

Pareto-Optimal Search-Based Software Engineering (POSBSE): A Literature Survey

Abdel Salam Sayyad

Hany Ammar

*Lane Department of Computer Science and Electrical Engineering
West Virginia University
Morgantown, WV, USA*

asayyad@mix.wvu.edu

hany.ammar@mail.wvu.edu

Abstract—The Search-Based Software Engineering (SBSE) community is increasingly recognizing the inherit “multiobjectiveness” in Software Engineering problems. The old ways of aggregating all objectives into one may very well be behind us. We perform a well-deserved literature survey of SBSE papers that used multiobjective search to find Pareto-optimal solutions, and we pay special attention to the chosen algorithms, tools, and quality indicators, if any. We conclude that the SBSE field has seen a trend of adopting the Multiobjective Evolutionary Optimization Algorithms (MEOAs) that are widely used in other fields (such as NSGA-II and SPEA2) without much scrutiny into the reason why one algorithm should be preferred over the others. We also find that the majority of published work only tackled two-objective problems (or formulations of problems), leaving much to be desired in terms of exploiting the power of MEOAs to discover solutions to intractable problems characterized by many trade-offs and complex constraints.

Index Terms—Multiobjective Optimization, Pareto-Optimal Solutions, Search-Based Software Engineering.

I. INTRODUCTION

Many real-life challenges involve trading off multiple evaluation criteria. Rather than aggregating all the criteria in one equation that leads to a unique optimal solution, it is more pragmatic to present the end-user with a bouquet of candidate solutions that are reasonably distributed along the possible objective values. These are formally known as Pareto-optimal solutions or the Pareto front.

Historically, the field of Search-Based Software Engineering (SBSE) has seen a slow adoption of Pareto optimization techniques, generally known as multiobjective optimization techniques¹. Back in 2001, when Harman and Jones coined the term SBSE [24], all surveyed and suggested techniques were based on single-valued fitness functions. In 2007, Harman commented on the current state and future of SBSE [23], and in the “Road-map for Future Work” section he suggested using multiobjective optimization. Then in 2009, Harman et al. [26] were able to cite several works in which multiobjective optimization techniques were deployed. Still, most of the work reviewed therein, as well as work done thereafter, optimized two objectives only, while higher numbers appeared only occasionally. Also, the typical tendency

was to use algorithms that are popular in other domains, such as NSGA-II and SPEA2. In this survey, we attempt to systematically identify these trends and provide some recommendations to the growing SBSE community.

We were motivated to undertake this survey by our recent finding [44] that IBEA, an algorithm never used before in software engineering, outperformed several others (including NSGA-II and SPEA2) when applied to complex models and many objectives, hence our questions regarding suitability of optimizers to software engineering problems.

Several surveys and review papers were published in the field of SBSE [23], [26], [27], [11]. Some surveys focused on subfields, such as search-based software design [40], and search-based test data generation [37]. This paper is the first of its kind, in which we survey SBSE research work that employed Pareto optimization techniques. Such work spans the application areas of software requirements, design, testing, and management.

The rest of the paper is organized as follows: section II provides brief background material on Pareto optimality. Section III describes the survey’s inclusion/exclusion criteria. Section IV presents the results of the survey. In sections V, we analyze the survey data, and in section VI we present our conclusions and recommendations.

II. BACKGROUND

A. Pareto Optimization

Pareto optimization is the process of finding a set of non-dominated solutions from a pool of candidate solutions. Each non-dominated solution can be viewed as an optimal trade-off in all the objectives together.

Formally, a vector $u = \{u_1, \dots, u_k\}$ is said to dominate a vector $v = \{v_1, \dots, v_k\}$ if and only if u is partially less than v , i.e.

$$\forall i \in \{1, \dots, k\}, u_i \leq v_i \text{ and } \exists i \in \{1, \dots, k\} : u_i < v_i \quad (1)$$

The set of all points in the objective space that are not dominated by any other points is called the *Pareto Front*.

B. Multiobjective Evolutionary Optimization Algorithms (MEOAs)

MEOAs are the most popular class of Pareto optimizers. Among them, the most popular in SBSE are NSGA-II [13],

¹ Some techniques (e.g. awGA, HATS, Max-Min) are labeled as multiobjective but they actually aggregate the objectives and provide a single optimal solution. Such techniques are outside the scope of this survey.

and SPEA2 [62]. Many other MEOAs are used in SBSE, as can be seen in sections IV and V.

C. Tools/Frameworks

Tools are available in which MEOAs are coded and ready to use. The most comprehensive tool we know is jMetal [14], which implements 17 different MEOAs in addition to Random Search, and provides a framework for experimentation with standard and user-defined problems.

D. Quality Indicators

Quality indicators can be calculated to assess the performance of MEOAs by computing aggregate values from the objective values in the Pareto front. For example, jMetal [14] allows the computation of six quality indicators, namely: Hypervolume (HV), Spread, Generational Distance (GD), Inverted Generational Distance (IGD), Epsilon, and Generalized Speed.

Other quality indicators were found in the SBSE literature that we surveyed. They were: coverage, convergence, error ratio, and attainment surfaces.

III. INCLUSION/EXCLUSION CRITERIA

We seek to include all published research works, up to date, in the field of search-based software engineering (SBSE) that provide Pareto-optimal solutions using multiobjective optimization algorithms. In each paper, we look for the number of objectives, the algorithms and how they were chosen, whether a framework for metaheuristic algorithms was used, and whether quality indicators were used to judge the quality of the Pareto front.

When more than one paper is published by the same author(s) that apply the same technique(s) to the same class of problems, we list only one of those papers, unless the extra paper adds algorithms for comparison, or changes the number of objectives, or adds quality indicators.

We utilized the CREST center SBSE repository², which is the primary listing of SBSE papers. The “area of application” classification that we follow here is, generally, the same one used in that repository.

IV. RESULTS

The total number of surveyed papers was 51. Figure 1 breaks them down by area of application, while Figure 2 classifies them by year of publication.

Table 1 is a list of all the surveyed papers, along with the multiobjective algorithms, the number of objectives, tools and quality indicators.

V. ANALYSIS

In this section, we analyze and provide commentary about the data collected from the surveyed papers with regard to algorithms, number of objectives, tools, and quality indicators.

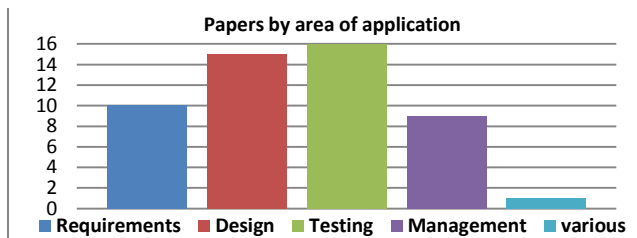


Figure 1: Papers by area of application

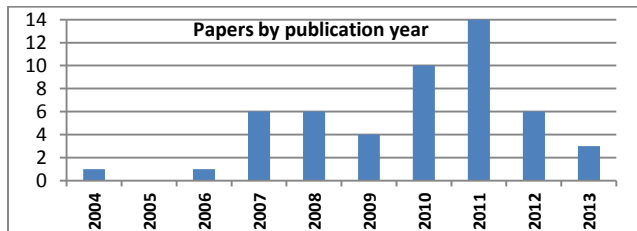


Figure 2: Papers by publication year

A. Algorithms

The total number of different algorithms used in all 51 papers was 25 algorithms. Figure 3 shows the frequency of use for each of the most used algorithms. The “other” category includes 20 algorithms that were used only once or twice in all the surveyed papers.

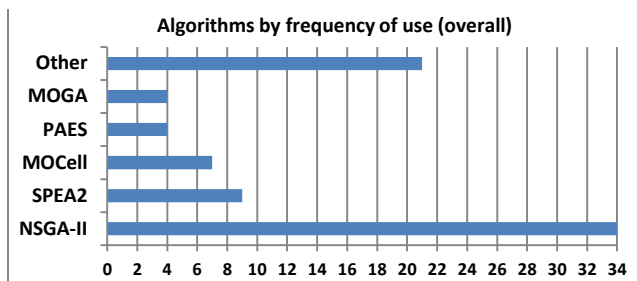


Figure 3: Algorithms by frequency of use

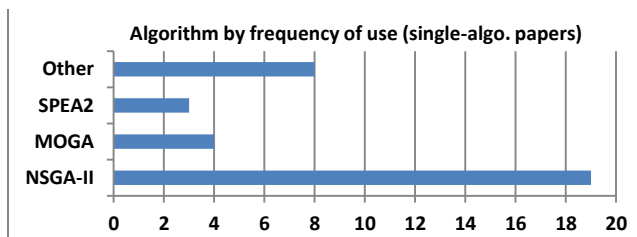


Figure 4: Algorithms by frequency of use (single-algorithm papers)

We observe that 36 papers (70%) used a single algorithm, whereas 15 papers (30%) used multiple algorithms for comparison purposes.

Figure 4 shows the frequency of use in the papers that only used a single Pareto optimization algorithm. NSGA-II was the algorithm of choice in 53% of those papers. The “other” category includes 9 algorithms that were used only once or twice in this category of papers.

We notice that MOCeII and PAES appear nowhere in this category. They were only used in comparison to other algorithms, but never alone as the algorithm of choice.

For those 35 papers that only used one algorithm, we pose an important question: why choose that particular algorithm? The answers are charted in Figure 5.

² http://crestweb.cs.ucl.ac.uk/resources/sbse_repository/

TABLE 1: LIST OF ALL SURVEYED WORKS

Ref	Author(s)	Year	Title	Number of Objectives	Algorithm(s)	Tool	Quality Indicators
<i>Application Area: Requirements/Specifications</i>							
[61]	Y. Zhang, M. Harman, S. A. Mansouri	2007	The Multi-Objective Next Release Problem	2	NSGA-II, Pareto GA	--	--
[19]	A. Finkelstein, M. Harman, S. Mansouri, J. Ren, Y. Zhang	2008	"Fairness analysis" in requirements assignments	2	NSGA-II	--	--
[16]	J. J. Durillo, Y. Zhang, E. Alba, A. J. Nebro	2009	A Study of the Multi-Objective Next Release Problem	2	NSGA-II, MOCeII	jMetal	HV, Spread
[60]	Y. Zhang, E. Alba, J. J. Durillo, S. Eldh, M. Harman	2010	Today/Future Importance Analysis	3	NSGA-II	--	--
[15]	J. J. Durillo, Y. Zhang, E. Alba, M. Harman, A. J. Nebro	2011	A Study of the Bi-Objective Next Release Problem	2	NSGA-II, MOCeII, PAES	jMetal	HV, Spread
[5]	M. M. A. Brasil, T. G. N. da Silva, F. G. de Freitas, J. T. de Souza, M. I. Cortés	2011	A Multiobjective Optimization Approach to the Software Release Planning with Undefined Number of Releases and Interdependent Requirements	3	NSGA-II, MOCeII	jMetal	HV, Spread
[29]	W. Heaven, E. Letier	2011	Simulating and Optimising Design Decisions in Quantitative Goal Models	2	NSGA-II	Matlab	--
[47]	V. Veerappa, E. Letier	2011	Understanding Clusters of Optimal Solutions in Multi-Objective Decision Problems	2	GA with Clustering	Matlab	--
[33]	A.C. Kumari, K. Srinivas, M.P. Gupta	2012	Software Requirements Selection using Quantum-inspired Elitist Multi-objective Evolutionary Algorithm	2	QEMEA	--	HV, Spread, Convergence
[44]	A. S. Sayyad, T. Menzies, H. Ammar	2013	On the Value of User Preferences in Search-Based Software Engineering: A Case Study in Software Product Lines	2, 3, 4, 5	IBEA, NSGA-II, ssNSGA-II, SPEA2, FastPGA, MOCeII, MOCHC	jMetal	HV, Spread
<i>Application Area: Design Tools and Techniques</i>							
[31]	T. M. Khoshgoftaar, Y. Liu, N. Seliya	2004	A Multiobjective Module-Order Model for Software Quality Enhancement	4	NSGA-II	--	--
[21]	L. Grunske	2006	Identifying "Good" Architectural Design Alternatives with Multi-Objective Optimization Strategies	2	MOGA	--	--
[35]	Z. Liu, H. Guo, D. Li, T. Han, J. Zhang	2007	Solving Multi-objective and Fuzzy Multi-attributive Integrated Technique for QoS-Aware Web Service Selection	5	MOGA	Frontier	--
[28]	M. Harman, L. Tratt	2007	Pareto Optimal Search Based Refactoring at the Design Level	2	Pareto Hill Climbing	--	--
[1]	A. Arcuri, D. R. White, J. Clark, X. Yao	2008	Multi-Objective Improvement of Software using Co-Evolution and Smart Seeding	2	SPEA2	--	--
[50]	J. Wang, Y. Hou	2008	Optimal Web Service Selection based on Multi-Objective Genetic Algorithm	2	MOGA	--	--
[48]	H. Wada, P. Champrasert, J. Suzuki, K. Oba	2008	Multiobjective Optimization of SLA-Aware Service Composition	10	E ³ -MOGA	--	--
[45]	C. L. Simons, I. C. Parmee	2008	User-centered, Evolutionary Search in Conceptual Software Design	2	NSGA-II	--	--
[46]	C. L. Simons, I. C. Parmee, R. Gwynllyw	2010	Interactive, evolutionary search in upstream object-oriented class design	3	NSGA-II	--	--
[4]	M. Bowman, L. C. Briand, Y. Labiche	2010	Solving the Class Responsibility Assignment Problem in Object-Oriented Analysis with Multi-Objective Genetic Algorithms	5	SPEA2	--	--
[41]	O. Räihä, K. Koskimies, E. Mäkinen	2011	Generating Software Architecture Spectrum with Multi-objective Genetic Algorithms	2	MOGA	--	--
[39]	K. Praditwong, M. Harman, X. Yao	2011	Software Module Clustering as a Multi-Objective Search Problem	5	Two-Archive MOEA	--	--
[3]	M. O. Barros	2012	An Analysis of the Effects of Composite Objectives in Multiobjective Software Module Clustering	4, 5	NSGA-II	jMetal	GD, Error Ratio
[10]	T. E. Colanzi, S.R. Vergilio	2012	Applying Search Based Optimization to SPL Architectures: Lessons Learned	5	NSGA-II	--	--
[53]	J. L. Wilkerson, D. R. Tauritz, J. M. Bridges	2102	Multi-objective Coevolutionary Automated Software Correction	4	NSGA-II	--	--
[20]	S. Frey, F. Fittkau, W. Hasselbring	2013	Search-based Genetic Optimization for Deployment and Reconfiguration of Software in the Cloud	3	NSGA-II	Opt4J	HV, IGD

Ref	Author(s)	Year	Title	Number of Objectives	Algorithm(s)	Tool	Quality Indicators
Application Area: Testing/Debugging							
[56]	S. Yoo, M. Harman	2007	Pareto Efficient Multi-Objective Test Case Selection	2, 3	NSGA-II, vNSGA-II	--	--
[25]	M. Harman, K. Lakhotia, P. McMinn	2007	A Multi-Objective Approach to Search-based Test Data Generation	2	NSGA-II	--	--
[51]	Z. Wang, K. Tang, X. Yao	2008	Multi-objective Approach to Testing Resource Allocation in Modular Software	2, 3	NSGA-II, MODE	--	--
[36]	C. L. B. Maia, R. A. Carmo, F. G. de Freitas, G. A. L. de Campos, J. T. de Souza	2009	A Multi-Objective Approach for the Regression Test Case Selection Problems	3	NSGA-II	jMetal	--
[55]	T. Yano, E. Martins, F. L. de Sousa	2010	Generating Feasible Test Paths from an Executable Model Using a Multi-Objective Approach	2	M-GEO _{vsf}	--	--
[38]	G. H. L. Pinto, S. R. Vergilio	2010	A Multi-Objective Genetic Algorithm to Test Data Generation	3	NSGA-II	--	--
[34]	W. B. Langdon, M. Harman, Y. Jia	2010	Efficient Multi Objective Higher Order Mutation Testing with Genetic Programming	2	NSGA-II	--	--
[57]	S. Yoo, M. Harman	2010	Using Hybrid Algorithm For Pareto Efficient Multi-Objective Test Suite Minimisation	2, 3	HNSGA-II	--	--
[52]	Z. Wang, K. Tang, X. Yao	2010	Multi-Objective Approaches to Optimal Testing Resource Allocation in Modular Software Systems	2, 3	NSGA-II, HaD-MOEA	--	HV
[7]	R. da Veiga Cabral, A. T. R. Pozo, S. R. Vergilio	2010	A Pareto Ant Colony Algorithm applied to Class Integration and Test Order Problem	2	Pareto Ant Colony	--	--
[9]	T. E. Colanzi, W. Assuncao, S. R. Vergilio, A. Pozo	2011	Generating integration test orders for aspect-oriented software	2	NSGA-II, SPEA2	jMetal	GD, ED
[2]	W. K. G. Assunção, T. E. Colanzi, A. T. R. Pozo, S. R. Vergilio	2011	Establishing Integration Test Orders of Classes with Several Coupling Measures	4	NSGA-II, SPEA2	jMetal	GD, IGD, Coverage, ED
[59]	S. Yoo, R. Nilsson, M. Harman	2011	Faster Fault Finding at Google using Multi-Objective Regression Test Optimisation	3	Two-Archive MOEA	--	--
[58]	S. Yoo, M. Harman, S. Ur	2011	Highly Scalable Multi-Objective Test Suite Minimisation Using Graphics Cards	2	NSGA-II	jMetal	--
[17]	J. Ferrer, F. Chicano, E. Alba	2011	Evolutionary Algorithms for the Multi-objective Test Data Generation Problem	2	NSGA-II, SPEA2, PAES, MOCeII, Random	jMetal	HV, attainment surfaces
Application Area: Software Project Management							
[30]	T. M. Khoshgoftaar, Y. Liu	2007	A Multi-Objective Software Quality Classification Model Using Gen. Prog.	3	NSGA-II	--	--
[22]	S. Gueorguiev, M. Harman, G. Antoniol	2009	Software project planning for robustness and completion time	2	SPEA2	--	--
[49]	Z. Wang, T. Chen, K. Tang, X. Yao	2009	A Multi-objective Approach to Redundancy Allocation Problem in Parallel-series Systems	2	NSGA-II	--	--
[32]	T. Kremmel, J. Kubalik, S. Biffl	2011	Software Project Portfolio Optimization with Advanced Multiobjective Evolutionary Algorithms	5	mPOEMS, NSGA-II, SPEA2	--	HV, Coverage
[8]	F. Chicano, F. Luna, A. J. Nebro, E. Alba	2011	Using Multi-objective Metaheuristics to Solve the Software Project Scheduling Problem	2	NSGA-II, SPEA2, PAES, MOCeII, GDE3	jMetal	HV, attainment surfaces
[42]	D. Rodríguez, M. Ruiz, J. C. Riquelme, R. Harrison	2011	Multiobjective Simulation Optimisation in Software Project Management	2, 3, 5	NSGA-II	jMetal	--
[6]	R. Britto, P. S. Neto, R. Rabelo, W. Ayala, T. Soares	2012	A Hybrid Approach to Solve the Agile Team Allocation Problem	2	NSGA-II	--	--
[43]	F. Sarro, F. Ferrucci, C. Gravino	2012	Single and Multi Objective GP for Software Development Effort Estimation	5	MOGP	--	--
[18]	F. Ferrucci, M. Harman, J. Ren, F. Sarro	2013	Not Going to Take this Anymore: Multi-Objective Overtime Planning for Software Engineering Projects	3	NSGA-II, NSGA-II _v	--	HV, GD, Coverage
Application Area: Various Applications (Next Release Problem & Test Case Selection)							
[12]	J. T. de Souza, C. L. Maia, F. G. de Freitas, D. P. Coutinho	2010	The human competitiveness of search based software engineering	2	NSGA-II, MOCeII, Random	jMetal	HV, Spread, Coverage

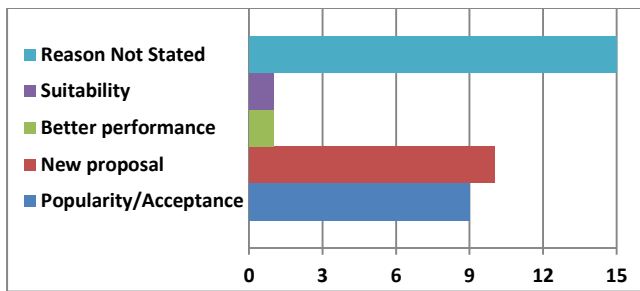


Figure 5: Reasons for adopting an algorithm

Most researchers didn't state any reason for adopting a certain MEOA (42%) or stated popularity of the algorithm (particularly NSGA-II) as the sole reason (25%). This is usually justified by the fact that many of those papers introduced Pareto-optimal solutions to their problems for the first time, which in itself is considered a contribution. A sizeable portion of these papers (10, 28%) introduced new MEOAs to solve a problem, but only compared their outcome with that of single-objective algorithms. Looking at the 15 papers that actually compared MEOAs to one another, 5 of them introduced new MEOAs and compared their outcomes with popular MEOAs.

In the single-MEOA papers, only one stated that the MEOA was chosen because it was more suitable for the particular problem, and one paper stated that the chosen algorithm had been reported to have better performance in higher-dimension objective spaces.

B. Number of Objectives

The first step in Pareto optimization is to re-formulate the problem to bring out the competing objectives. Although the objectives were known all along, legacy research combined them into one fitness function. Pareto optimization researchers then had to decide which objectives to represent as separate, which to combine into one, and which to ignore; thus a different number of objectives may have been defined for the same problem. Figure 6 shows the variety of "number of objectives in the surveyed papers. There were 7 papers that presented different formulations for each problem, with a different number of objectives in each formulation.

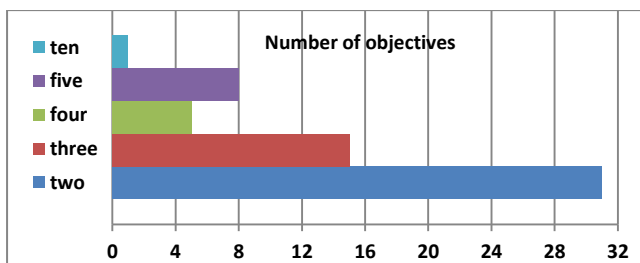


Figure 6: Number of Objectives by frequency of use

C. Tools/Frameworks

17 papers (33%) reported using implementations of the algorithms that are available in tools such as jMetal (13 papers) and Matlab (2 papers). Two thirds of the time, the researchers had to code their own implementations, which is the assumption we made when tools weren't mentioned.

D. Quality Indicators

15 papers (30%) used quality indicators to assess the quality of the Pareto fronts and, most of the time (12 papers), to compare the performance of various MEOAs against one another. Hypervolume (HV) was the most widely used indicator (12 papers), while there was lesser agreement on the use of other indicators.

Quality indicators are useful as aggregate measures for large sets of solutions, especially with higher dimensions in the objective space. Skipping the computation of quality indicators is understandable when it's possible to directly assess the objective values, which is the case when the Pareto front can be charted in 2-D or 3-D.

VI. CONCLUSIONS AND RECOMMENDATIONS

In this paper, we surveyed 51 research papers that applied multiobjective search-based optimization methods to software engineering problems. This being a relatively young trend in SBSE, we have observed certain shortcomings in many papers:

1. Lack of clarity regarding the reasons why an algorithm is chosen for a problem (Figure 5).
2. Tendency to simplify problems by specifying fewer objectives to evaluate (Figure 6),
3. Heavy reliance on personal implementations of widely-used algorithms (Table 1, the "Tool" column).
4. Lack of agreement on whether to utilize quality indicators, and which indicators to use (Table 1, the "Quality Indicators" column).

But we also noticed some promising directions:

1. Researchers are comparing algorithms against one another to discover better performance, and to reason about suitability of the algorithms to the problems at hand (15 papers out of 51).
2. Some papers are exploring different formulations of their problems wherein the complexity is increased and more objectives are evaluated (7 papers out of 51).
3. Increasing use of the open-source jMetal framework with its rich set of MEOAs and quality indicators (13 papers out of 51).

Finally, we offer some recommendations to the SBSE research community:

- 1- Single-valued fitness functions are a thing of the past. Software engineering problems are multiobjective by nature, and Pareto optimization is the best way to find all the possible trade-offs among the objectives such that the stakeholders can make enlightened decisions.
- 2- More attention needs to be paid to the suitability of an algorithm to the type of problem at hand. It is true that the right optimizer for a specific problem remains an open question [54], but there has to be a thought process about the structure of the problem and the suitability of the metaheuristic.
- 3- More comparisons regarding the performance of various algorithms when applied to specific problems.

4- Reformulating two- and three-objective problems to bring out objectives that might have been aggregated or ignored. In addition to being closer to the business reality of many competing objectives, this should put MEOAs under increased stress, and enable testing out reported results about certain MEOAs performing better than others at higher dimensions (e.g. [44], [4]).

5- Utilizing and contributing to the MEOAs available in jMetal, as well as its quality indicator offerings.

ACKNOWLEDGMENT

This research work was funded by the Qatar National Research Fund (QNRF) under the National Priorities Research Program (NPRP) Grant No.: 09-1205-2-470.

REFERENCES

- [1] A. Arcuri, D. R. White, J. Clark, and X. Yao, "Multi-Objective Improvement of Software using Co-Evolution and Smart Seeding," in *Proc. SEAL*, Melbourne, Australia, 2008, pp. 61-70.
- [2] W. K. G. Assunção, T. E. Colanzi, A. T. R. Pozo, and S. R. Vergilio, "Establishing Integration Test Orders of Classes with Several Coupling Measures," in *Proc. GECCO*, Dublin, Ireland, 2011, pp. 1867-1874.
- [3] M. O. Barros, "An Analysis of the Effects of Composite Objectives in Multiobjective Software Module Clustering," in *Proc. GECCO*, Philadelphia, USA, 2012, pp. 1205-1212.
- [4] M. Bowman, L.C. Briand, and Y. Labiche, "Solving the class responsibility assignment problem in object-oriented analysis with multi-objective genetic algorithms," *IEEE Transactions on Software Engineering*, vol. 36, no. 6, pp. 817-837, 2010.
- [5] M. M. A. Brasil, da Silva T. G. N., F. G. de Freitas, J. T. de Souza, and M. I. Cortés, "A Multiobjective Optimization Approach to the Software Release Planning with Undefined Number of Releases and Interdependent Requirements," in *Proc. ICEIS*, Beijing, China, 2011.
- [6] R. Britto, P. S. Neto, R. Rabelo, W. Ayala, and T. Soares, "A Hybrid Approach to Solve the Agile Team Allocation Problem," in *Proc. CEC*, Brisbane, Australia, 2012, pp. 1-8.
- [7] R. da Veiga Cabral, A. T. R. Pozo, and S. R. Vergilio, "A Pareto Ant Colony Algorithm applied to the Class Integration and Test Order Problem," in *Proc. ICTSS*, Natal, Brazil, 2010, pp. 16-29.
- [8] F. Chicano, F. Luna, A. J. Nebro, and E. Alba, "Using Multi-objective Metaheuristics to Solve the Software Project Scheduling Problem," in *Proc. GECCO*, Dublin, Ireland, 2011, pp. 1915-1922.
- [9] T. E. Colanzi, W. Assuncao, S. R. Vergilio, and A. Pozo, "Generating Integration Test Orders for Aspect-Oriented Software with Multi-objective Algorithms," in *Latin American Workshop on Aspect-Oriented Software Development*, 2011.
- [10] T. E. Colanzi and S. R. Vergilio, "Applying Search Based Optimization to Software Product Line Architectures: Lessons Learned," in *Proc. SSBSE*, Riva del Garda, Italy, 2012, pp. 259-266.
- [11] F. G. de Freitas and J. T. de Souza, "Ten Years of Search Based Software Engineering: A Bibliometric Analysis," in *Proc. SSBSE*, Szeged, Hungary, 2011, pp. 18-32.
- [12] J. T. de Souza, C. L. Maia, F. G. de Freitas, and D. P. Coutinho, "The human competitiveness of search based software engineering," in *Proc. SSBSE*, Benevento, Italy, 2010, pp. 143-152.
- [13] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182-197, 2002.
- [14] J. J. Durillo and A. J. Nebro, "jMetal: A Java framework for multi-objective optimization," *Advances in Engineering Software*, vol. 42, pp. 760-771, 2011.
- [15] J. J. Durillo, Y. Zhang, E. Alba, M. Harman, and A. J. Nebro, "A Study of the Bi-Objective Next Release Problem," *Empirical Software Engineering*, vol. 16, no. 1, pp. 29-60, February 2011.
- [16] J. J. Durillo, Y. Zhang, E. Alba, and A. J. Nebro, "A Study of the Multi-Objective Next Release Problem," in *Proc. SSBSE*, Windsor, UK, 2009, pp. 49-58.
- [17] J. Ferrer, F. Chicano, and E. Alba, "Evolutionary Algorithms for the Multi-objective Test Data Generation Problem," *Software: Practice and Experience*, vol. 42, pp. 1331-1362, 2012.
- [18] F. Ferrucci, M. Harman, J. Ren, and F. Sarro, "Not Going to Take this Anymore: Multi-Objective Overtime Planning for Software Engineering Projects," in *Proc. ICSE*, San Francisco, USA, 2013.
- [19] A. Finkelstein, M. Harman, S. A. Mansouri, J. Ren, and Y. Zhang, "'Fairness Analysis' in Requirements Assignments," in *Proc. RE*, 2008, pp. 115-124.
- [20] S. Frey, F. Fittkau, and W. Hasselbring, "Search-based Genetic Optimization for Deployment and Reconfiguration of Software in the Cloud," in *Proc. ICSE*, San Francisco, USA, 2013.
- [21] L. Grunske, "Identifying "Good" Architectural Design Alternatives with Multi-Objective Optimization Strategies," in *Proc. ICSE*, Shanghai, China, 2006, pp. 849-852.
- [22] S. Gueorguiev, M. Harman, and G. Antoniol, "Software Project Planning for Robustness and Completion Time in the Presence of Uncertainty using Multi Objective Search Based Software Engineering," in *Proc. GECCO*, 2009, pp. 1673-1680.
- [23] M. Harman, "The Current State and Future of Search Based Software Engineering," in *Proc. FOSE*, 2007, pp. 342-357.
- [24] M. Harman and B. F. Jones, "Search based software engineering," *Information and Software Technology*, vol. 43, no. 14, pp. 833-839, December 2001.
- [25] M. Harman, K. Lakhoria, and P. McMinn, "A Multi-Objective Approach to Search-based Test Data Generation," in *Proc. of GECCO*, London, UK, 2007, pp. 1098-1105.
- [26] M. Harman, S. A. Mansouri, and Y. Zhang, "Search Based Software Engineering: A Comprehensive Analysis and Review of Trends Techniques and Applications," King's College, London, UK, Technical Report TR-09-03, 2009.
- [27] M. Harman, P. McMinn, J. T. de Souza, and S. Yoo, "Search Based Software Engineering: Techniques, Taxonomy, Tutorial," *Empirical Software Engineering and Verification*, vol. 7007, pp. 1-59, 2012.
- [28] M. Harman and L. Tratt, "Pareto Optimal Search Based Refactoring at the Design Level," in *Proc. GECCO*, London, England, 2007, pp. 1106-1113.
- [29] W. Heaven and E. Letier, "Simulating and Optimising Design Decisions in Quantitative Goal Models," in *Proc. RE*, 2011, pp. 79-88.
- [30] T. M. Khoshgoftaar and Y. Liu, "A Multi-Objective Software Quality Classification Model Using Genetic Programming,"

- IEEE Transactions on Reliability*, vol. 56, no. 2, pp. 237-245, June 2007.
- [31] T. M. Khoshgoftaar, Y. Liu, and N. Seliya, "A Multiobjective Module-Order Model for Software Quality Enhancement," *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 6, pp. 593-608, December 2004.
- [32] T. Kremmel, J. Kubalik, and S. Biffl, "Software Project Portfolio Optimization with Advanced Multiobjective Evolutionary Algorithms," *Applied Soft Computing*, vol. 11, no. 1, pp. 1416-1426, January 2011.
- [33] A.C. Kumari, K. Srinivas, and M.P. Gupta, "Software Requirements Selection using Quantum-inspired Elitist Multi-objective Evolutionary Algorithm," in *Proc. ICAESM*, India, 2012, pp. 782-787.
- [34] W. B. Langdon, Harman M., and Jia Y., "Efficient Multi Objective Higher Order Mutation Testing with Genetic Programming," *Journal of Systems and Software*, vol. 83, no. 12, pp. 2416-2430, December 2010.
- [35] Z. Liu, H. Guo, D. Li, T. Han, and J. Zhang, "Solving Multi-objective and Fuzzy Multi-attributive Integrated Technique for QoS-Aware Web Service Selection," in *Proc. WiCom*, Shanghai, China, 2007, pp. 21-25.
- [36] C. L. B. Maia, R. A. F. do Carmo, F. G. de Freitas, G. A. L. de Campos, and J. T. de Souza, "A Multi-Objective Approach for the Regression Test Case Selection Problems," in *Proc. XLI Brazilian Symposium of Operational Research*, 2009, pp. 1824-1835.
- [37] P. McMinn, "Search-based Software Test Data Generation: A Survey," *Software Testing, Verification and Reliability*, vol. 14, no. 2, pp. 105-156, 2004.
- [38] G. H. L. Pinto and S. R. Vergilio, "A Multi-Objective Genetic Algorithm to Test Data Generation," in *Proc. ICTAI*, Arras, France, 2010, pp. 129-134.
- [39] K. Praditwong, M. Harman, and X. Yao, "Software Module Clustering as a Multi-Objective Search Problem," *IEEE Transactions on Software Engineering*, vol. 37, no. 2, pp. 264-282, March/April 2011.
- [40] O. Räihä, "A survey on search-based software design," *Computer Science Review*, vol. 4, pp. 203-249, 2010.
- [41] O. Räihä, K. Koskimies, and E. Mäkinen, "Generating Software Architecture Spectrum with Multi-Objective Genetic Algorithms," in *Proc. NaBIC*, 2011, pp. 29-36.
- [42] D. Rodríguez, M. Ruiz, J. C. Riquelme, and R. Harrison, "Multiobjective Simulation Optimisation in Software Project Management," in *Proc. GECCO*, Dublin, Ireland, 2011, pp. 1883-1890.
- [43] F. Sarro, F. Ferrucci, and C. Gravino, "Single and Multi Objective Genetic Programming for Software Development Effort Estimation," in *Proc. SAC*, Trento, Italy, 2012, pp. 1221-1226.
- [44] A. S. Sayyad, T. Menzies, and H. Ammar, "On the Value of User Preferences in Search-Based Software Engineering: A Case Study in Software Product Lines," in *Proc. ICSE*, San Francisco, USA, 2013.
- [45] C. L. Simons and I. C. Parmee, "User-centered, Evolutionary Search in Conceptual Software Design," in *Proc. CEC*, Hong Kong, China, 2008, pp. 869-876.
- [46] C. L. Simons, I. C. Parmee, and R. Gwynllyw, "Interactive, evolutionary search in upstream object-oriented class design," *IEEE Transactions on Software Engineering*, vol. 36, no. 6, pp. 798-816, Nov/Dec 2010.
- [47] V. Veerappa and E. Letier, "Understanding Clusters of Optimal Solutions in Multi-Objective Decision Problems," in *Proc. RE*, Trento, Italy, 2011, pp. 89-98.
- [48] H. Wada, P. Champrasert, J. Suzuki, and K. Oba, "Multiobjective Optimization of SLA-Aware Service Composition," in *Proc. IEEE Workshop on Methodologies for Non-functional Properties in Services Computing*, Honolulu, USA, 2008, pp. 368-375.
- [49] Z. Wang, T. Chen, K. Tang, and X. Yao, "A Multi-objective Approach to Redundancy Allocation Problem in Parallel-series Systems," in *Proc. CEC*, Trondheim, Norway, 2009, pp. 582-589.
- [50] J. Wang and Y. Hou, "Optimal Web Service Selection based on Multi-Objective Genetic Algorithm," in *Proc. ISCID*, 2008, pp. 553-556.
- [51] Z. Wang, K. Tang, and X. Yao, "A Multi-objective Approach to Testing Resource Allocation in Modular Software Systems," in *Proc. CEC*, Hong Kong, China, 2008, pp. 1148-1153.
- [52] Z. Wang, K. Tang, and X. Yao, "Multi-Objective Approaches to Optimal Testing Resource Allocation in Modular Software Systems," *IEEE Transactions on Reliability*, vol. 59, no. 3, pp. 563-575, September 2010.
- [53] J. L. Wilkerson, D. R. Tauritz, and J. M. Bridges, "Multi-objective Coevolutionary Automated Software Correction," in *Proc. GECCO*, Philadelphia, USA, 2012, pp. 1229-1236.
- [54] D.H. Wolpert and W.G. Macready, "No Free Lunch Theorems for Optimization," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 67-82, 1997.
- [55] T. Yano, E. Martins, and F. L. de Sousa, "Generating Feasible Test Paths from an Executable Model Using a Multi-Objective Approach," in *Proc. SBST*, Paris, France, 2010, pp. 236-239.
- [56] S. Yoo and M. Harman, "Pareto efficient multi-objective test case selection," in *Proc. ISSSTA*, 2007, pp. 140-150.
- [57] S. Yoo and M. Harman, "Using Hybrid Algorithm For Pareto Efficient Multi-Objective Test Suite Minimisation," *Systems and Software*, vol. 83, no. 4, pp. 689-701, April 2010.
- [58] S. Yoo, M. Harman, and S. Ur, "Highly Scalable Multi-Objective Test Suite Minimisation Using Graphics Cards," in *Proc. SSBSE*, Szeged, Hungary, 2011, pp. 219-236.
- [59] S. Yoo, R. Nilsson, and M. Harman, "Faster Fault Finding at Google using Multi-Objective Regression Test Optimisation," in *Proc. ESEC/FSE*, Szeged, Hungary, 2011.
- [60] Y. Zhang, E. Alba, J. J. Durillo, S. Eldh, and M. Harman, "Today/Future Importance Analysis," in *Proc. GECCO*, Portland, USA, 2010, pp. 1357-1364.
- [61] Y. Zhang, M. Harman, and S. A. Mansouri, "The Multi-Objective Next Release Problem," in *Proc. GECCO*, 2007, pp. 1129-1136.
- [62] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the strength pareto evolutionary algorithm," in *Evolutionary Methods for Design, Optimization and Control with Applications to Industrial Problems*. Athens, Greece, 2001, pp. 95-100.