Despite the inefficiency of exact approaches to deliver solutions within a reasonable frame of time, for more details about this matter please refer to Table 1, for reasons of comparison we formulate the constrained portfolio optimization problem as a mixed integer quadratic program (MIQP) and we solve it for a number of instances that range from 31 stocks for the smaller instance to the 225 stocks for the bigger instance. We also consider two bigger instances having 457 and 1317 stocks respectively, however because the required computational time for each exact efficient point exceeded considerably the maximum allowed time frame (*1,000* seconds) we only use approximate approaches for these two instances. In the following section we formulate the constrained portfolio optimization problem as a MIQP and solve it for the port1-port5 instances, for extracting the true efficient frontier (TEF).

8.3 Formulation of the Constrained Portfolio Optimization problem (CPOP) as a Mixed Integer Quadratic Programming (MIQP) problem

Below, we formulate the constrained portfolio optimization problem (CPOP) as a mixed integer quadratic programming (MIQP) problem and we extract the true efficient frontier for port1-port5 instances. Then, the relevant results are compared with the approximate efficient frontiers that are derived by the proposed MOEA and we draw useful conclusions about the efficiency of the proposed approach.

The MIQP problem for the CPOP can be expressed as follows:

Minimize portfolio risk $$\sum_{i=1}^{N}\sum_{j=1}^{N} w_i w_j \sigma_{ij} \quad (6)$$

Subject to: $$\sum_{i=1}^{N} w_i \bar{r}_i = R^* \quad (7)$$

$$\sum_{i=1}^{N} w_i = 1 \quad (8)$$

$$K_{\min} \le \sum_{i=1}^{N} z_i \le K_{\max} \quad (9)$$

$$l_i\, z_i \le w_i \le u_i\, z_i, \forall = 1,2,...,N \quad (10)$$

$$z_i \in [0,1], i = 1,...,N \quad (11)$$

Portfolio’s risk is expressed by total variance as shown by Eq. (6), which is minimized. The objective function Eq. (6), involves the covariance matrix (σ_{ij}), which is positive semidefinite and thus we are minimizing a convex function. Portfolio’s return is treated as a constraint as shown by Eq. (7). The budget constraint is represented by Eq. (8). The cardinality constraints are represented by Eq. (9). The lower limit to the number of assets held in a portfolio is represented by K_{min}.

Respectively, the upper limit to the number of assets held in a portfolio is represented by K_{max}. Eq. (10) represents the floor and ceiling constraints. Eq. (10) ensures that if any of asset i is held (i.e. $z_i = 1$) its weight must lie between the lower bound l_i and the upper bound u_i.

We set the minimum cardinality of the portfolio to two ($K_{min} = 2$) and the maximum cardinality of the portfolio to ten ($K_{max} = 10$) for all test problems. The participation of each asset in the portfolio is determined by the lower and upper bounds. We set the lower bound $l_i = 0.01$ and the upper bound $u_i = 0.99$, for each asset i, where $i = 1, \ldots, N$.

The main advantage of the MIQP approach is that it can find the optimum solution with certainty if the algorithm terminates. On the downsides, the running times can be prohibitive and the true efficient frontier (TEF) can only be calculated pointwise. Also, the representation of complex constraints is not always straightforward [24].

Having formulated the MIQP problem, the calculation of the true constrained efficient frontier for port1-5 problems is straightforward. We minimize portfolio risk as it is expressed by Eq. (6), subject to Eq. (7)-(11), for different levels of the R^* of Eq. (7). By varying R^* and solving the Mixed Integer Quadratic Program (MIQP) we obtain different levels of minimum portfolio risk. We used CPLEX version 12.5 to solve the above MIQP problem.

We utilized the following strategy [216], [217] to pointwise calculate the True Efficient Frontiers for port1-5 problems:

6. For each one of the port1-5 problems we know the range of return values that should be examined, as the securities' returns are known.
7. Let's say that we examine the port1 problem and r_L^{port1} represents the security with the lowest return among all securities of port1 problem. Respectively r_U^{port1} represents the security with the maximum return among all securities of port1 problem.
8. Given the return range [r_L^{port1}, r_U^{port1}] let M be the number of points that we wish to plot on the frontier. With the assistance of the r^{step} equation we divide the r_U^{port1} - r_L^{port1} interval into M equally distant points:

$$r^{step} = \frac{r_U^{port1} - r_L^{port1}}{M}$$

9. Then, we calculate the minimum portfolio risk for each of the following returns levels: [r_L^{port1} + r^{step}, r_L^{port1} + $2r^{step}$, ..., r_L^{port1} + Mr^{step}]. Please notice that we start from r_L^{port1} + r^{step} return level and not from r_L^{port1} because we have set $K_{min} = 2$. Solving the MIQP for r_L^{port1} return level and $K_{min} = 2$ would lead to an infeasible solution.
10. For port1-2 problems we used $M = 100$, for port3 we used $M = 50$ and for port4-5 problems we used $M = 25$, due to the excessively time-consuming process (i.e. in terms of running time) for calculating optimal points for the bigger instances of the problem.

8.4 Efficiently Encoded Multiobjective Portfolio Optimization Solver (EEMPOS)

The limitations of mixed integer approaches to handle efficiently the complexities associated with the solution of the constrained portfolio optimization problem especially for the bigger instances of the problem led to the development of metaheuristic approaches. In particular, multiobjective evolutionary algorithms (MOEAs) are well suited for dealing with the constrained portfolio optimization problem due to the multi-objective nature of the problem.

In the literature there are a considerable number of studies that utilize MOEAs to tackle the CPOP [155]. However, according to Liagkouras and Metaxiotis [142] the majority of the studies share a common element: the representation scheme. The solution representation is the most important element of a MOEA since it determines the search space [5]. An efficient representation scheme can improve significantly the performance of a MOEA. The purpose of this study is to introduce an efficiently encoded MOEA named Efficiently Encoded Multiobjective Portfolio Optimization Solver (EEMPOS) that through the more efficient utilization of the available computational resources, manage to achieve good quality solutions within a reasonable frame of time.

We start the presentation of the Efficiently Encoded Multiobjective Portfolio Optimization Solver (EEMPOS) by explaining the proposed representation scheme. According to Liagkouras and Metaxiotis [142] the selection of the most appropriate technique for the solution representation is critical for the performance and functionality of the entire MOEA.

According to the same study [142] the most popular type of encoding for solution vector representation in the case of the portfolio optimization problem is the hybrid representation introduced by Streichert et al. [200]. In the hybrid representation two vectors are used for defining a portfolio:

A binary vector that specifies whether a particular asset participates in the portfolio and a real-valued vector used to compute the proportions of the budget invested in the assets.

$\Delta = \{z_1, \dots, z_N\}, \quad z_i = \{0, 1\}, \quad i = 1, \dots, N$

$W = \{w_1, \dots, w_N\}, \quad 0 \leq w_i \leq 1, \quad i = 1, \dots, N$

Moreover through the hybrid encoding a number of constraints like for example the cardinality constraint can be satisfied. For instance, if the number of 1's in Δ overcomes the maximum cardinality, then those assets that have the minimum weight in $W$are deleted, by changing its value from 1 to 0 in Δ.

According to Streichert et al. [200] the usage of hybrid encoding for the solution representation facilitates the process of adding or removing an asset to the portfolio by simply mutating the bit-string Δ. Finally, the hybrid encoding is altered by mutating and crossing each genotype W and Δ separately from each other.

In this study we propose a new encoding scheme that makes better use of the available computational resources especially for big instances of the cardinality constrained portfolio optimization problem.

The proposed encoding scheme is composed of a real-valued vector and a binary vector. The real-valued vector contains information about the selected assets and the corresponding budget invested in each one of these assets. The binary vector specifies whether a particular asset participates in the portfolio. Both real-valued and binary vectors must be of same size and equal to the upper limit of the cardinality constraint.

Below we provide an analytical description of the proposed encoding scheme.

step 1. The algorithm starts by creating a random parent population p.

step 2. The parent population is composed of *2p* randomly generated solution vectors, half of them (i.e. p) are real-valued and the other half (i.e. p) binary-valued.

step 3. Each real-valued and binary-valued vector is populated as follows:

step 3.1. Randomly are generated K_{max} weights, within the specified range set by the floor and ceiling constraints [0.01, 0.99] to populate the real-valued vector. Please recall that K_{max} is the maximum number of assets that a portfolio can hold and corresponds to the upper limit of the cardinality constraint.

Suppose K_{max} *= 10, floor constraint = 0.01* and *ceiling constraint = 0.99,* a random real-valued solution vector will looks like the following:

A real-valued vector before being populated:

r_1	r_2	r_3	r_4	r_5	r_6	r_7	r_8	r_9	r_{10}

⇩

A real-valued vector after being populated:

0.05	0.64	0.73	0.12	0.24	0.89	0.33	0.71	0.54	0.42

step 3.2. The binary-valued vector is populated as follows: A random number r_{value} is generated that must satisfy both the lower (K_{min}) and the upper (K_{max}) bound of the cardinality constraint. The r_{value} determines the number of *1s* in each binary-valued solution vector. The allocation of *1s* in the binary-valued solution vector is done randomly. The remaining K_{max} - r_{value} positions are filled up with *0s*.

Suppose K_{min} *= 2,* K_{max} *= 10* and r_{value} *=5. The 1s in the binary-valued vector are randomly allocated.* Thus, the binary-valued solution vector will looks like the following:

1	0	0	0	0	1	1	0	1	1

step 3.3. At the end of this process we end up with a real-valued vector and the binary-valued vector:

real-valued vector

0.05	0.64	0.73	0.12	0.24	0.89	0.33	0.71	0.54	0.42

binary-valued vector

1	0	0	0	0	1	1	0	1	1

Then, we perform a normalization mechanism to satisfy the budget constraint and the floor and ceiling constraints. For the normalization of the weights and the satisfaction of the floor and ceiling constraints we apply the following formula proposed by Mishra et al. [158]:

$w_i' = a_i z_i + \frac{w_i z_i}{\sum_{i=1}^{m} w_i z_i}(b_i z_i - \sum_{i=1}^{m} a_i z_i)$, where $z_i = \{0,1\}$ and $a_i \leq w_i \leq b_i, \forall\, i = 1,2,\dots,m$. Please notice that the aforementioned formula takes into account both the real-valued and the binary-valued solution vector into the normalization process.

Thus results the following normalized weight vector:

normalized weight vector

0.031	0	0	0	0	0.395	0.149	0	0.238	0.187

step 3.4. Then from the available pool of assets *N,* are selected randomly with equal probability K_{max} assets, where K_{max} is the maximum number of assets that a portfolio can hold and corresponds to the upper limit of the cardinality constraint. Please notice that once a stock is selected, automatically is excluded furthermore from the selection process. This is done to make sure that the same asset is not picked up more than once for the same real-valued solution vector.

For explanatory reasons, suppose that we examine the *port5* test problem which is composed from *225* assets and $K_{max} = 10$. Let's assume that the following assets are randomly being selected.

23	5	112	42	14	8	211	48	178	73

step 3.5. So far we have shown that each decision variable of the real-valued vector contains information about the budget invested in each asset. Now, we add information regarding the selected assets. Thus, each decision variable of the real-valued vector will contain information about the selected asset and the corresponding budget invested in this asset. Below, we provide an example of a typical real-valued vector with $K_{max} = 10$.

r_1	r_2	r_3	r_4	r_5	r_6	r_7	r_8	r_9	r_{10}

Each decision variable of the real-valued vector $r_j \; , \; j = 1, .., K_{max}$ is composed from two parts, the budget invested in each asset and the selected asset.

normalized weight vector

0.031	0	0	0	0	0.395	0.149	0	0.238	0.187

+

selected assets

23	5	112	42	14	8	211	48	178	73

⇩

real-valued vector

0.031 + 023	0.000 + 005	0.000 + 112	0.000 + 042	0.000 + 014	0.395 + 008	0.149 + 211	0.000 + 048	0.238 + 178	0.187 + 073

For the representation of the budget invested in each asset we allow three decimal places for better accuracy, followed by the selected asset. Thus, the $r_j \; , \; j = 1, .., K_{max}$ decision variables of the real-valued vector are formulated as follows:

$r_1 = 0.031023$	$r_2 = 0.000005$	$r_3 = 0.000112$	$r_4 = 0.000042$
$r_5 = 0.000014$	$r_6 = 0.395008$	$r_7 = 0.149211$	$r_8 = 0.000048$
$r_9 = 0.238178$	$r_{10} = 0.187073$		

Thus, the proposed representation scheme is composed of a real-valued vector and a binary vector. The real-valued vector contains information about the selected assets and the corresponding budget invested in each one of these assets. The binary-valued vector specifies whether a particular asset participates in the portfolio. Please notice that the information included in real-valued vector regarding the selected assets and the corresponding budget invested in each asset can be easily extracted when it is being required by the algorithm.

There are two striking differences between the proposed encoding scheme and the one introduced by Streichert et al. [200]:

(i.) In Streichert et al. [200] approach the real-valued vector contains only information about the proportions of the budget invested in the assets, while in our approach the real-valued vector contains information about the selected assets and the corresponding budget invested in each one of these assets.

(ii.) In Streichert et al. [200] approach the size of real-valued and binary-valued vectors is equal to the available pool of stocks, while in our approach the size of real-valued and binary-valued vectors is equal to the upper limit of the cardinality constraint (K_{max}).

Below, we will attempt to analyze the benefits from the introduction of the proposed representation scheme. Also, in section 6 we provide an experimental analysis of the proposed representation scheme when is applied along with the new mutation and crossover operators for solving the constrained portfolio optimization problem (CPOP).

A first obvious benefit from the adoption of the proposed representation scheme can be analysed in terms of computational resources. In the proposed representation scheme the size of both the real-valued and binary-valued vectors is limited by the upper limit of the cardinality constraint (K_{max}) in contrast to the Streichert et al. [200] approach where the size of real-valued and binary-valued vectors is equal to the available pool of stocks. This benefit of the proposed approach can be very important in terms of computational resources especially for big instances of the CPOP. It is not hard to imagine that traversing a *10* decision variables vector (assuming K_{max} = *10*) is way much more cost-efficient than traversing a *1,317* decision variables vector for a big instance of the CPOP like the port7.

Moreover given the fact that according to a number of empirical studies [21], [84] , [89] the full diversification benefit is realized for portfolios sizes up to twenty securities, there is no need to set the upper limit of the cardinality constraint (K_{max}) more than 20 or 30 assets in most of the cases independently of the pool of the available assets. Thus, it becomes clear that in most of the cases it is sufficient to set K_{max} = 20 independently of how big is the universe of the examined instances. This in turn can be translated into considerable benefits in terms of computational resource especially for big instances of the CPOP as in our case the solution vectors are limited by the K_{max} up to a size of maximum 20 or 30 assets while in the case of Streichert et al. [200] approach the solution vectors increase in accordance to the available pool of stocks.

Another benefit of the proposed representation scheme is that it provides more efficient utilization of the genetic operators. To explain the above statement just assume that we examine the port7 test problem having 1317 assets and the minimum and maximum cardinality (i.e. non zero weights) are set equal to *10* ($K_{min} = K_{max} = 10$) and the mutation probability is $P_m = 0.2$. In the case of the proposed representation scheme the size of the solution vector will be equal to $K_{max} = 10$, however case of the Streichert et al. [200] approach both the real-valued and the binary-valued solution vectors will be composed of *1,317* decision variables.

Now consider the case where the real-valued solution vector undergoes the mutation process with *0.2* probability. Given the probability rate of *0.2*, on average one in every five decision variables will undergo the mutation process. Thus, in the case of the Streichert et al. [200] approach on average *264* (*=0.2*1,317*) decision variables will undergo the mutation process in total of *1,317* variables. Likewise, in the case of the proposed representation scheme *2* decision variables will be mutated in total of *10* variables.

Needless to say that a mutation or recombination process that involves *1,317* decision variables is way much more expensive in terms of computational resources (i.e. required processing power and computational time) than a process that involves only *10* variables.

Besides the proposed representation scheme makes better utilization of the genetic operators. To provide an explanation of the aforementioned statement, just consider the case where the minimum and maximum cardinality are set equal to *10* ($K_{min} = K_{max} = 10$) for both examined cases.

In the case of the proposed encoding scheme, given a mutation probability rate of *0.2* it should be expected with certainty that two non-zero (i.e holding weight) decision variables will participate in the mutation process from each solution vector. However, in the case of Streichert et al. [200] approach it is quite possible that none of the non-zero decision variables will participate in the mutation process. Actually, in the case of Streichert et al. approach the probability of mutating a single non-zero decision variable, given the length of solution vector (*1,317*) for port7 problem and a mutation rate of 0.2 is *20%*, while the probability of mutating *2* non-zero decision variables from the same solution vector is only *4%.*

From the aforementioned analysis it becomes easily understood that the proposed representation scheme is more cost-efficient in terms of computational resources and furthermore provides better utilization of the genetic operators, as it is designed to focus the search effort to the most meaningful areas of the search space. To fully benefit from the proposed representation scheme, in section 5 we introduce a new mutation and recombination operator specially designed to work well with the proposed encoding scheme.

8.5 The proposed Mutation and Recombination operator

In this section we introduce a new mutation and recombination operator specially designed to work well with the proposed representation scheme. The proposed genetic operators have been designed to satisfy the need of improving the already found good solutions and simultaneously to be able to explore other promising areas of the search landscape. This much asked balance between exploration and exploitation is achieved through the introduction of innovative mechanisms that will be analyzed below.

8.5.1 A Two-Phase Mutation Operator for the Constrained Portfolio Optimization Problem

Typically, the mutation operator takes responsibility for the preservation of population's diversity. In this chapter, we propose a two-phase mutation operator (TPMO) specially designed to work well with the proposed representation scheme for solving the constrained portfolio optimization problem (CPOP).

We employ a variation of the polynomial mutation (PLM) [61] operator for the real-valued solution vector specially adapted to fit the needs of the proposed representation scheme and binary mutation for the discrete (i.e. binary-valued) solution vector.

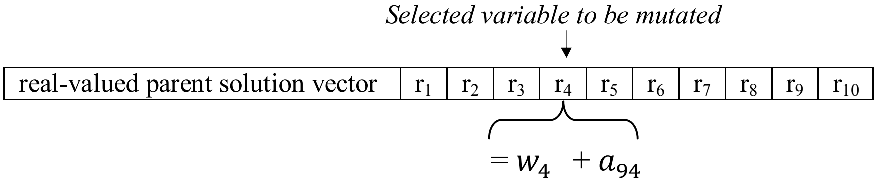
We start the presentation of the proposed TPMO by analyzing the mutation process of the real-valued solution vector. As we already explained is section 8.4, the real-valued solution vector contains information about the selected assets and the corresponding budget invested in each one of these assets. Also, the size of both the real and binary solution vectors is restricted by the upper limit of the cardinality constraint (K_{max}). Given the fact that the available pool of stocks for all examined test instances (port1-7) is considerably larger than K_{max} , assuming a K_{max} value equal to *10*, we should devise an updating mechanism that allows the replacement of some assets of the real-valued solution vector with other assets from the available pool of securities. At the same time, the mutation operator should be able to probe limited areas of the search landscape with the hope of improving a promising solution that has already been found. Below, we analyze step-by-step how we incorporate the aforementioned principals to the proposed mutation operator.

We start our analysis by presenting Fig. 8.1 that provides the flowchart of the proposed mutation operator. As can be seen from Fig.1, *if rand* $\leq P_m$ then a decision variable is selected to be mutated. Suppose a hypothetical real-valued solution vector $\mathbf{r} = (r_1\ , r_2\ , \dots , r_{K_{max}})$. Each real-valued decision variable r_i , $i = 1, 2, \dots , K_{max}$ contains information about the selected asset a_j, $j = 1, 2, \dots , N$, where N is the available pool of assets and the corresponding budget w_i , $i = 1, 2, \dots , K_{max}$ invested in this particular asset. The budget w_i invested in each asset, can take values in the interval: $w_i^{(L)} \leq w_i \leq w_i^{(U)}$, $i = 1, 2, \dots , K_{max}$ where $w_i^{(L)}$ and $w_i^{(U)}$ stand respectively for the floor and ceiling constraints.

```
Begin
$\eta_m$ = distribution index;
 for i=0 to P; // where P is the population size
     for z=0 to $K_{max}$;  // where $K_{max}$ is the upper limit of the cardinality constraint
       rand → [0, 1];
         if (rand <= mutation_probability) then
             $r_p$ = getValue(z);
             $r_p = w_p + a_p$ //where $w_p$ is the assigned weight and $\alpha_p$ the corresponding asset of the parent solution
             $w^L$ = getLowerBound(z);
             $w^U$ = getUpperBound(z);

             binary rand_mut = generateValue (0 or 1);

             if (rand_mut = 0)
                 $\delta_1 = \frac{w_p - w^L}{w^U - w^L}$      $\delta_2 = \frac{w^U - w_p}{w^U - w^L}$

                 $r$ → [0, 1];

                 if (r<= 0.5) then
                     $\delta_q = [2r + (1-2r)(1-\delta_1)^{\eta_m+1}]^{\frac{1}{\eta_m+1}}$
                 else
                     $\delta_q = 1 - [2(1-r) + 2(r-0.5)(1-\delta_2)^{\eta_m+1}]^{\frac{1}{\eta_m+1}}$
                 end if

                 $w^{child} = w_p + \delta_q(w^U - w^L)$
                 $w^{child'}$ = normalized($w^{child}$)  //$w^{child'}$ satisfies the budget and floor and ceiling constraint
                 $r_c = w^{child'} + a_p$
                 child_solution_vector = parent_solution_vector.setValue(z , $r_c$ );
             else if (rand_mut = 1)
                 rand_asset→ [1, N]; //where N is the available pool of assets, e.g. for port5, N = 225

                 $\delta_1 = \frac{w_p - w^L}{w^U - w^L}$      $\delta_2 = \frac{w^U - w_p}{w^U - w^L}$

                 $r$ → [0, 1];

                 if (r<= 0.5) then
                     $\delta_q = [2r + (1-2r)(1-\delta_1)^{\eta_m+1}]^{\frac{1}{\eta_m+1}}$
                 else
                     $\delta_q = 1 - [2(1-r) + 2(r-0.5)(1-\delta_2)^{\eta_m+1}]^{\frac{1}{\eta_m+1}}$
                 end if

                 $w^{child} = w_p + \delta_q(w^U - w^L)$
                 $w^{child'}$ = normalized($w^{child}$)  //$w^{child'}$ satisfies the budget and floor and ceiling constraint
                 $r_c = w^{child'}$ + rand_asset
                 child_solution_vector = parent_solution_vector.setValue(z , $r_c$ );
             endif
         endif
     endfor
 endfor
```

Fig. 8.1. Two-phase mutation operator (TPMO) pseudo code

Also suppose that we examine the *port5* problem with *225* assets, K_{max} = 10 and a random number *rand* ≤ P_m occurs for the 4th decision variable. That means that the 4th decision variable r_4 of the real-valued parent solution vector should be mutated. Before being able to proceed with the mutation, we need to decompose the r_4 and extract the information about the assigned weight w_4 and the selected asset a_{94} (assuming that the 94th asset is selected).



Selected variable to be mutated

real-valued parent solution vector	r_1	r_2	r_3	r_4	r_5	r_6	r_7	r_8	r_9	r_{10}

$= w_4 \ + a_{94}$

Then with the assistance of a random binary variable *rand_mut* taking values either *0* or *1*, the mutation operator is separated into two different phases. As shown in Fig. 1 in the case that the random binary variable *rand_mut = 0* we perform the standard polynomial mutation (PLM) operator to the extracted weight w_4 .

As soon as the PLM process is completed, a new weight w_4^{child} emerges. A constraints correction mechanism is applied to make sure that both the budget constraint and the floor and ceiling constraints are satisfied [158]. After completing the correction process the w_4^{child} takes its new adjusted value $w_4^{child'}$. Please notice that the constraints correction mechanism besides $w_4^{child'}$ will adjust all other weights w_i , $i = \{1,2,3\}\, and\, \{5, \ldots ,10\}$ of the parent solution vector. The adjusted weights are represented by w_i', $i = \{1,2,3\}\, and\, \{5, \ldots ,10\}$. Accordingly, will be updated the real-valued decision variables r_i' = w_i'+ a_j $i = \{1,2,3\}\, and\, \{5, \ldots ,10\}$. Step 3.3 of section 8.4 provides a description of the constraints correction mechanism.

Subsequently, the adjusted weight $w_4^{child'}$ will be used to recompose the child real-valued decision variable as follows r_4^c = $w_4^{child'}$+ a_{94}. Then, the r_4^c decision variable is reinserted into the solution vector at the position of the selected variable to be mutated (i.e. *4th*) and thus is created the child solution vector.

Mutated variable

real-valued child solution vector	r_1'	r_2'	r_3'	r_4^c	r_5'	r_6'	r_7'	r_8'	r_9'	r_{10}'

$$= w_4^{child'} + a_{94}$$

The aforementioned process allows the algorithm to explore regions of the search space close to the already found solutions.

However, in the case that the random binary variable *rand_mut* takes value *1* we follow a different approach. In particular, in this case with the assistance of a random variable *rand_asset* a single asset is selected from the available pool of stocks and inserted into the position of the selected asset to be mutated. All assets from the available pool of assets participate in this random process with equal probability. Thus for instance, in the case of the *port5* problem, all *225* assets will participate in this random selection process with equal probability. Please notice that all available securities participate in this process independently of their presence or not in the solution vector.

We assume that the *4th* decision variable (r_4) is randomly selected to be mutated and *rand_mut = 1.* Again, as in the previous example before being able to proceed with the mutation we need to decompose the r_4 and extract the information about the assigned weight w_4 and the selected asset a_{94} (i.e. assuming that the 94th asset is selected).

Selected variable to be mutated

real-valued parent solution vector	r_1	r_2	r_3	r_4	r_5	r_6	r_7	r_8	r_9	r_{10}

$$= w_4 + a_{94}$$

Then, with the assistance of the random variable *rand_asset* a single asset is selected from the available pool of assets $a_j,\ j = 1, 2, \dots, N$ (*225* assets in the case of the *port5* problem) and replaces the a_{94} asset into the selected decision variable (i.e. r_4) to be mutated. Thus for instance, assuming that with the assistance of the random variable *rand_asset* the a_{158} asset is randomly selected to replace a_{94} asset.

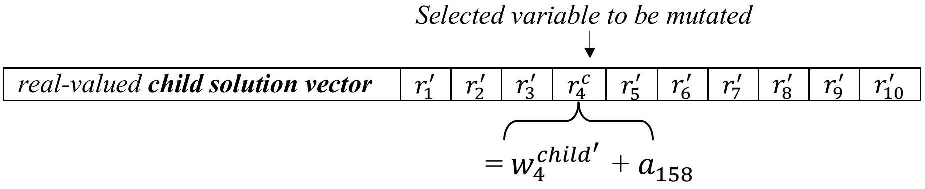
At the same time the polynomial mutation (PLM) operator is applied to the extracted weight w_4 and thus is generated a child weight w_4^{child}. Then a constraints correction mechanism is applied to make sure that both the budget constraint and the floor and ceiling constraints are satisfied [158]. After completing the correction process the w_4^{child} takes its new adjusted value $w_4^{child'}$.

The constraints correction mechanism besides $w_4^{child'}$ will adjust all other weights w_i , $i = \{1, 2,3\}\, and\ \{5, \dots ,10\}$ of the parent solution vector. The adjusted weights are represented by w_i', $i = \{1, 2,3\}\, and\ \{5, \dots ,10\}$. Accordingly, will be updated the real-valued decision variables $r_i' = w_i' + a_j$ $i = \{1, 2,3\}\, and\ \{5, \dots ,10\}$. Step 3.3 of section 4 provides a description of the constraints correction mechanism.

The updated decision variable after the completion of the mutation process will be formulated as follows: $r_4^c = w_4^{child'} + a_{158}$.

At this particular point we identify two cases:

(i) In the first case that the randomly selected asset a_{158} is not already represented in the parent solution vector, we simply reinsert the child decision variable $r_4^c = w_4^{child'} + a_{158}$ at the position of the parent decision variable $r_4 = w_4 + a_{94}$ and thus is formulated the child solution vector.



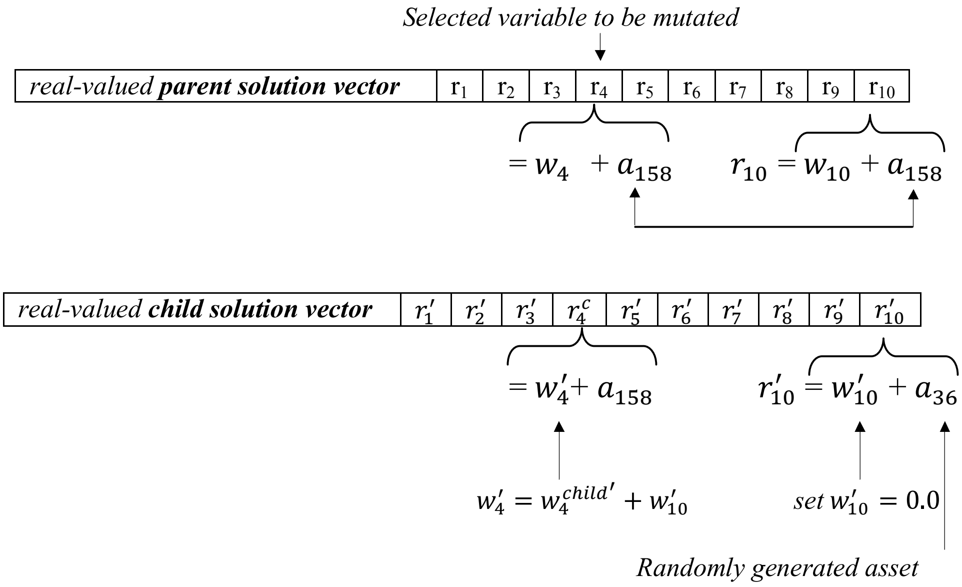
Selected variable to be mutated

real-valued ***child solution vector***	r_1'	r_2'	r_3'	r_4^c	r_5'	r_6'	r_7'	r_8'	r_9'	r_{10}'

$= w_4^{child'} + a_{158}$

(ii) In the second case that the randomly selected asset a_{158} is already represented in the parent solution vector, we need to calculate the aggregate weight for asset a_{158} by taking into account both occurrences of a_{158} asset into the same solution vector. As soon as we calculate the aggregate weight of a_{158} we assign it to the decision variable to be mutated (r_4^c). Obviously to the other decision variable (let's say r_{10}) that is holding weight of the a_{158} asset, we assign zero weight and the a_{158} asset is replaced by a randomly generated asset from the available pool of assets with only restriction the exclusion from this process of the already present assets in the solution vector.

Finally a constraints correction mechanism is applied to make sure that the floor and ceiling constraints are satisfied [158].


Selected variable to be mutated
real-valued **parent solution vector**
r_1 r_2 r_3 r_4 r_5 r_6 r_7 r_8 r_9 r_{10}
$= w_4 + a_{158}$
$r_{10} = w_{10} + a_{158}$
real-valued **child solution vector**
r'_1 r'_2 r'_3 r^c_4 r'_5 r'_6 r'_7 r'_8 r'_9 r'_{10}
$= w'_4 + a_{158}$
$r'_{10} = w'_{10} + a_{36}$
$w'_4 = w_4^{child'} + w'_{10}$
set $w'_{10} = 0.0$
Randomly generated asset

Mutation operator for the binary solution vector

For the binary solution vector we apply a bit-flip (BF) mutation operator that is applied to each gene with a certain given probability.

8.5.2 A Two-Phase Recombination Operator for the Constrained Portfolio Optimization Problem

In this section we propose a recombination operator specially designed to work well with the proposed representation scheme for solving the constrained portfolio optimization problem (CPOP).

As we explained in section 8.5.1 due to the fact that in the proposed representation scheme the size of both the real and binary solution vectors is restricted by the upper limit of the cardinality constraint (K_{max}) we need to introduce into the mutation and crossover operator an updating mechanism that will allow the replacement of some assets of the real-valued solution vector with other assets from the available pool of securities. At the same time, the recombination operator should efficiently exploit the available search space (i.e. to perform small adjustments in the weight of the already selected assets). The proposed two-phase recombination operator (TPRO) efficiently satisfies these two requirements.

We employ a variation of the simulated binary crossover (SBX) [54] operator for the real-valued solution vector specially adapted to fit the needs of the proposed representation scheme and single point crossover for the discrete (i.e. binary-valued) solution vector.

We start the presentation of the proposed TPRO by analyzing the recombination process of the real-valued solution vector.

Figure 8.2 provides the flowchart of the proposed recombination operator. As shown from Fig.8.2, *if rand* $\leq P_c$ then two decision variables are selected to be crossovered one from each parent solution vector. Suppose a hypothetical real-valued solution vector $\mathbf{r} = (r_1 \ , \ r_2 \ , \dots , r_{K_{max}})$. As we explained in section 8.4 each real-valued decision variable $r_i \ , \ i = 1, 2, \dots , K_{max}$ contains information about the selected asset a_j, $j = 1, 2, \dots , N$, where N is the available pool of assets and the corresponding budget $w_i \ , \ i = 1, 2, \dots , K_{max}$ invested in this particular asset. The budget w_i invested in each asset, can take values in the interval: $w_i^{(L)} \leq w_i \leq w_i^{(U)}, \ i = 1, 2, \dots , K_{max}$ where $w_i^{(L)}$ and $w_i^{(U)}$ stand respectively for the floor and ceiling constraints.

```
for z=0 to $K_{max}$; // where $K_{max}$ is the upper limit of the cardinality constraint
    rand → [0, 1];
    if (rand <= crossover_probability) then

        $r_p^{(1)}$ = $parent\_solution^{(1)}$.getValue(z);
        $r_p^{(2)}$ = $parent\_solution^{(2)}$.getValue(z);
        $w^L$ = getLowerBound(z);
        $w^U$ = getUpperBound(z);

        //where $w_p$ is the assigned weight and $\alpha_p$ the corresponding asset of the parent solution 1 and 2 respectively
        $r_p^{(1)} = w_p^{(1)} + a_p^{(1)}$
        $r_p^{(2)} = w_p^{(2)} + a_p^{(2)}$

        binary rand_cross = generateValue (0 or 1);

        if (rand_cross = 0)
            $\beta = 1 + \frac{2}{w_p^{(2)} - w_p^{(1)}} min\left[(w_p^{(1)} - w^{(L)}), (w^{(U)} - w_p^{(2)})\right]$

            $\alpha = 2 - \beta^{-(\eta_c+1)}$

            $u$ → [0, 1];

            if ($u \leq \frac{1}{a}$) then
                $\beta_q = (au)^{1/(\eta_c+1)}$
            else if ($u > \frac{1}{a}$)
                $\beta_q = \left(\frac{1}{2-au}\right)^{1/(\eta_c+1)}$
            end if

                //The updated weights are then calculated as follows:
                $w_c^{(1)} = 0.5\left[\left(w_p^{(1)} + w_p^{(2)}\right) - \beta_q |w_p^{(2)} - w_p^{(1)}|\right]$
                $w_c^{(2)} = 0.5\left[\left(w_p^{(1)} + w_p^{(2)}\right) + \beta_q |w_p^{(2)} - w_p^{(1)}|\right]$

                //The child decision variables are then formulated as follows:
                $c^{(1)} = w_c^{(1)} + a_p^{(1)}$
                $c^{(2)} = w_c^{(2)} + a_p^{(2)}$

                child_solution_vector $^{(1)}$ = parent_solution_vector$^{(1)}$.setValue(z , $c^{(1)}$);
                child_solution_vector $^{(2)}$ = parent_solution_vector$^{(2)}$.setValue(z , $c^{(2)}$);
```

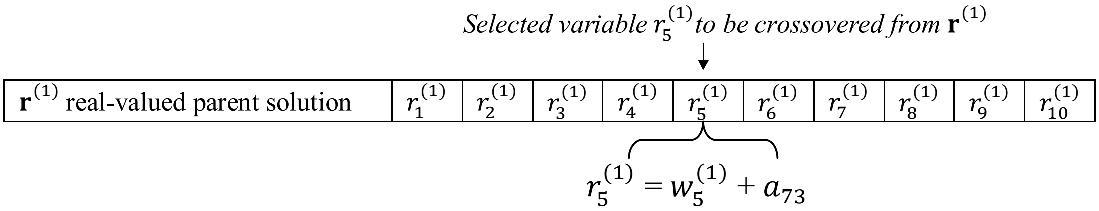
Continued on next page…

Continued from the previous page…

```
            else if ($rand\_cross = 1$)

                $rand\_asset^{(1)} \longrightarrow$ [1, N]; //where N is the available pool of assets, e.g. for port5, N=225
                $rand\_asset^{(2)} \longrightarrow$ [1, N];

                $\beta = 1 + \frac{2}{w_p^{(2)} - w_p^{(1)}} min\left[(w_p^{(1)} - w^{(L)}), (w^{(U)} - w_p^{(2)})\right]$

                $\alpha = 2 - \beta^{-(\eta_c+1)}$

                $u \longrightarrow$ [0, 1];

                if ($u \leq \frac{1}{a}$) then
                    $\beta_q = (au)^{1/(\eta_c+1)}$
                else if ($u > \frac{1}{a}$)
                    $\beta_q = \left(\frac{1}{2-au}\right)^{1/(\eta_c+1)}$
                end if

                    //The updated weights are then calculated as follows:
                    $w_c^{(1)} = 0.5[(w_p^{(1)} + w_p^{(2)}) - \beta_q |w_p^{(2)} - w_p^{(1)}|]$
                    $w_c^{(2)} = 0.5[(w_p^{(1)} + w_p^{(2)}) + \beta_q |w_p^{(2)} - w_p^{(1)}|]$

                    //The child decision variables are then formulated as follows:
                    $c^{(1)} = w_c^{(1)} + rand\_asset^{(1)}$
                    $c^{(2)} = w_c^{(2)} + rand\_asset^{(2)}$

                    child_solution_vector $^{(1)}$ = parent_solution_vector$^{(1)}$.setValue(z , $c^{(1)}$);
                    child_solution_vector $^{(2)}$ = parent_solution_vector$^{(2)}$.setValue(z , $c^{(2)}$);

        endif
    endfor
```

Fig. 8.2. Two-phase recombination operator (TPRO) pseudo code

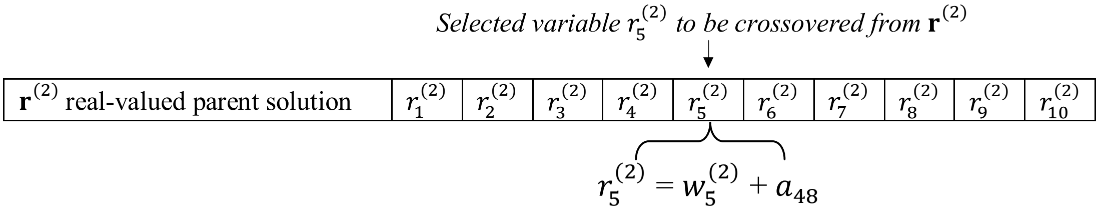
In the proposed two-phase recombination operator (TPRO), two parent decision variables $r_p^{(1)}$ and $r_p^{(2)}$ generate two child decision variables $c^{(1)}$ and $c^{(2)}$. For explanatory purposes we assume that the $r_5^{(1)}$and $r_5^{(2)}$ decision variables of the $\mathbf{r}^{(1)}$ and respectively $\mathbf{r}^{(2)}$ parent solution vectors are randomly selected to be crossovered. Before being able to proceed with the recombination we need to decompose $r_5^{(1)}$ and $r_5^{(2)}$ decision variables and extract the information about the assigned weight and the selected asset. For explanatory purposes we assume $r_5^{(1)} = w_5^{(1)} + a_{73}$ and $r_5^{(2)} = w_5^{(2)}$ + a_{48} , where $w_5^{(1)}$ and a_{73} correspond to the weight and asset of $r_5^{(1)}$ decision variable and respectively $w_5^{(2)}$ and a_{48} correspond to the weight and asset of $r_5^{(2)}$ decision variable.



Selected variable $r_5^{(1)}$ *to be crossovered from* $\mathbf{r}^{(1)}$

$\mathbf{r}^{(1)}$ real-valued parent solution	$r_1^{(1)}$	$r_2^{(1)}$	$r_3^{(1)}$	$r_4^{(1)}$	$r_5^{(1)}$	$r_6^{(1)}$	$r_7^{(1)}$	$r_8^{(1)}$	$r_9^{(1)}$	$r_{10}^{(1)}$

$$r_5^{(1)} = w_5^{(1)} + a_{73}$$



Selected variable $r_5^{(2)}$ *to be crossovered from* $\mathbf{r}^{(2)}$

$\mathbf{r}^{(2)}$ real-valued parent solution	$r_1^{(2)}$	$r_2^{(2)}$	$r_3^{(2)}$	$r_4^{(2)}$	$r_5^{(2)}$	$r_6^{(2)}$	$r_7^{(2)}$	$r_8^{(2)}$	$r_9^{(2)}$	$r_{10}^{(2)}$

$$r_5^{(2)} = w_5^{(2)} + a_{48}$$

Then with the assistance of a random binary variable *rand_cross* taking values either *0* or *1*, the recombination operator is separated into two different phases. As shown in Fig. 2 in the case that the random binary variable *rand_cross = 0* we perform the standard simulated binary crossover (SBX) operator to the extracted $w_5^{(1)}$ and $w_5^{(2)}$ weights. As soon as the SBX process is completed, the updated $w_{c_5}^{(1)}$ and $w_{c_5}^{(2)}$ weights are generated. A constraints correction mechanism is applied to make sure that both the budget constraint and the floor and ceiling constraints are satisfied [142]. After completing the correction process the $w_{c_5}^{(1)}$ and $w_{c_5}^{(2)}$ take their new adjusted value $w_{c_5'}^{(1)}$ and $w_{c_5'}^{(2)}$.

Please notice that the constraints correction mechanism besides $w_{c_5'}^{(1)}$ and $w_{c_5'}^{(2)}$ will adjust all other weights of both parent solution vectors (i.e. $\mathbf{r}^{(1)}$ and $\mathbf{r}^{(2)}$). The adjusted weights are represented by $w_{i'}^{(1)}$ and $w_{i'}^{(2)}$, $i = \{1, 2,3,4\}\ and\ \{6, \dots ,10\}$. Step 3.3 of section 4 provides a description of the constraints correction mechanism.

Accordingly, will be updated the real-valued decision variables $r_i' = w_i' + a_j \quad i = \{1, 2,3,4\}\ and\ \{6, \dots ,10\}$.

Subsequently, the adjusted $w_{c_5'}^{(1)}$ and $w_{c_5'}^{(2)}$ weights will be used to recompose the child decision variables as follows $r_{c_5}^{(1)} = w_{c_5'}^{(1)} + a_{73}$ and $r_{c_5}^{(2)} = w_{c_5'}^{(2)} + a_{48}$. Then, the $r_{c_5}^{(1)}$and $r_{c_5}^{(2)}$decision variables are reinserted into the solution vectors at the position of the selected variable to be crossovered (i.e. *5th*) and thus are created the child solution vectors.

Selected variable to be crossovered from $\mathbf{r}^{(1)}$

↓

$\mathbf{r}^{(1)}$ *real-valued* ***child solution***	$r_{1'}^{(1)}$	$r_{2'}^{(1)}$	$r_{3'}^{(1)}$	$r_{4'}^{(1)}$	$r_{c_5}^{(1)}$	$r_{6'}^{(1)}$	$r_{7'}^{(1)}$	$r_{8'}^{(1)}$	$r_{9'}^{(1)}$	$r_{10'}^{(1)}$

$$r_{c_5}^{(1)} = w_{c_5'}^{(1)} + a_{73}$$

Selected variable to be crossovered from $\mathbf{r}^{(2)}$

↓

$\mathbf{r}^{(2)}$ *real-valued* ***child solution***	$r_{1'}^{(2)}$	$r_{2'}^{(2)}$	$r_{3'}^{(2)}$	$r_{4'}^{(2)}$	$r_{c_5}^{(2)}$	$r_{6'}^{(2)}$	$r_{7'}^{(2)}$	$r_{8'}^{(2)}$	$r_{9'}^{(2)}$	$r_{10'}^{(2)}$

$$r_{c_5}^{(2)} = w_{c_5'}^{(2)} + a_{48}$$

However, in the case that the random binary variable *rand_cross* take value *1* we follow a different strategy.

In this case with the assistance of a random variable *rand_asset* two assets from the available pool of assets are selected.

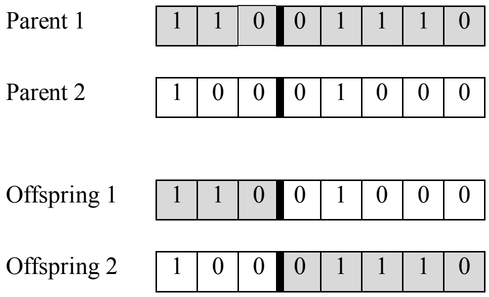
All assets from the available pool of assets participate in this random process with equal probability. Thus for instance, in the case of the *port5* problem, all *225* assets will participate in this random selection process with equal probability for the election of the two random assets (a_{r_1} and a_{r_2}). Please notice that it is possible to be elected the same asset in both trials as it is not excluded from the selection process an asset that has been already selected in the first election. This is happening because each one of the two randomly elected assets will be allocated in a different solution vector.

Then is performed the simulated binary crossover (SBX) operator to the extracted $w_p^{(1)}$ and $w_p^{(2)}$ weights, following the typical SBX operator process as illustrated in Fig. 8.2. We also apply a constraint correction mechanism to make sure that both the budget constraint and the floor and ceiling constraints are satisfied [158]. After the completion of the correction process the $w_c^{(1)}$ and $w_c^{(2)}$ take their normalized value of $w_{c'}^{(1)}$ and $w_{c'}^{(2)}$.

Finally, the normalized $w_{c'}^{(1)}$ and $w_{c'}^{(2)}$ weights will be used to recompose the child decision variables as follows $r_c^{(1)} = w_{c'}^{(1)} + a_{r_1}$ and $r_c^{(2)} = w_{c'}^{(2)} + a_{r_2}$. As we can see, the originally selected assets have been replaced by the randomly generated a_{r_1} and a_{r_2} assets. This updating mechanism allows the replacement of some assets of the real-valued solution vector with some other assets from the available pool of securities. Given the fact that the available pool of assets (e,g, 1317 securities for port7 problem) is in most of the cases considerably larger than the K_{max}, this updating mechanism is particularly useful as it allows new assets to enter into the solution vector contributing to the variety of the solutions.

Recombination operator for the binary solution vector

For the binary solution vector we apply single-point crossover. In the single-point crossover one crossover point is randomly selected and then is copied everything before this point from parent 1 and then everything after the crossover point is copied from parent 2. Figure 3 provides an example of single-point crossover.



Parent 1	1	1	0	0	1	1	1	0
Parent 2	1	0	0	0	1	0	0	0
Offspring 1	1	1	0	0	1	0	0	0
Offspring 2	1	0	0	0	1	1	1	0

Fig.8.3. Single-point crossover

8.6. Experimental results for the constrained portfolio optimization problem

8.6.1 The test problems

In this section we run a number of computational experiments, as shown in Table 1, to test the performance of the proposed Efficiently Encoded Multiobjective Portfolio Optimization Solver (EEMPOS) for solving the constrained portfolio optimization problem (CPOP). The test problems range from 31 assets for the smaller instance and up to the 1317 assets for the bigger instance. The performance of the proposed algorithm is assessed in comparison with the True Efficient Frontier (TEF).

Table 8.1: The OR-library portfolio optimization problems

Problem Name	Stock Market Index	Assets
port1	Hang Seng	31
port2	DAX100	85
port3	FTSE100	89
port4	S&P100	98
port5	Nikkei225	225
port6	S&P500	457
port7	Russell 2000	1317

In order to calculate the TEF, we formulated the CPOP as a Mixed Integer Quadratic Program (MIQP) (see section 8.3) and we solve the problem by using CPLEX version 12.5. Table 8.2 indicates the average running time per *approximate efficient point* in the case of the EEMPOS and *exact efficient point* in the case of MIQP for port1-5 test instances. Please notice that for port6 and port7 test instances was not possible to calculate the TEF as for each exact efficient point the required computational time exceeded considerably the maximum allowed time frame (*1,000* seconds). Based on the above, the use of evolutionary algorithms is clearly justified as the exact approaches fail to produce solutions within a reasonable time frame especially for the bigger instances of the CPOP.

Table 8.2: Average runtime (in seconds) required per efficient point

Constrained Problem*	Running Time (in seconds) per	
	Approximate Efficient Point	Exact Efficient Point - MIQP
	EEMPOS	CPLEX
port1	1.24	15
port2	15	125
port3	17	148
port4	21	213
port5	45	521
port6	78	*unspecified > 1,000 secs*
port7	128	*unspecified > 1,000 secs*

* *Minimum cardinality of the portfolio K_{min} = 2 and maximum cardinality K_{max} = 10 for all test problems. Lower bound l_i = 0.01 and the upper bound u_i = 0.99, for each asset i, where i = 1, . . . , n.*

8.6.2 Parameter Setup

All algorithms have been implemented in Java and run on a 2.1GHz Windows Server 2012 machine with 6GB RAM. We set the minimum cardinality of the portfolio to two ($K_{min} = 2$) and the maximum cardinality of the portfolio to ten ($K_{max} = 10$) for all test problems, unless explicitly stated otherwise. The participation of each asset in the portfolio is determined by the lower and upper bounds. We set the lower bound $l_i = 0.01$ and the upper bound $u_i = 0.99$, for each asset i, where $i = 1, . . . , N$.

In all tests we use binary tournament, two-phase recombination operator (TPRO) and two-phase mutation operator (TPMO) as selection, crossover and mutation operator, respectively. The crossover probability is $P_c = 0.2$ and respectively the mutation probability is $P_m = 0.2$ for all test problems, for both the real and the binary solution vectors, unless explicitly stated otherwise. The distribution indices for the crossover and mutation operators are $\eta_c = 20$ and $\eta_m = 20$, respectively. We decided to use the aforementioned values for the crossover and mutation probability as the proposed representation scheme makes more efficient utilization of the mutation and the crossover operators than the conventional approaches with the Streichert et al. [200] encoding scheme. For more details please refer to section 4, where the proposed representation scheme is analyzed.

In the relevant studies in the field [5] , [6], in order to achieve good approximate solutions, the most common values for mutation and crossover probability with the Streichert et al. [200] representation scheme are *0.9* (or even *1.0*) and *0.9* respectively. Clearly, the higher level of mutation and crossover probability required for the configuration of the algorithm with the conventional Streichert et al. [200] encoding scheme, involves a computational cost in terms of required processing power and computational time.

We also tested the following configurations of the algorithm: N^{pop} = 100, 200, 300, 400, for 100, 250, 500 and 1000 generations respectively, for identifying the best population size. After examining the aforementioned configurations we reached the conclusion that the proposed algorithm approaches its top performance for N^{pop} = 100 and 1,000 generations or 100,000 function evaluations. Beyond, these values the gains in performance are marginal if any. Thus, the population size is set to *100*, using *100,000* function evaluations with *5* independent runs for all test problems, unless explicitly stated otherwise.

8.6.3 Performance Metrics

For the evaluation of the performance of the attained approximation sets the following three performance metrics are employed:

Hypervolume indicator (HV) [233]

According to Zitzler et al. [234] Hypervolume indicator is the only known unary indicator that can assess the quality of a solutions set by the single value of its hypervolume. According to the same study [234], hypervolume indicator is the only known indicator which is compliant with the concept of Pareto dominance. Thus, whenever a set of solutions is better than another set, its hypervolume indicator value is higher than the latter one.

Inverted Generational Distance (IGD)

The inverted generational distance (IGD) can be defined as follows:

$$\mathrm{IGD(P, S)} = \frac{\left(\sum_{i=1}^{|P|} d_i^q\right)^{1/q}}{|P|}$$

where $d_i = min_{\vec{s}\in S}||F(\vec{p}_i) - F(\vec{s}))||, \vec{p}_i \in P, q = 2 \; and \; d_i$ is the smallest distance of $\vec{p} \in P$ to the closest solutions in S. The smaller the IGD value the better is the performance of the approach. The IGD metric is able to provide a measure for both convergence and diversity.

Epsilon Indicator (I_ε)

There are two versions of epsilon indicator the multiplicative and the additive [229]. In this study we use the unary additive epsilon indicator. The epsilon indicator of an approximation set *A* ($I_{\varepsilon+}$) provides the minimum factor ε by which each point in the real front *R* can be added such that the resulting transformed approximation set is dominated by *A*. The additive epsilon indicator is a good measure of diversity, since it focuses on the worst case distance and reveals whether or not the approximation set has gaps in its trade-off solution set.

8.6.4 Experimental Results

Below we provide a number of computational experiments to test the performance of the proposed Efficiently Encoded Multiobjective Portfolio Optimization Solver (EEMPOS) for solving the constrained portfolio optimization problem (CPOP). The performance of the proposed algorithm is assessed in comparison with the True Efficient Frontier (TEF) for a number of test problems. In particular we use data sets from the publicly available OR-Library retained by Beasley. These data sets correspond to seven portfolio optimization problems from seven different capital markets as shown in Table 1. More specifically, we used weekly price data for the period March 1992 to September 1997, from Hang Seng 31 in Hong Kong, DAX 100 in Germany, FTSE 100 in UK, S&P 100 in USA, Nikkei 225 in Japan, S&P 500 in USA and Russell 2000 in USA.

Figure 8.4 presents the evolutionary process of Epsilon metric for the port1 test problem obtained by EEMPOS. Epsilon indicator takes its optimal value when zero. As shown by Fig. 8.4 as the number of function evaluations increases, the Epsilon metric decreases but with lower rate. The algorithm converges to its minimum Epsilon metric value when approximately *30,000* solutions are generated. Fig. 8.5 presents the evolutionary process of HV metric for the port1 test problem obtained by

EEMPOS. The results for HV metric are in consistence with the results for the Epsilon metric for the constrained port1 problem.

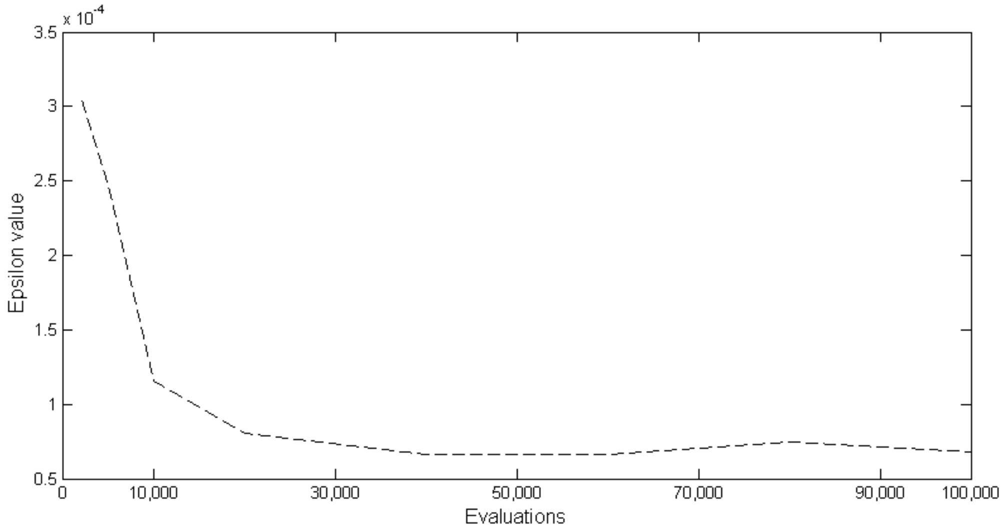


Fig. 8.4 The evolutionary process of Epsilon metric for the constrained port1 test problem obtained by EEMPOS

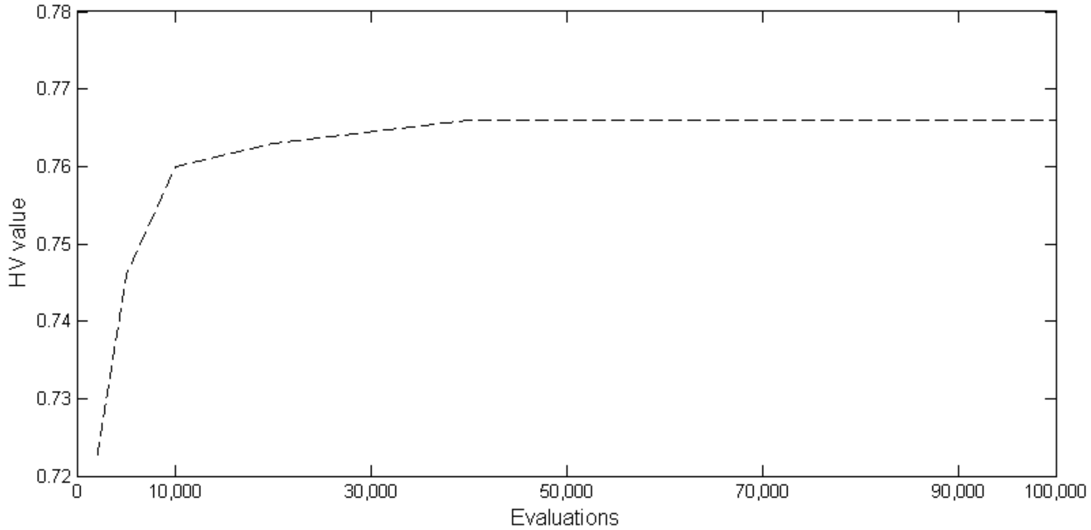


Fig. 8.5 The evolutionary process of HV metric for the constrained port1 test problem obtained by EEMPOS

Fig.8.6 - 8.10 illustrate the approximate efficient frontier of the EEMPOS along with the true efficient frontier (TEF) for the constrained port1-5 problems, after *100,000* evaluations. The TEF is calculated by formulating the constrained portfolio optimization problem (CPOP) as a mixed integer quadratic programming (MIQP) problem. As shown by Fig. 8.6 - 8.10 the proposed EEMPOS performs well both in

terms of convergence to the TEF and coverage of the efficient frontier for the constrained port1-5 problems.

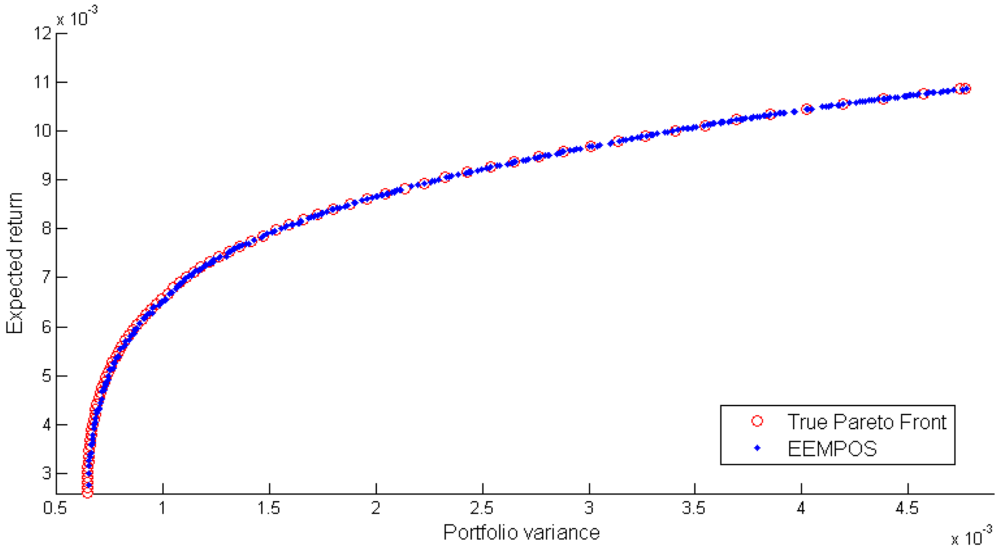


Fig. 8.6 The Mean–Variance efficient frontier for the constrained port1 problem under two different configurations

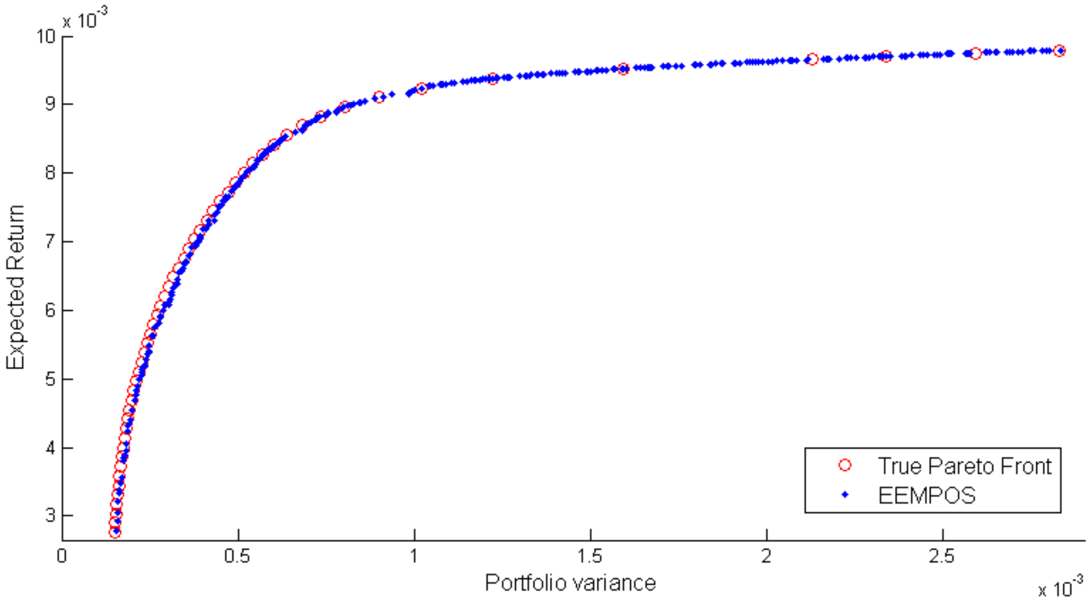


Fig. 8.7 The Mean–Variance efficient frontier for the constrained port2 problem under two different configurations

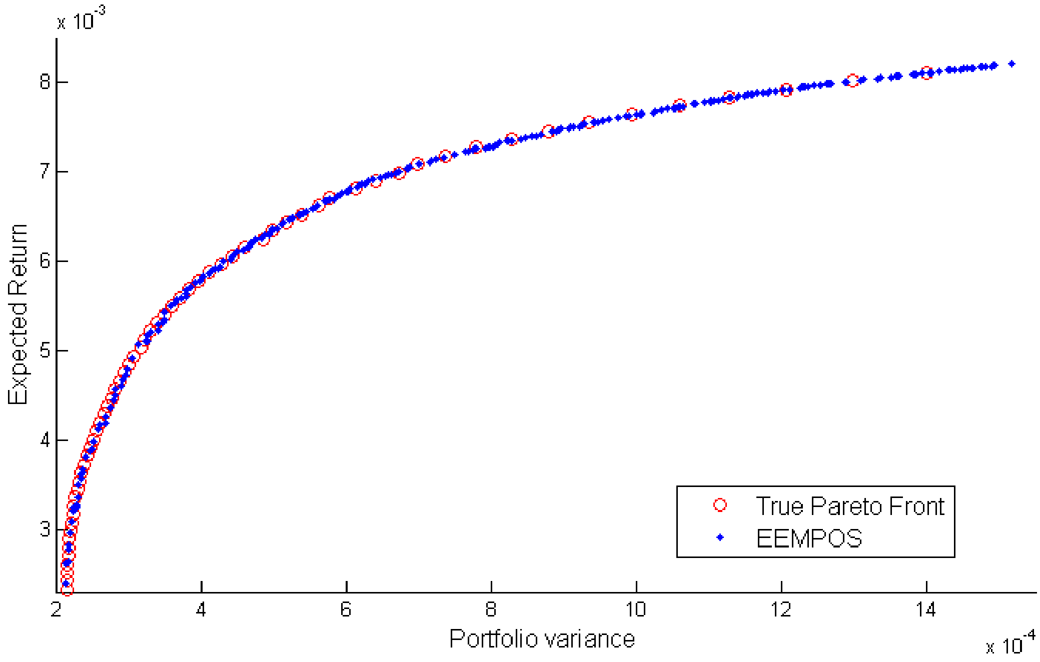


Fig. 8.8 The Mean–Variance efficient frontier for the constrained port3 problem under two different configurations

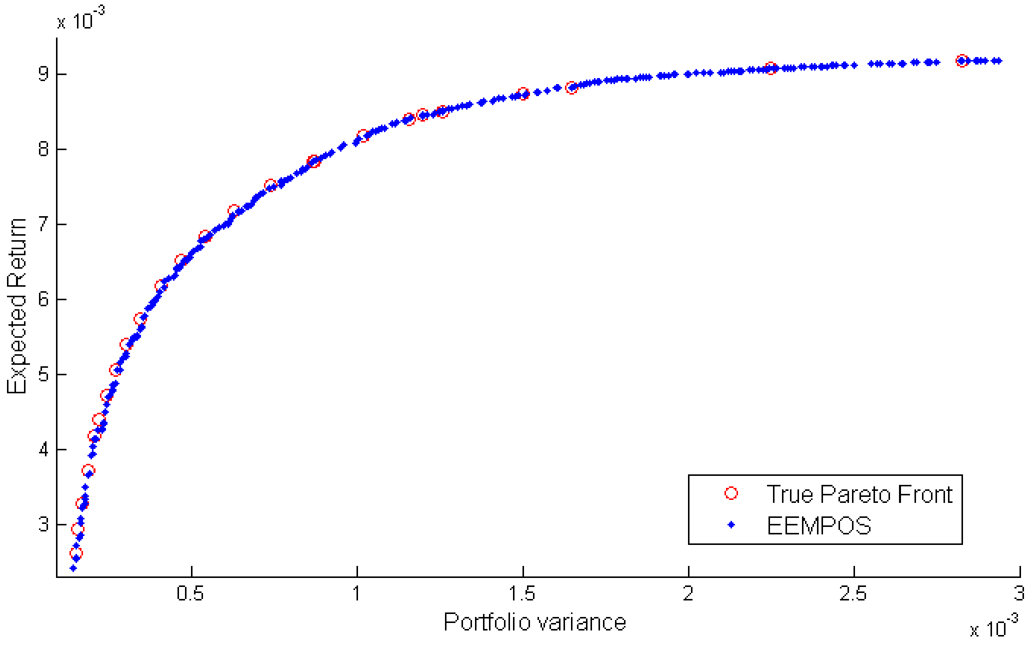


Fig. 8.9 The Mean–Variance efficient frontier for the constrained port4 problem under two different configurations

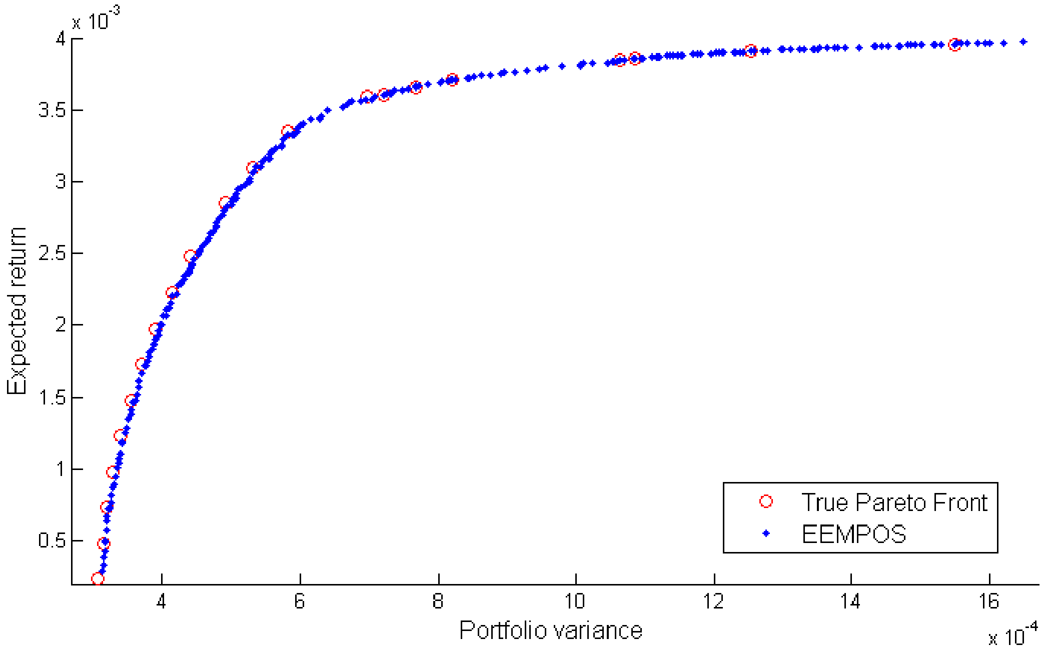


Fig. 8.10 The Mean–Variance efficient frontier for the constrained port5 problem under two different configurations

Fig.8.11 presents the evolutionary process of IGD metric for the port6 test problem obtained by EEMPOS. IGD indicator takes its optimal value when zero. As shown by Fig.8.11 as the number of function evaluations increases, the IGD metric decreases but with lower rate. The algorithm converges to its minimum IGM metric value when approximately *30,000* solutions are generated

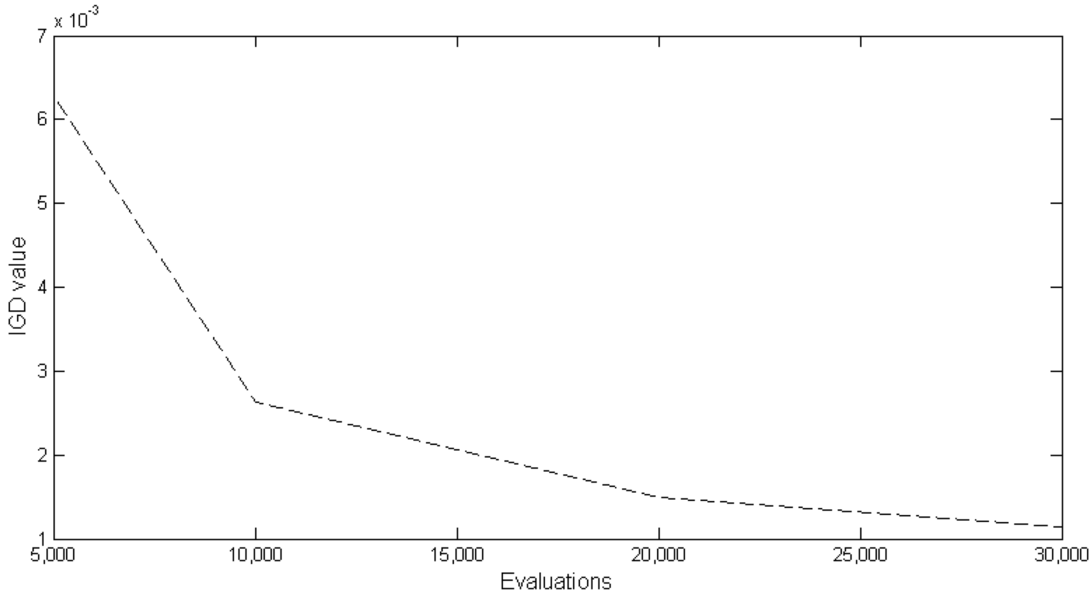


Fig. 8.11 The evolutionary process of IGD metric for the constrained port6 test problem obtained by EEMPOS

Fig. 8.12 illustrates the approximate efficient frontier of the EEMPOS for the constrained port6 problem after *30,000* evaluations. It was not possible to calculate the true efficient frontier (TEF) for port6 and 7 instances as for each exact efficient point the required computational time exceeded considerably the maximum allowed time frame (*1,000* seconds). As shown by Fig. 8.12 the proposed EEMPOS performs well in terms of coverage of the efficient frontier for the constrained port6 problem although only *30,000* evaluations were used.

Fig. 8.13 illustrates the approximate efficient frontier of the EEMPOS for the constrained port7 problem after *50,000* evaluations. As shown by Fig. 8.13 the proposed EEMPOS performs well in terms of coverage of the efficient frontier for the constrained port7 problem although only *50,000* evaluations were used.

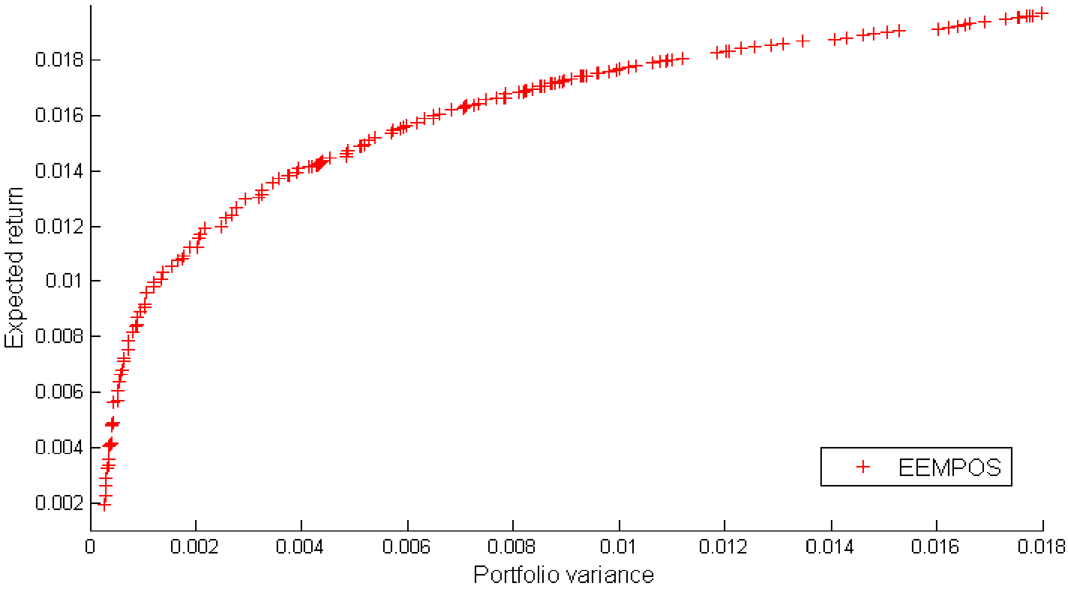


Fig. 8.12 The Mean–Variance efficient frontier for the constrained port6 problem obtained by EEMPOS

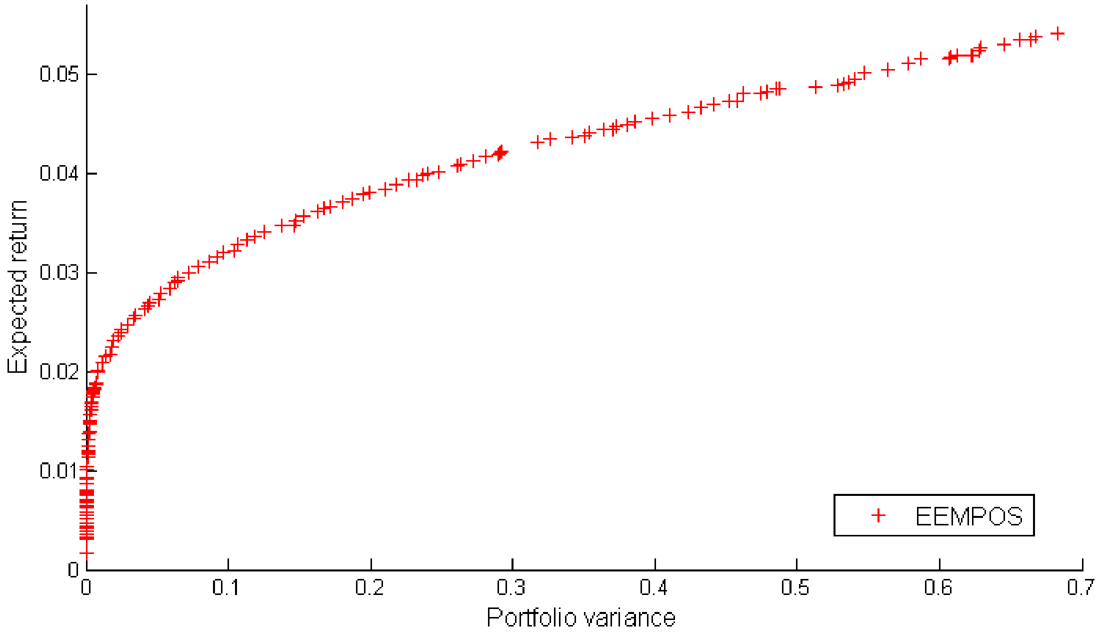


Fig. 8.13 Mean–Variance efficient frontier for the constrained port7 problem obtained by EEMPOS

Fig. 8.6 - 8.10 show that the EEMPOS have generated a large number of portfolios that lie on the true constrained efficient frontier or very close to it. Fig. 8.11 illustrates the evolutionary process of IGD metric for the constrained port6 test problem obtained by EEMPOS. The IGD indicator is used to measure how far the elements are in the Pareto optimal set from those in the set of non-dominated vectors found. A value of IGD equal zero indicates that all the generated elements are in the Pareto front and they cover all the extension of the Pareto front. As is demonstrated by the IGD values of Fig. 8.11 for the constrained port6 problem, as the algorithm reaches the approximately *30,000* solutions, the generated portfolios are very close to the optimal ones and provide a good coverage of the Pareto front. The relevant findings of IGD metric are verified by Fig. 8.12 that illustrates the Mean–Variance efficient frontier for the constrained port6 problem obtained by EEMPOS.

To conclude, experiments indicate that the proposed EEMPOS generates good quality approximate efficient frontiers that lie on the true constrained efficient frontier or very close to it, with good coverage of the entire range of the efficient frontier.

8.7 Conclusions

In this thesis we propose a novel multiobjective evolutionary Algorithm (MOEA) for the solution of the constrained portfolio optimization problem (CPOP) named efficiently encoded multiobjective portfolio optimization solver (EEMPOS). The proposed algorithm introduces an efficient representation scheme specially designed for dealing with the complexities of the CPOP. The proposed representation scheme is more cost-efficient in terms of computational resources and furthermore provides better utilization of the genetic operators, when compared with other approaches, as it is designed to focus the search effort to the most meaningful areas of the search space. The algorithm also incorporates a new mutation and recombination operator specially designed to work well with the proposed encoding scheme.

Datasets ranging from *31* assets for the smaller instance and up to *1317* assets for the bigger instance from seven different stock markets are utilized for testing the efficiency of the proposed approach. In particular, we formulate the constrained portfolio optimization problem (CPOP) as a Mixed Integer Quadratic Programming (MIQP) problem and we solve it for a number of test instances. Then, we compare the extracted true efficient frontiers (TEF) with the approximate efficient frontiers obtained by EEMPOS. The experiments indicate that the proposed EEMPOS generates good quality approximate efficient frontiers that lie on the true constrained efficient frontier or very close to it, with good coverage of the entire range of the efficient frontier in a fraction of time required by the exact approaches.

In our future work, we will attempt to develop a technique that will replace the mutation (P_m) and crossover probability (P_c) with a variable that will be updated at run-time according to the performance of the algorithm in a number of metrics such as the hypervolume or IGD metric. We also intend to experiment with alternative objectives and additional constraints.

Chapter 9

Conclusions

9.1 Summary of the Work Done

In this thesis we have presented novel multiobjective evolutionary algorithms (MOEAs) approaches with application in the constrained portfolio optimization problem (CPOP). The work done in this thesis can be summarized as follows:

Chapter 1 Provides an introduction to the constrained portfolio optimization problem alongside with the motivation and the objectives of this thesis.

Chapter 2 Briefly reviews the most well-known MOEAs alongside with their strengths and weaknesses.

Chapter 3 Briefly reviews the existing literature in the area of portfolio optimization with the support of MOEAs.

Chapter 4 Presents a methodological framework for conducting a comprehensive literature study based on the papers published in MOEAs for the Portfolio Management over a long time span. This framework is being used to gain an understanding of the current state of the MOEAs for the Portfolio Management research field.

Chapter 5 Based on the examination of the state-of-the art, presents the best practices from a technical and algorithmic point of view for dealing with the complexities of the constrained portfolio optimization problem.

Chapter 6 Proposes a new probe guided version of the polynomial mutation (PLM) operator. The experimental results reveal that the proposed Probe Guided Mutation (PGM) operator outperforms with confidence the performance of the classical PLM operator for all performance metrics when applied to the solution of the cardinality constrained portfolio optimization problem (CCPOP), but also to Zitzler-Deb-Theile (ZDT) [229] and Deb-Theile-Laumanns- Zitzler (DTLZ) [60] families of test functions.

Chapter 7 Proposes a new Two Stage Crossover (TSX) operator for more efficient exploration of the search space. The performance of the proposed TSX operator is assessed in comparison with the Simulated Binary Crossover (SBX) operator with the assistance of three well-known MOEAs, namely the NSGAII, the SPEA2 and the MOCELL, for the solution of the DTLZ1-7 set of test functions [60]. We also compare the proposed TSX operator with other popular reproduction operators like the Differential Evolution (DE) and the Particle Swarm Optimization (PSO). It is shown with the assistance of the DTLZ set of test functions that the TSX operator can substantially improve the results generated by three popular performance metrics for most of the cases.

Chapter 8 Introduces a new multiobjective evolutionary algorithm (MOEA), called Efficiently Encoded Multiobjective Portfolio Optimization Solver (EEMPOS) that incorporates an efficient encoding scheme and specially designed genetic operators for the solution of the constrained portfolio optimization problem. In order to evaluate the performance of the proposed EEMPOS, we calculate the True Efficient Frontier (TEF) by formulating the constrained portfolio optimization problem as a Mixed Integer Quadratic Program (MIQP) and we compare the relevant results with the approximate efficient frontiers that are derived by the proposed EEMPOS. The relevant results verify that the EEMPOS generates solutions that lie on the TEF for a fraction of the time required by the MIQP.

Chapter 9 This dissertation is concluded with a summary of the research and proposed directions for future study.

9.2 Summary of the Contribution

The original contributions of this research are summarized in the following:

- We develop a methodological framework for conducting a comprehensive literature study based on the papers published in MOEAs for the Portfolio Management over a long time span across various disciplines.
- This framework is being used to gain an understanding of the current state of the MOEAs for the Portfolio Management research field.
- Based on the literature study, we identify potential areas of concern in regard to MOEAs for the Portfolio Management.
- Based on the examination of the state-of-the art we present the best practices from a technical and algorithmic point of view for dealing with the complexities of the constrained portfolio optimization problem
- A new probe guided version of the well-known polynomial mutation operator is being proposed that enhance considerably algorithms' performance.
- We propose a new Two Stage Crossover (TSX) operator for more efficient exploration of the search space.
- This research introduces a new multiobjective evolutionary algorithm (MOEA), called Efficiently Encoded Multiobjective Portfolio Optimization Solver (EEMPOS) that incorporates an efficient encoding scheme and specially designed genetic operators for the solution of the constrained portfolio optimization.

9.3 Future Work

The work of this thesis has been concerned with the application of multiobjective evolutionary algorithms (MOEAs) to the solution of the constrained portfolio optimization problem (CPOP). In this thesis, we decided to tackle the portfolio optimization problem as a bi-objective problem. We believe that the examination of alternative objectives beyond the mean – variance framework, could lead to interesting findings that should be thoroughly studied. Also, it would be interesting to be examined the application of more than two objectives on the efficient frontier formulation. Finally, it is worthwhile the incorporation of additional real-world constraints into the examined model.

References

[1] Albayrak, M. & Allahverdi, N. (2011). Development a new mutation operator to solve the Traveling Salesman Problem by aid of Genetic Algorithms. Expert Systems with Applications, Volume 38, Issue 3, March 2011, Pages 1313-1320.

[2] Alguliev, R.M., Aliguliyev,R.M. & Isazade, N.R. (2012). DESAMC+DocSum: Differential evolution with self-adaptive mutation and crossover parameters for multi-document summarization, Knowledge-Based Systems 36 (2012) 21–38.

[3] Amjady N, Nasiri-Rad H.(2009) Nonconvex economic dispatch with AC constraints by a new real coded genetic algorithm. IEEE Trans Power Syst 2009; 24(3):1489–502.

[4] Anagnostopoulos, K.P. and Mamanis, G. (2011), Multiobjective evolutionary algorithms for complex portfolio optimization problems, Comput Manag Sci (2011) 8:259–279.

[5] Anagnostopoulos K. and Mamanis G. (2010) A portfolio optimization model with three objectives and discrete variables, Computers & Operations Research 37 (2010): 1285 – 1297.

[6] Anagnostopoulos, K.P. and Mamanis, G. (2011), The mean–variance cardinality constrained portfolio optimization problem: An experimental evaluation of five multiobjective evolutionary algorithms, Expert Systems with Applications 38 (2011) 14208–14217.

[7] Andriosopoulos, K. and Nomikos, N. (2014), Performance replication of the Spot Energy Index with optimal equity portfolio selection: Evidence from the UK, US and Brazilian markets, European Journal of Operational Research 234 (2014) 571–582.

[8] Armananzas R. and Lozano J. A., (2005) A Multiobjective Approach to the Portfolio Optimization Problem, in 2005 IEEE Congress on Evolutionary Computation (CEC'2005), vol. 2. Edinburgh, Scotland: IEEE Service Center, September 2005, pp. 1388–1395.

[9] Arnone S., Loraschi A., and Tettamanzi A., A genetic approach to portfolio selection. Neural Network World – International Journal on Neural and Mass-Parallel Computing and Information Systems, 3(6):597–604, 1993.

[10] Arora, S. & Arora, S.R.(2010). Multiobjective capacitated plant location problem. Int. J. of Operational Research 2010 - Vol. 7, No.4 pp. 487 – 505.

[11] Artzner, P., Delbaen, F., Eber, J., & Heath, D. (1999). Coherent Measures of Risk. Mathematical Finance, Vol. 9, pp.203-228.

[12] Babaei, S., Mehdi Sepehri, M. and Babaei, E. (2015) Multi-objective portfolio optimization considering the dependence structure of asset returns, European Journal of Operational Research 244 (2015) 525–539.

[13] Balzer, L. A. (1994). Measuring investment risk. The Journal of Investing, 3(3):47-58.

[14] Banerjee, A. (2013) A novel probabilistically-guided context-sensitive crossover operator for clustering, Swarm and Evolutionary Computation 13(2013)47–62.

[15] Banzi, A.S., Nobre, T., Pinheiro, G.B. Arias, J.C., Pozo, A. & Vergilio, S.R. (2012). Selecting mutation operators with a multiobjective approach. Expert Systems with Applications, Volume 39, Issue 15, 1 November 2012, Pages 12131-12142.

[16] Barkat-Ullah ASSM, Sarker R, Cornforth D. (2008) Search space reduction technique for constrained optimization with tiny feasible space. In: GECCO'08, Atlanta, GA, USA, 2008. p. 881–8.

[17] Bertsimas, D., Shioda, R. (2009) Algorithm for cardinality-constrained quadratic optimization, Computational Optimization and Applications, 43, 1–22.

[18] Beume, N., Naujoks, B. and Emmerich, M. (2007) SMS-EMOA: Multiobjective selection based on dominated hypervolume, European Journal of Operational Research, Volume 181, Issue 3, 16 September 2007, Pages 1653–1669.

[19] Beyer, H.G., Finck, S. and Breuer, T. (2014) Evolutionon trees:On the design of an evolution strategy for scenario-based multi-period portfolio optimization under transaction costs, Swarm and Evolutionary Computation 17(2014)74–87.

[20] Bienstock, D. (1996) Computational study of a family of mixed-integer quadratic pro-gramming problems, Math. Program. 74 (2) (1996) 121–140.

[21] Bloomfield, T., Leftwich, R. And Long, J. (1977) Portfolio Strategies and Performance, Journal of Financial Economics, 5, 201-218.

[22] Boonma, P. & Suzuki, J. A. (2009). Confidence-based Dominance Operator in Evolutionary Algorithms for Noisy Multiobjective Optimization Problems. Proceedings of the 21st IEEE Conf. on Tools with Artificial Intelligence, pages 387-394.

[23] Bosman, P. A. N. and Thierens, D. (2000), Expanding from Discrete to Continuous EDAs: The IEDA, Proceedings of Parallel Problem Solving from Nature, PPSN-VI, pp. 767-776.

[24] Branke, J., Scheckenbach, B., Stein, M., Deb, K. and Schmeck, (2009) H. Portfolio optimization with an envelope-based multi-objective evolutionary algorithm, European Journal of Operational Research 199 (2009) 684–693.

[25] Cannon, W. D. (1932), The Wisdom of the Body, Norton and Company, New York, NY.

[26] Carazoa, A.F., Gomez, T., Molina, J., Diaz, A.G.H., Guerrero, F.M. and Caballero, R. (2010) Solving a comprehensive model for multiobjective project portfolio selection, Computers & Operations Research 37 (2010) 630—639.

[27] Cesarone, F., Scozzari, A. & Tardella, F. (2009) Effficient algorithms for mean-variance portfolio optimization with hard real-world constraints, Giornale dell' Istituto Italiano degli Attuari, volume LXXII, Roma, (2009). 37-56.

[28] Chang, T.J., Meade, N., Beasley, J.E. and Sharaiha, Y.M. (2000) Heuristics for cardinality constrained portfolio optimisation. Computers & Operations Research, 27(13): (2000) 1271-1302.

[29] Chang, J.F., Shi, P. (2011) Using investment satisfaction capability index based particle swarm optimization to construct a stock portfolio, Information Sciences 181 (2011) 2989–2999.

[30] Chang, T.J., Yang, S.C. and Chang, K.J. (2009) Portfolio optimization problems in different risk measures using genetic algorithm, Expert Systems with Applications 36 (2009) 10529–10537.

[31] Chen C., Kwon, R.H. (2010) Robust portfolio selection for index tracking, Computers & Operations Research 2010.

[32] Chen, Y., Mabu, S. & Hirasawa, K. (2011) General relation algorithm with guided mutation for the large-scale portfolio optimization, Expert Systems with Applications 38 (2011): 3353-3363.

[33] Chen, Y. and Wang, X. (2015), A hybrid stock trading system using genetic network programming and mean conditional value-at-risk, European Journal of Operational Research (2015) 240(3):861–871. DOI: 10.1016/j.ejor.2014.07.034.

[34] Chiam, S.C., Al Mamun A., & Low, Y.L., (2007) A realistic approach to evolutionary multiobjective portfolio optimization, IEEE Congress on Evolutionary Cmputation (CEC 2007).

[35] Chiam, S.C., Tan, K.C. and Al Mamum, A. (2008) Evolutionary Multi-objective Portfolio Optimization in Practical Context, International Journal of Automation and Computing, 05(1), January 2008, 67-80.

[36] Chiam, S.C., Tan, K.C. and Al. Mamun, A. (2009) A memetic model of evolutionary PSO for computational finance applications, Expert Systems with Applications 36 (2009) 3695–3711.

[37] Chiam, S.C., Tan, K.C. and Al Mamun, A. (2009) Investigating technical trading strategy via an multi-objective evolutionary platform, Expert Systems with Applications 36 (2009) 10408–10423.

[38] Claro J. and Sousa. J. P. (2010) A multiobjective metaheuristic for a mean-risk multistage capacity investment problem with process flexibility, Computers & Operations Research 2010.

[39] Coello Coello, C. A. (1996). An Empirical Study of Evolutionary Techniques for Multiobjective Optimization in Engineering Design. Unpublished doctoral dissertation, Department of Computer Science, Tulane University, New Orleans, Louisiana, USA.

[40] Coello Coello, C.A. (2006), Evolutionary multi-objective optimization and its use in finance. In J.-P. Rennard (Ed.), Evolutionary Computation Group, Mexico.

[41] Coello Coello, C. A., & Toscano Pulido, G. (2001). Multiobjective Optimization using a Micro-Genetic Algorithm. In L. Spector et al. (Eds.), Proceedings of the genetic and evolutionary computation conference (gecco'2001) (pp. 274–282). San Francisco, California: Morgan Kaufmann Publishers.

[42] Coello Coello, C. A., Van Veldhuizen, D. A., & Lamont, G. B. (2002). Evolutionary Algorithms for Solving Multi-Objective Problems. New York: Kluwer Academic Publishers. (ISBN 0-3064-6762-3).

[43] Corazza, M., Fasano, G. and Gusso, R. (2013), Particle Swarm Optimization with non-smooth penalty reformulation, for a complex portfolio selection problem, Applied Mathematics and Computation 224 (2013) 611–624.

[44] Corne, D. W., Jerram, N. R., Knowles, J. D., & Oates, M. J.(2001). PESA-II: Region-based Selection in Evolutionary Multiobjective Optimization. In L. Spector et al. (Eds.), Proceedings of the genetic and evolutionary computation conference (gecco'2001) (pp. 283–290). San Francisco, California: Morgan Kaufmann Publishers.

[45] Corne, D. W., Knowles, J. D., & Oates, M. J. (2000). The Pareto Envelope-based Selection Algorithm for Multiobjective Optimization. In M. Schoenauer et al. (Eds.), Proceedings of the parallel problem solving from nature vi conference (pp. 839–848). Paris, France: Springer. Lecture Notes in Computer Science No. 1917.

[46] Crama, Y. & Schyns, M. (2003): Simulated annealing for complex portfolio selection problems, European Journal of Operational Research 150: 546-571.

[47] Cura, T. (2009) Particle swarm optimization approach to portfolio optimization, Nonlinear Analysis: Real World Applications, 10, 2396–2406.

[48] Da Ronco, C.C. & Benini, E., (2013), A Simplex Crossover based evolutionary algorithm including the genetic diversity as objective. Applied Soft Computing 13 (2013) 2104–2123.

[49] Dang, J., Brabazon, A., Edelman, D. and O'Neill, M. (2009) An Introduction to Natural Computing in Finance, Applications of Evolutionary Computing - Lecture Notes in Computer Science, 2009, Volume 5484/2009, 182-192.

[50] Das, S., Mallipeddi, R., & Maity, D. (2013). Adaptive evolutionary programming with p-best mutation strategy Swarm and Evolutionary Computation 9 (2013) 58–68.

[51] Deb, K. (1998). Multiobjective genetic algorithms: Problem difficulties and Construction of test problems. Technical report, Indian Institute of Technology, Kanpur.

[52] Deb, K. (2000) An efficient constraint handling method for genetic algorithms, Comput. Methods Appl. Mech. Engrg. 186 (2000) 311-338.

[53] Deb, K. (2001) Multiobjective optimization using evolutionary algorithms. (2001) Wiley, Chichester, UK.

[54] Deb, K. and Agrawal, R.B. (1995) Simulated binary crossover for continuous search space, Complex Systems 9 (2), (1995) 115–148.

[55] Deb, K., Anand, A. and Joshi, D. (2002) A Computationally Efficient Evolutionary Algorithm for Real-Parameter Optimization, Evolutionary Computation 10(4): 371-395

[56] Deb, K. & Goyal, M. (1996). A combined genetic adaptive search (geneas) for engineering design, Computer Science and Informatics 26 (4) 30–45.

[57] Deb, K, Pratap, A., Agarwal, S. and Meyarivan, T. (2002) A Fast and Elitist Multi Objective Genetic Algorithm: NSGA-II, IEEE Transactions on Evolutionary Computation, Vol. 6. No. 2, 2002, pp. 182-197.

[58] Deb, K. & Saxena, D.K. On Finding Pareto-Optimal Solutions Through Dimensionality Reduction for Certain Large-Dimensional Multi-Objective Optimization Problems, KanGAL Report Number 2005011.

[59] Deb, K., Steuer, R.E., Tewari, R. & Tewari, R. (2011) Bi-objective portfolio optimization using a customized hybrid NSGA-II procedure, R.H.C. Tekahashi et al. (Eds.): EMO 2011, LNCS 6576, pp. 358-373, Springer-Verlag Berlin Heidelberg 2011.

[60] Deb K, Thiele L, Laumanns M, Zitzler E. (2005), Scalable test problems for evolutionary multiobjective optimization. In: Abraham A, Jain L, Goldberg R, editors. Evolutionary multiobjective optimization. theoretical advances and applications. USA: Springer; 2005. p. 105–45.

[61] Deb, K. and Tiwari, S. (2008) Omni-Optimizer: A Generic Evolutionary Algorithm for Single and Multi-Objective Optimization. European Journal of Operational Research, Vol. 185.

[62] Deep, K. and Thakur, M. A new crossover operator for real coded genetic algorithms, *Applied Mathematics and Computation*, vol.188, pp.895-911, 2007.

[63] Deng, G.F & Lin W.T. (2010) Swarm intelligence for cardinality - constrained portfolio problems, J.-S. Pan, S.-M. Chen and N.T. Nguyen (Eds): ICCCI 2010, Part III, LNAI 6423, pp. 406-415, Springer – Verlag Berlin Heidelberg 2010.

[64] Deng, G.F & Lin W.T. (2010) Ant colony optimization for Markowitz mean-variance portfolio model, B.K. Panigrahi et al. (Eds): SEMCCO 2010, LNCS 6466, pp.238-245, Springer – Verlag Berlin Heidelberg 2010.

[65] Derigs, U., Nickel, N.H. (2003) Meta-heuristic based decision support for portfolio optimization with a case study on tracking error minimization in passive portfolio management, OR Spectrum.

[66] Derrac, J., García, S., Molina, D. and Herrera, F. "A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms," *Swarm Evol. Comput.*, vol. 1, no. 1, pp. 3–18, Mar. 2011.

[67] Di Gaspero, L., Di Tollo, G., Roli, A. and Schaerf, A. (2007) Hybrid Local Search for Constrained Financial Portfolio Selection Problems, Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems - Lecture Notes in Computer Science, 2007, Volume 4510/2007, 44-58.

[68] Di Gaspero, L., Di Tollo, G., Roli, A. and Schaerf, A. (2007) A hybrid solver for constrained portfolio selection problems. Proceedings of Learning and Intelligent Optimization, LION, 2007.

[69] Di Tollo, G., Rolli, A.(2008) Metaheuristics for the Portfolio Selection Problem, International Journal of Operations Research Vol. 5, No. 1, 13-35 (2008).

[70] Doerner K., Gutjahr W., Hartl R., Strauss C., Stummer C., (2004) Pareto Ant Colony Optimization: A Metaheuristic Approach to Multiobjective Portfolio Selection, Annals of Operations Research 131, 79-99, 2004.

[71] Doerner K., Gutjahr W., Hartl R., Strauss C., Stummer C., (2006) Pareto ant colony optimization with ILP preprocessing in multiobjective project portfolio selection, European Journal of Operational Research - Volume 171, Issue 3, 16 June 2006, Pages 830-841.

[72] Dominguez-Jimenez, J.J., Estero-Botaro, A., Garcia-Dominguez, A. & Medina-Bulo, I. (2011) Evolutionary mutation testing. Information and Software Technology, Volume 53, Issue 10, October 2011, Pages 1108-1123.

[73] Drezewski, R., Obrocki, K. and Siwik, L. (2009) Comparison of Multi-agent Co-operative Co-evolutionary and Evolutionary Algorithms for Multi-objective Portfolio Optimization, M. Giacobini et al. (Eds.): EvoWorkshops 2009, LNCS 5484, pp. 223–232, 2009. Springer-Verlag Berlin Heidelberg 2009.

[74] Drezewski, R., Sepielak, J. and Siwik, L. (2008) Generating Robust Investment Strategies with Agent-Based Co-evolutionary System, M. Bubak et al. (Eds.): ICCS 2008, Part III, LNCS 5103, pp. 664–673, 2008. Springer-Verlag Berlin Heidelberg 2008.

[75] Dueck & Winker (1992) New concepts and algorithms for portfolio choice, Applied Stochastic Models and Data Analysis, 8, 159-178.

[76] Duran, F.C., Cotta, C. and Fernandez, A.J. (2009) Evolutionary Optimization for Multiobjective Portfolio Selection Under Markowitz's Model with Application to the Caracas Stock Exchange Nature-Inspired Algorithms for Optimisation, Studies in Computational Intelligence Volume 193, 2009, pp 489-509, Springer Berlin Heidelberg.

[77] Durillo, J. J. & Nebro, A. J. (2011). jMetal: A Java framework for multi-objective optimization, Advances in Engineering Software 42, 760–771.

[78] Ehrgott, M., Klamroth, K. and Schwehm, C. (2004) An MCDM approach to portfolio optimization, European Journal of Operational Research 155 (2004) 752–770.

[79] Elazouni, A., Abido, M. (2011) Multiobjective evolutionary finance-based scheduling: Individual projects within a portfolio, Automation in Construction.

[80] Elsayed, S.M., Sarker, R.A. & Essam, D.L. (2011) Multi-operator based evolutionary algorithms for solving constrained optimization problems, Computers & Operations Research 38 (2011) 1877–1896.

[81] Eshelman L.J. & Schaffer J.D. (1993). Real Coded Genetic Algorithms and Interval Schemata. Foundation of Genetic Algorithms 2, L.Darrell Whitley (Ed.) Morgan Kaufmann Publishers, San Mateo, 187–202.

[82] Emmerich, M., Beume, N. & Naujoks, B. (2005). An EMO Algorithm Using the Hypervolume Measure as Selection Criterion. In Conference on Evolutionary Multi-Criterion Optimization (EMO 2005), pages 62–76. Springer.

[83] Erickson, M., Mayer, A. and Horn, J. (2001) The Niched Pareto Genetic Algorithm 2 Applied to the Design of Groundwater Remediation Systems , Lecture Notes in Computer Science, 2001, Volume 1993/2001, 681-695.

[84] Evans, J. L., and Archer, S.H. (1968) "Diversification and the Reduction of Dispersion: An Empirical Analysis." Journal of Finance, 23, Dec. 1968, 761-767.

[85] Fernandez, A., Gomez, S. (2007) Portfolio selection using neural networks, Computers & Operations Research 34 (2007) 1177–1191.

[86] Fernandez, E., Lopez, E., Mazcorro, G., Olmedo, R. and Coello Coello, C.A. (2013) Application of the non-outranked sorting genetic algorithm to public project portfolio selection, Information Sciences 228 (2013) 131–149.

[87] Fieldsend, J.E., Everson, R.M. and Singh, S. (2003) Using Unconstrained Elite Archives for Multiobjective Optimization, IEEE Transactions on Evolutionary Computation, vol. 7, no. 3, pp. 305–323, June 2003.

[88] Fieldsend JE, Matatko J, Peng M. (2004) Cardinality constrained portfolio optimisation, 5th International Conference on Intelligent Data Engineering and Automated Learning (IDEAL 2004), Execter, England, 25th - 27th Aug 2004, Intelligent Data Engineering and Automated Learning Ideal 2004, Proceedings, volume 3177, pages 788-793.

[89] Fisher, L. and Lorie, J.H. (1970) Some studies of Variability of Returns on Investments in Common stocks, Journal of Business, 43, 2 April, 99-134.

[90] Fleischer, M. (2003). The measure of Pareto optima: Applications to multi-objective metaheuristics, in Evolutionary Multiobjective Optimization. ser. Lecture Notes in Computer Science, C. M. Fonseca et al., Eds. Berlin, Germany: Springer-Verlag, vol. 2632, pp. 519–533.

[91] Fogel, L. J. (1966), Artificial Intelligence through Simulated Evolution, John Wiley, New York, NY.

[92] Fogel, L. J. (1999), Artificial Intelligence through Simulated Evolution. Forty Years of Evolutionary Programming, John Wiley & Sons, New York, NY.

[93] Fonseca, C. and Fleming, P. (1993) Genetic algorithms for multiobjective optimization: formulation, discussion and generalization, Proc. 5th Inter. Conf. on Genetic Algorithms, (1993) pp. 416–423.

[94] Fonseca C, Flemming P., Multiobjective optimization and multiple constraint handling with evolutionary algorithms - part ii: Application example. *IEEE Transactions on System, Man, and Cybernetics*, 28:38–47, 1998.

[95] Fu, T.C., Chung, C.P.and Chung, F.L. (2013), Adopting genetic algorithms for technical analysis and portfolio management, Computers and Mathematics with Applications 66 (2013) 1743–1757.

[96] Gabrel, V., Murat, C. and Thiele, A. (2014), Recent advances in robust optimization: An overview, European Journal of Operational Research 235 (2014) 471–483.

[97] [86] Gandibleux, X., Ehrgott, M. (2005) 1984-2004 20 Years of Multiobjective Metaheuristics. But What About the Solution of Combinatorial Problems with Multiple Objectives? Evolutionary Multi-Criterion Optimization - Lecture Notes in Computer Science, 2005, Volume 3410/2005, 33-46.

[98] Geem Z.W., Kim, JH, Loganathan, G.V. (2001) A new heuristic optimization algorithm: harmony search. Simulation 2001;76(2):60–8.

[99] Gilli, M., Kellezi, E. and Hysi, H. (2006) A data-driven optimization heuristic for downside risk minimization, Journal of Risk, 8, 1–18.

[100] Goldberg, D. E.(1989), Genetic Algorithms in Search, Optimization and Machine Learning. Reading, MA: Addison-Wesley Publishing Company.

[101] Goldberg, D.E. and Deb, K. (1991) A Comparative Analysis of Selection Schemes Used in Genetic Algorithms. In: G.J.E. Rawlins (Ed), Foundations of Genetic Algorithms, Morgan Kaufmann, Los Altos, 1991, 69 – 93.

[102] Golmakani, H. R. & Fazel, M. (2011) Constrained portfolio selection using particle swarm optimization, Expert Systems with Applications 38 (2011): 8327-8335.

[103] Gong, W., Cai, Z. & Liang, D. (2014) Engineering optimization by means of an improved constrained differential evolution. Comput. Methods Appl. Mech. Engrg. 268 (2014) 884–904.

[104] Gorgulho, A., Neves, R. and Horta, N. (2011) Applying a GA kernel on optimizing technical analysis rules for stock picking and portfolio composition, Expert Systems with Applications 38 (2011) 14072–14085.

[105] Gupta, P., Inuiguchi, M., Mehlawat, M.K. and Mittal, G. (2013) Multiobjective credibilistic portfolio selection model with fuzzy chance-constraints, Information Sciences 229 (2013) 1–17.

[106] Hanne, T. (2007) A multiobjective evolutionary algorithm for approximating the efficient set, European Journal of Operational Research, 176, 1723–1734.

[107] Hansen, M.P. and Jaszkiewicz, A. (1998) Evaluating the quality of approximations to the nondominated set, Technical Report IMM-REP-1998-7, Institute of Mathematical Modeling, Technical University of Denmark, 1998.

[108] Harik, G., Cantu -Paz, E., Goldberg, D.E. and Miller, B.L. (1999) The Gambler's ruin problem, genetic algorithms, and the sizing of populations, Evol. Comput., vol. 7, no. 3, pp. 231–253, 1999.

[109] Hassanzadeh, F., Nemati, H. and Sun, M. (2014) Robust optimization for interactive multiobjective programming with imprecise information applied to R&D project portfolio selection, European Journal of Operational Research 238 (2014) 41–53.

[110] Haubelt, C., Mostaghim, S., Teich, J. and Tyagi, A. (2003) Solving Hierarchical Optimization Problems Using MOEAs, In Evolutionary Multi-Criterion Optimization - In Lecture Notes in Computer Science (LNCS), Volume 2632.

[111] Herbawi, W. & Weber, M. (2011). Evolutionary Multiobjective Route Planning in Dynamic Multi-hop Ridesharing.Evolutionary Computation in Combinatorial Optimization Lecture Notes in Computer Science, Volume 6622/2011, 84-95.

[112] Hervás-Martínez, C., Ortiz-Boyer, D. and García-Pedrajas, N. Theoretical Analysis of the Confidence Interval Based Crossover for Real-Coded Genetic Algorithms, J.J. Merelo Guervós et al. (Eds.): PPSN VII, LNCS 2439, pp. 153–161, 2002. Springer-Verlag Berlin Heidelberg 2002.

[113] Hirschberger, M., Qi, Y., Steuer, R.E. (2010) Large-scale MV efficient frontier computation via a procedure of parametric quadratic programming, European Journal of Operational Research, 204, 581–588.

[114] Hochreiter R. (2007) A evolutionary computation approach to scenario risk-return portfolio optimization for general risk measures, Dept. of Statistics and Decision Support Systems, Univ. of Vienna.

[115] Holland, J. H. (1962), Outline for a logical theory of adaptive systems', Journal of the Association for Computing Machinery 9, 297– 314.

[116] Horn J., Nafpliotis N., Goldberg D.E. (1994), A niched Pareto genetic algorithm for multiobjective optimization. In: Proceedings of the First IEEE Conference on Evolutionary Computation (ICEC '94), Piscataway, NJ: IEEE Service Center; 1994. p. 82–87.

[117] Huband, S., Hingston, P., While, L. & Barone, L. (2003). An evolution strategy with probabilistic mutation for multi-objective optimization, in Proc. Congr. Evol. Comput., vol. 4, pp. 2284–2291.

[118] Jiang, Z.Z., Ip, W.H., Lau, H.C.W. and Fan, Z.P. (2011) Multi-objective optimization matching for one-shot multi-attribute exchanges with quantity discounts in E-brokerage, Expert Systems with Applications 38 (2011) 4169–4180.

[119] Jobst, N.J., Horniman, M.D., Lucas, C.A. and Mitra, G. (2001) Computational aspects of alternative portfolio selection models in the presence of discrete asset choice constraints, QUANTITATIVE FINANCE VOLUME 1 (2001) 1–13.

[120] Kaucic, M. (2010) Investment using evolutionary learning methods and technical rules, European Journal of Operational Research 207 (2010) 1717–1727.

[121] Kellerer, H. and Maringer, D. (2001) Optimization of Cardinality Constrained Portfolios with an Hybrid Local Search Algorithm, MIC'2001 - 4th Metaheuristics International Conference, Porto, Portugal, July 16-20, 2001.

[122] Kennedy, J. and Eberhart, R. C. A discrete binary version of the particle swarm algorithm. Proceedings of the World Multiconference on Systemics,Cybernetics and Informatics 1997, Piscataway, NJ. pp. 4104-4109, 1997.

[123] Khalili-Damghani, K., Sadi-Nezhad, S., Lotfi, F.H. and Tavana, M.(2013) A hybrid fuzzy rule-based multi-criteria framework for sustainable project portfolio selection, Information Sciences 220 (2013) 442–462.

[124] Kibzun, A. I. and Kuznetsov, E. A. (2006). Analysis of criteria VaR and CVaR. Journal of Banking & Finance, 30(2):779-796.

[125] Knowles, J. and Corne, D. (1999) Approximating the non-dominated front using the Pareto Archieved Evolution Strategy, Dept. of Computer Science, University of Reading, UK, 1999.

[126] Knowles, J., Thiele, L. & Zitzler. E. (2006). A Tutorial on the Performance Assessment of Stochastic Multiobjective Optimizers. Technical Report 214, Computer Engineering and Networks Laboratory (TIK), ETH Zurich.

[127] Kou, G., Ergu, D. and Shang, J. (2014) Enhancing data consistency in decision matrix: Adapting Hadamard model to mitigate judgment contradiction, European Journal of Operational Research 236 (2014) 261–271.

[128] Kou, G. and Lin, C. (2014) A cosine maximization method for the priority vector derivation in AHP, European Journal of Operational Research 235 (2014) 225–232.

[129] Kou, G., Lu, Y., Peng, Y. and Shi, Y. (2012) Evaluation of Classification Algorithms Using MCDM and Rank Correlation, International Journal of Information Technology & Decision Making, Vol. 11, Issue: 1, (2012)197-225.

[130] Kou, G., Peng, Y. and Wang, G. (2014) Evaluation of clustering algorithms for financial risk analysis using MCDM methods, Information Sciences 275 (2014) 1–12.

[131] Koza, J.R. Genetic Programming. MIT Press, Cambridge; 1992.

[132] Kukkonen, S. and Lampinen, J. (2005) GDE3: the third evolution step of generalized differential evolution, Evolutionary Computation, 2005. The 2005 IEEE Congress on (Volume:1) , 443 - 450 Vol.1, 2005, DOI: 10.1109/CEC.2005.1554717.

[133] Larraenaga, P. and Lozano, J. A. (2001), Estimation of Distribution Algorithms: A New Tool for Evolutionary Computation, Kluwer Academic Publishers.

[134] Laumanns, M., Zitzler, E. and Thiele, L. (2000) A unified model for multiobjective evolutionary algorithms with elitism, in Proc. Congr. Evol. Comput., vol. 1, (2000), pp. 46–53.

[135] Li, Q., Sun, L. and Bao, L. (2011) Enhanced index tracking based on multi-objective immune algorithm, Expert Systems with Applications 38 (2011) 6101–6106.

[136] Li, D., Sun, X. and Wang, J. (2006) Optimal lot solution to cardinality constrained mean-variance formulation for portfolio selection, Math. Finance 16 (1) (2006)83–101.

[137] Li, J. and Xu, J. (2013) Multi-objective portfolio selection model with fuzzy random returns and a compromise approach-based genetic algorithm, Information Sciences 220 (2013) 507–521.

[138] Li, H. & Zhang, Q. (2009). Multiobjective Optimization Problems With Complicated Pareto Sets, MOEA/D and NSGA-II, IEEE Transactions on Evolutionary Computation, 13(2):284–302.

[139] Liagkouras, K. and Metaxiotis, K. (2013) An Elitist Polynomial Mutation Operator for Improved Performance of MOEAs in Computer Networks, pp. 1-5, Computer Communications and Networks (ICCCN), 2013 22nd International Conference on, DOI: 10.1109/ICCCN.2013.6614105.

[140] Liagkouras, K. and Metaxiotis, K. (2014) A new Probe Guided Mutation operator and its application for solving the cardinality constrained portfolio optimization problem, Expert Systems with Applications 41 (2014) 6274–6290.

[141] Liagkouras, K. and Metaxiotis, K. (2015) An Experimental Analysis of a New Interval-Based Mutation Operator, International Journal of Computational Intelligence and Applications, Vol. 14, No. 3 (2015) 1550018, DOI: 10.1142/S1469026815500182.

[142] Liagkouras, K. and Metaxiotis, K. (2015) Efficient Portfolio Construction with the Use of Multiobjective Evolutionary Algorithms: Best Practices and Performance Metrics, International Journal of Information Technology & Decision Making, Vol. 14, No. 3 (2015) 535–564.

[143] Liagkouras, K. and Metaxiotis, K. (2013) The Constrained Mean-Semivariance Portfolio Optimization Problem with the Support of a Novel Multiobjective Evolutionary Algorithm, Journal of Software Engineering and Applications, (2013) 6 (07), 22.

[144] Lin, D., Wang, S. (2002) A genetic algorithm for portfolio selection problems, Advanced Modeling and Optimization 4 (1), 13–27.

[145] Lin D., Wang S., and Yan H., "A multiobjective genetic algorithm for portfolio selection," Working Paper, Institute of Systems Science, Academy of Mathematics and Systems Science Chinese Academy of Sciences, Beijing, China, 2001.

[146] Lipinski, P., Korczak, J.J. (2004) Performance Measures in an Evolutionary Stock Trading Expert System, M. Bubak et al. (Eds.): ICCS 2004, LNCS 3039, pp. 835–842, 2004. Springer-Verlag Berlin Heidelberg 2004.

[147] Lwin, K., Qu, R. and Kendall, G. (2014) A learning-guided multi-objective evolutionary algorithm for constrained portfolio optimization, Applied Soft Computing 24 (2014) 757–772.

[148] Mardle, S., Pascoe, S. and Tamiz, M. (2000) An investigation of genetic algorithms for the optimisation of multi-objective fisheries bioeconomic models, International Transactions of Operations Research, vol. 7, no. 1, pp. 33–49, 2000.

[149] Maringer, D., (2005) Portfolio Management with Heuristic Optimization. Advanced in Computational Management Science Series Vol. 8, Springer, 2005.

[150] Maringer, D. & Kellerer, H., (2003) Optimization of cardinality constrained portfolios with a hybrid local search algorithm. OR Spectrum (2003) 25: 481-495.

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

[152] Masri, H., Krichen, S. & Guitouni, A. (2012). Generating efficient faces for multiobjective linear programming problems. Int. J. of Operational Research 2012 - Vol. 15, No.1 pp. 1 – 15.

[153] Mazza, A., Chicco, G. & Russo, A. (2014) Optimal multi-objective distribution system reconfiguration with multi criteria decision making-based solution ranking and enhanced genetic operators. International Journal of Electrical Power & Energy Systems, Volume 54, January 2014, Pages 255-267.

[154] Medaglia, A.L., Graves, S.B. and Ringuest, J.L. (2007) A multiobjective evolutionary approach for linearly constrained project selection under uncertainty, European Journal of Operational Research 179 (2007) 869–894.

[155] Metaxiotis, K. and Liagkouras, K. (2012) Multiobjective evolutionary algorithms for portfolio management: a comprehensive literature review, Expert Systems with Applications, 39 (14), (2012), 11685-11698.

[156] Metaxiotis, K and Liagkouras, K., (2013) A fitness guided mutation operator for improved performance of MOEAs Electronics, Circuits, and Systems (ICECS), 2013 IEEE 20th International Conference on, pp. 751-754, DOI: 10.1109/ICECS.2013.6815523.

[157] Mira, J. and Alvarez, J.R. (2005) Artificial Intelligence and Knowledge Engineering Applications: A Bioinspired Approach: First International Work-Conference on the Interplay Between Natural and Artificial Computation, IWINAC 2005, Las Palmas, Canary Islands, Spain, June 15-18, 2005, Proceedings.

[158] Mishra, S.K., Panda, G. and Majhi, R. (2014) A comparative performance assessment of a set of multiobjective algorithms for constrained portfolio assets selection, Swarm and Evolutionary Computation 16 (2014) 38–51.

[159] Moral-Escudero,R., Ruiz-Torrubiano, R. and Suarez, A. (2006), Selection of optimal investment portfolios with cardinality constraints. In: Proceedings of the 2006 IEEE Congress on Evolutionary Computation, 2006, pp. 2382–2388.

[160] Mousavi, S., Esfahanipour, A. and Fazel Zarandi, M.H. (2014) A novel approach to dynamic portfolio trading system using multitree genetic programming, Knowledge-Based Systems 66 (2014) 68–81.

[161] Mukerjee, A., Biswas, R., Deb, K. and Mathur, A.P. (2002) Multi-objective evolutionary algorithms for the risk-return trade-off in bank loan management, International Transactions in Operational Research 9 (2002) 583–597.

[162] MULTIOBJ-1.3: Speed-constrainedMulti-objective PSO (SMPSO), DIRICOM, November 2008.

[163] Nebro, A.J. ; Dipt. Lenguajes y Cienc. de la Comput., Univ. of Malaga, Malaga, Spain ; Durillo, J.J. ; Coello, C.A.C. (2013) Analysis of Leader Selection Strategies in a Multi-Objective Particle Swarm Optimizer. Evolutionary Computation (CEC), 2013 IEEE Congress on, 20-23 June 2013, pp. 3153 - 3160, 10.1109/CEC.2013.6557955.

[164] Nebro, A. J. , Durillo, J. J., Luna, F., Dorronsoro, B. and Alba, E. (2007) MOCell: A cellular genetic algorithm for multiobjective optimization, International Journal of Intelligent Systems, 25-36, 2007.

[165] Nebro, A.J.; Durillo, J.J. ; Garcia-Nieto, J. ; Coello Coello, C.A. ; Luna, F. ; Alba, E. (2009) SMPSO: A new PSO-based metaheuristic for multi-objective optimization, Computational intelligence in multi-criteria decision-making, 2009. mcdm '09. ieee symposium on, March 30 2009-April 2 2009 , 66 - 73 ,DOI: 10.1109/MCDM.2009.4938830.

[166] Nebro, A.J.; Luna, F. ; Alba, E. ; Dorronsoro, B. ; Durillo, J.J. ; Beham, A. (2008) AbYSS: Adapting Scatter Search to Multiobjective Optimization, Evolutionary Computation, IEEE Transactions on (Volume:12, Issue:4), 439 - 457, 2008, DOI: 10.1109/TEVC.2007.913109.

[167] Niu, Q., Zhang, H., Wang, X., Li, K. and Irwin, G.W. (2014) A hybrid harmony search with arithmetic crossover operation for economic dispatch Electrical Power and Energy Systems 62 (2014) 237–257.

[168] Qi, Y., Hou, Z., Yin, M. Sun, H. and Huang, J. (2015) An immune multi-objective optimization algorithm with differential evolution inspired recombination, Applied Soft Computing 29 (2015) 395–410.

[169] Ong, C.S., Huang, J.J. and Tzeng, G.H. (2005) A novel hybrid model for portfolio selection, Applied Mathematics and Computation 169 (2005) 1195–1210.

[170] Ono, I. and Kobayashi, S. (1997). A real-coded genetic algorithm for function optimization using unimodal normal distribution crossover. In Back, T., editor, Proceedings of the Seventh International Conference on Genetic Algorithms (ICGA-7), pages 246–253, Morgan Kaufmann, San Francisco, California.

[171] Peng, Y., Wang, G., Kou, G. and Shi, Y. (2011) An empirical study of classification algorithm evaluation for financial risk prediction, Applied Soft Computing 11 (2011) 2906–2915.

[172] Phan, D.H.; Suzuki, J. ; Boonma, P. (2011) SMSP-EMOA: Augmenting SMS-EMOA with the Prospect Indicator for Multiobjective Optimization, Tools with Artificial Intelligence (ICTAI), 2011 23rd IEEE International Conference on , 7-9 Nov. 2011 , 261 - 268 , Conference Location : Boca Raton, FL , DOI: 10.1109/ICTAI.2011.47.

[173] Pindoriya, N.M., Singh, S.N. and Singh, S.K. (2010) Multi-objective mean–variance–skewness model for generation portfolio allocation in electricity markets, Electric Power Systems Research 80 (2010) 1314–1321.

[174] Piotrowski, A.P. (2013). Adaptive Memetic Differential Evolution with Global and Local neighborhood-based mutation operators. Information Sciences, Volume 241, 20 August 2013, Pages 164-194.

[175] Price, K., Storn, R.M. & Lampinen, J.A. (2005) Differential Evolution: A Practical Approach to Global Optimization, Natural Computing Series, Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2005.

[176] Purshouse, R. (2003). On the evolutionary optimization of many objectives, Ph.D. dissertation, Univ. Sheffield, Sheffield, U.K.

[177] Rebiasz, B. (2013) Selection of efficient portfolios–probabilistic and fuzzy approach, comparative study, Computers & Industrial Engineering 64 (2013) 1019–1032.

[178] Schaerf, A. (2002) Local search techniques for constrained portfolio selection problems. Computational Economics, 20(3): (2002), 177-190.

[179] Schaffer, J. D. (1984) Some Experiments in Machine Learning Using Vector Evaluated Genetic Algorithms. Doctoral Dissertation, Department of Electrical and Biomedical Engineering, Vanderbilt University, Nashville TN.

[180] Schaffer, J.D. (1985), Multiple Objective Optimization with Vector Evaluated Genetic Algorithms. In Genetic algorithms and their applications: Proceedings of the first international conference on genetic algorithms (pp. 93–100). Hillsdale, NJ: Lawrence Erlbaum.

[181] Schaffer, J.D. & Grefenstette, J.J. (1985), Multiobjective learning via genetic algorithms. In Proceedings of the 9th international joint conference on artificial intelligence (ijcai-85) (pp. 593–595). Los Angeles, CA: AAAI.

[182] Scherer, B. and Douglas, R. (2005) Introduction to Modern Portfolio Optimization with NUOPT and S-Plus, Springer, 2005.

[183] Schlottmann, F., & Seese, D. (2004) Financial Applications of Multi-Objective Evolutionary Algorithms: Recent Developments and Future Research Directions. In C. A. Coello Coello & G. B. Lamont (Eds.), Applications of multi-objective evolutionary algorithms (pp. 627–652). Singapore: World Scientific.

[184] Schlottmann, F. and Seese, D. (2004) A hybrid heuristic approach to discrete multi-objective optimization of credit portfolios, Computational Statistics & Data Analysis 47 (2004) 373 – 399.

[185] Schwefel, H. P. (1965), 'Kybernetische evolution als strategie der experimentellen forschung inder stromungstechnik', Dipl.-Ing. thesis. (in German).

[186] Schwefel, H.P. (1981), Numerical Optimization of Computer Models, Wiley, Chichester, UK.

[187] Shaw, D. X., Liu, S. and Kopman, L. (2008) Lagrangian relaxation procedure for cardinality-constrained portfolio optimization, Optimization Methods and Software, 23, 411–420.

[188] Shoaf, J. & Foster J. (1996). The Efficient Set GA for Stock Portfolios, Dept. Of Computer Science, University of Idaho, Moscow, Idaho.

[189] Shoaf, J. S., & Foster, J. A. (1996). A Genetic Algorithm Solution to the Efficient Set Problem: A Technique for Portfolio Selection Based on the Markowitz Model. In Proceedings of the decision sciences institute annual meeting (pp. 571–573). Orlando, Florida.

[190] Shukla, N., Choudhary, A.K., Prakash, P.K.S., Fernandes, K.J. and Tiwari, M.K. (2013), Algorithm portfolios for logistics optimization considering stochastic demands and mobility allowance, Int. J.Production Economics 141(2013)146–166.

[191] Silva, D. and Bastos-Filho, C. (2013) A Multi-Objective Particle Swarm Optimizer Based on Diversity, INTELLI 2013 : The Second International Conference on Intelligent Systems and Applications, IARIA, 2013.

[192] Singh H.K., Ray T, Smith W. (2010) Performance of infeasibility empowered memetic algorithm for CEC 2010 constrained optimization problems. In: Proceedings of the IEEE congress on evolutionary computation (CEC2010), Barcelona, Spain, 2010. p. 1–8.

[193] Skolpadungket, P., Dahal, K. & Harnpornchai, N., (2007), Portfolio optimization using multi-objective genetic algorithms, IEEE Congress on Evolutionary Computation (CEC 2007).

[194] Smimou, K. (2014), International portfolio choice and political instability risk: A multi-objective approach, European Journal of Operational Research 234 (2014) 546–560.

[195] Soleimani, H., Golmakani, H.R. & Salimi, M.H. (2009) Markowitz-based portfolio selection with minimum transaction lots, cardinality constraints and

regarding sector capitalization using genetic algorithm, Expert Systems with Applications, 36 (2009):5058-5063.

[196] Srinivas N., Deb K., (1994), Multi-objective optimization using non-dominated sorting in genetic algorithms, Evolutionary Computation 2 (3) 221–248.

[197] Stein, M., Branke, J. and Schmeck, H. (2007) Efficient implementation of an active set algorithm for large-scale portfolio selection, Computers & Operations Research, 35, 3945–3961.

[198] Steuer, R.E., Qi, Y. and Hirschberger, M. (2007) Suitable-portfolio investors, nondominated frontier sensitivity, and the effect of multiple objectives on standard portfolio selection, Annals of Operations Research, 152, 297–317.

[199] Streichert F, Ulmer H, Zell A. (2003). Evolutionary algorithms and the cardinality constrained portfolio optimization problem. In: Selected papers of the international conference on operations research, 2003. pp. 253–60.

[200] Streichert, F., Ulmer, H. and Zell, A. (2004) Evaluating a hybrid encoding and three crossover operators on the constrained portfolio selection problem. Proceedings of the Congress on Evolutionary Computation (CEC 2004), Portland, Oregon, (2004), pp. 932-939.

[201] Streichert F., Ulmer H., and Zell A., (2004) Comparing Discrete and Continuous Genotypes on the Constrained Portfolio Selection Problem, in Genetic and Evolutionary Computation–GECCO 2004. Proceedings of the Genetic and Evolutionary Computation Conference. Part II, K. D. et al., Ed. Seattle, Washington, USA: Springer-Verlag, Lecture Notes in Computer Science Vol. 3103, June 2004, pp. 1239–1250.

[202] Streichert, F. and Tanaka-Yamawaki, M. (2006) The Effect of Local Search on the Constrained Portfolio Selection Problem. In Proceedings of IEEE Congress on Evolutionary Computation, Vancouver, BC, Canada, pp. 2368–2374, 2006.

[203] Subbu R., Bonissone P. P., Eklund N., Bollapragada S., and Chalermkraivuth K., (2005) "Multiobjective Financial Portfolio Design: A Hybrid Evolutionary Approach," in 2005 IEEE Congress on Evolutionary Computation (CEC'2005), vol. 2. Edinburgh, Scotland: IEEE Service Center, September 2005, pp. 1722–1729.

[204] Suksonghong, K., Boonlong, K. and Goh, K.L. (2014) Multi-objective genetic algorithms for solving portfolio optimization problems in the electricity market, Electrical Power and Energy Systems 58 (2014) 150–159.

[205] Tang, P.H. & Tseng, M.H. (2013). Adaptive directed mutation for real-coded genetic algorithms. Applied Soft Computing 13 (2013) 600–614.

[206] Thomaidis, N. (2010) Active portfolio management from a fuzzy multi-objective programming perspective, C Di Chio et al (Eds): EvoApplications 2010, Part II, LNCS 6025, pp. 222-231, Springer-Verlag Berlin Heidelberg 2010.

[207] Trail, C.D., Schmiedekamp, M.D., Phoha, S., Sustersic, J.P. (2013). Multiobjective planning for naval mine counter measures missions. Int. J. of Operational Research 2013 - Vol. 16, No.1 pp. 113 – 135.

[208] Tsutsui, S. and Yamamura, M. T. Higuchi, (1999) Multi-parent recombination with simplex crossover in real coded genetic algorithms, in: Proceedings of the GECCO-99, 1999, pp. 644–657.

[209] Vazhayil, J.P. and Balasubramanian, R. (2014) Optimization of India's electricity generation portfolio using intelligent Pareto-search genetic algorithm, Electrical Power and Energy Systems 55 (2014) 13–20.

[210] Vedarajan, G., Chan, L. C. and Goldberg, D. E. (1997). Investment Portfolio Optimization using Genetic Algorithms. In Koza, J. R., editor, Late Breaking Papers at the Genetic Programming 1997 Conference, pages 255–263, Stanford Bookstore, Stanford University, California.

[211] Veldhuizen, D.A.V. (1999) Multiobjective Evolutionary Algorithms: Classifications, Analyses, and New Innovations. PhD thesis, Graduate School of Engineering, Air Force Institute of Technology, Air University, June 1999.

[212] Veldhuizen, D.A.V. and Lamont, G.B. (1998) Multiobjective evolutionary algorithm research: A history and analysis. Technical Report TR-98-03, Department of Electrical and Computer Engineering, Graduate School of Engineering, Air Force Institute of Technology, Wright-Patterson AFB, OH, 1998.

[213] Vijayalakshmi Pai, G.A. and Thierry, M. (2014) Metaheuristic multi-objective optimization of constrained futures portfolios for effective risk management, Swarm and Evolutionary Computation, Article in press, http://dx.doi.org/10.1016/j.swevo.2014.08.002.

[214] Wang,F., Yu, P.L.H. and Cheung, D.W. (2014), Combining technical trading rules using particle swarm optimization, Expert Systems with Applications 41 (2014) 3016–3026.

[215] Winker, P., Gilli, M. (2004) Applications of optimization heuristics to estimation and modelling problems, Computational Statistics & Data Analysis 47 (2004) 211 – 223.

[216] Woodside-Oriakhi, M., Lucas, C. & Beasley, J.E. (2011). Heuristic algorithms for the cardinality constrained efficient frontier. European Journal of Operational Research 213, pp. 538-550.

[217] Woodside-Oriakhi, M., Lucas, C. and Beasley, J.E. (2013) Portfolio rebalancing with an investment horizon and transaction costs, Omega 41(2013)406–420.

[218] Xidonas, P., Mavrotas, G. & Psarras J. (2010). Portfolio management within the frame of multiobjective mathematical programming: a categorised bibliographic study. Int. J. of Operational Research 2010 - Vol. 8, No.1 pp. 21 – 41.

[219] Yoo, S. Harman, M. & Shmuel Ur. (2011). Highly Scalable Multi Objective Test Suite Minimisation Using Graphics Cards. Search Based Software Engineering Lecture Notes in Computer Science, Volume 6956.

[220] Yoon, Y., Kim, Y.H., Moraglio, A., Moon, B.R. (2012) A theoretical and empirical study on unbiased boundary-extended crossover for real-valued representation, Information Sciences 183 (2012) 48–65.

[221] (*38) Zeng, F., Decraene, J., Hean Low, M.Y., Hingston, P., Wentong, C., Suiping, Z. & Chandramohan, M. (2010). Autonomous Bee Colony Optimization for Multi-objective Function. Proceedings of the IEEE Congress on Evolutionary Computation.

[222] Zhang, Q. and Li, H. (2007): MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition. IEEE Trans. on Evolutionary Computation 11 (2007) 712-731.

[223] Zhang, Q., Liu, W. and Li, H. (2009): The Performance of a New Version of MOEA/D on CEC09 Unconstrained MOP Test Instances." IEEE Congress on Evolutionary Computation,Trondheim, Norway, (2009).

[224] Zhang, P. and Zhang, W.G. (2014) Multiperiod mean absolute deviation fuzzy portfolio selection model with risk control and cardinality constraints, Fuzzy Sets and Systems 255(2014)74–91.

[225] Zhou, Y., Li, X. & Gao, L. (2013). A differential evolution algorithm with intersect mutation operator. Applied Soft Computing, Volume 13, Issue 1, January 2013, Pages 390-401.

[226] Zhu, H., Wang Y., Wang K. & Chen Y. (2011) Particle swarm optimization (PSO) for the constrained portfolio optimization problem, Expert Systems with Applications 38 (2011): 10161-10169.

[227] Zitzler, E. (1999) Evolutionary algorithms for multiobjective optimization: Methods and applications, Ph.D. dissertation, Swiss Federal Inst. Technology (ETH) (1999), Zurich, Switzerland.

[228] Zitzler, E., Brockhoff, D. & Thiele. L.(2007). The Hypervolume Indicator Revisited: On the Design of Pareto-compliant Indicators Via Weighted Integration. In Conference on Evolutionary Multi-Criterion Optimization (EMO 2007), pages 862–876. Springer.

[229] Zitzler, E., Deb, K., and Thiele, L. (2000) Comparison of multiobjective evolutionary algorithms: Empirical results. Evolutionary Computation, 8(2). 173-195.

[230] Zitzler, E. and Kunzli, S. (2004) Indicator-Based Selection in Multiobjective Search. In Conference on Parallel Problem Solving from Nature (PPSN VIII), volume 3242 of LNCS, pages 832–842. Springer, 2004.

[231] Zitzler E., Laumanns M., and Thiele L., (2001) SPEA2: Improving the Strength Pareto Evolutionary Algorithm, Computer Engineering and Networks Laboratory (TIK) Department of Electrical Engineering, Swiss Federal Institute of Technology (ETH) Zurich, 2001.

[232] Zitzler, E., Laumanns, M., Thiele, L., Fonseca, C. M., & Grunert da Fonseca, V.(2002). Why Quality Assessment of Multiobjective Optimizers Is Difficult. In W. Langdon et al. (Eds.), Proceedings of the genetic and evolutionary computation conference (gecco'2002) (pp. 666–673). San Francisco, California: Morgan Kaufmann Publishers.

[233] Zitzler, E. and Thiele, L. (1999) Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach. IEEE Transactions on Evolutionary Computation, 3(4):257–271, 1999.

[234] Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C.M. and Da Fonseca, V.G. (2003) Performance assessment of multiobjective optimizers: an analysis and review. IEEE Transactions on Evolutionary Computation, (2003), 7(2):117–132.