Information and Software Technology 76 (2016) 31–64

Contents lists available at ScienceDirect

Information and Software Technology

journal homepage: www.elsevier.com/locate/infsof

Comparing the performance of quantum-inspired evolutionary algorithms for the solution of software requirements selection problem

A. Charan Kumari[a,*], K. Srinivas[b]

[a] The NorthCap University, Gurgaon, India
[b] Dayalbagh Educational Institute, Dayalbagh, Agra, India

ARTICLE INFO

Article history:
Received 3 November 2014
Revised 25 March 2016
Accepted 22 April 2016
Available online 23 April 2016

Keywords:
Software requirements selection
Requirements engineering
Quantum-inspired evolutionary algorithms
Search Based Software Engineering

ABSTRACT

Context: In requirements engineering phase of the software development life cycle, one of the main concerns of software engineers is to select a set of software requirements for implementation in the next release of the software from many requirements proposed by the customers, while balancing budget and customer satisfaction.
Objective: To analyse the efficacy of Quantum-inspired Elitist Multi-objective Evolutionary Algorithm (QEMEA), Quantum-inspired Multi-objective Differential Evolution Algorithm (QMDEA) and Multi-objective Quantum-inspired Hybrid Differential Evolution (MQHDE) in solving the software requirements selection problem.
Method: The paper reports on empirical evaluation of the performance of three quantum-inspired multi-objective evolutionary algorithms along with Non-dominated Sorting Genetic Algorithm-II (NSGA-II). The comparison includes the obtained Pareto fronts, the three performance metrics – Generational Distance, Spread and Hypervolume, attained boundary solutions, and size of the Pareto front.
Results: The results reveal that MQHDE outperformed other methods in producing high quality solutions; while QMDEA is able to produce well distributed solutions with extreme boundary solutions.
Conclusion: The hybridization of Differential Evolution with Genetic Algorithms coupled with quantum computing concepts (MQHDE) provided a means to effectively balance the two issues of multi-objective optimization - convergence and diversity.

1. Introduction

Software engineering is a systematic approach to the development and maintenance of the software. The main challenging drivers include scale, quality, productivity, consistency and change [1]. To meet these challenges a growing trend has begun in the recent years to reformulate the software engineering problems as search-based problems and to expend the human effort in guiding the automated search, rather than performing the search itself. As most real-world instances of these problems are NP-hard, metaheuristic search techniques are being applied for their solution. This promising and emerging Software Engineering approach is known as Search Based Software Engineering (SBSE). The term Search Based Software Engineering was coined by Harman and Jones [2] in 2001 and advocated that search-based optimization techniques could be applied right across the spectrum of software development. They provided an insight into applications of search techniques for the solution of various Software Engineering problems and since then a remarkable progress in this field has been witnessed. Search Based Software Engineering found its applications in all the phases of software development life cycle, starting from requirements engineering phase to planning, design and testing to maintenance. This paper focuses on initial stages of software development life cycle problems, centered on requirements engineering.

In the nascent stages of SBSE approach, the motive was to find an optimal or near-optimal solution for single-objective optimization problems with the metaheuristic search techniques like Hill Climbing, Simulated Annealing, Genetic Algorithms and Evolutionary Algorithms. Local search techniques have also been evolved to best suit the applications. As most problems in Software Engineering are multi-objective in nature with competing and conflicting objectives, the researchers stepped to extend the SBSE approach to address the multiple objectives. Accordingly many of the problems in Software Engineering were reformulated as

* Corresponding author.
E-mail address: charankumari@yahoo.co.in (A.C. Kumari).

http://dx.doi.org/10.1016/j.infsof.2016.04.010

multi-objective optimization problems, and various metaheuristic search techniques were developed to deal with these multiple objectives and reported better/realistic results in comparison to their single objective versions/approaches. Consequently the Pareto optimal approach has emerged as a natural solution to the multi-objective optimization, in which the outcome of the search is a set of non-dominated solutions. These non-dominated solutions help the decision maker to analyze the tradeoff objectives in multiple angles and to choose the solution that best suits his need.

In today's scenario, change is very rapid and therefore one of the challenges for software engineering is to accommodate and embrace this change. The radical increase in the demands and requirements of the customers, in addition to the size and complexities of the software, it is becoming a tough task for any software engineer to strike a balance between budget and customers satisfaction. The problem of finding a set of requirements to be included in the next release of a software product by minimizing the cost and maximizing the customer satisfaction is termed as Next Release Problem (NRP) in the literature and was formulated by Bagnall et al. [3] in 2001. Due to the relevance of the problem and need of the time, the NRP has been reformulated as a Multi-Objective Next Release Problem (MONRP) by Zhang et al. [4]. Since then many researchers experimented on MONRP problem with different metaheuristics (as will be presented in the last section on Related Work) to find a 'good' Pareto front that can produce high quality and well distributed solutions and reported their results on synthetic and real-application data sets.

Much of the previous work has been focused on development of formulations of the problem and their implementations using well experimented metaheuristic search techniques and their variations. The rising interest of the researchers not only visualizes a transformation from the traditional methodologies to more and more sophisticated and automated techniques but also accentuates the need for experimentation with new technologies for requirements selection process. Though the said problem was basically defined for requirements engineering, the problem is equally applicable to many businesses as every business in today's scenario should adapt to the changing needs of the customers while balancing their resources. Consequently further experimentation with the latest methodologies to improve the quality of solutions will definitely benefit the software engineering community in specific and metaheuristic community in general.

The single-objective version of software requirements selection problem has been solved using Simulated Annealing, Genetic Algorithm, Ant Colony Optimization etc. Due to the inherent nature of the problem, it has been formulated into its multi-objective version and experimented mostly with NSGA-II, MoCell, PAES in the literature. After experimentation it has been reported that MoCell outperforms NSGA-II in terms of range of solutions covered, while NSGA-II is able to obtain better solutions than MoCell in large instances [5]. In another study, it has been reported that the three algorithms show three different kinds of conclusions: NSGA-II is the technique computing the highest number of optimal solutions, MoCell provides the product manager with the widest range of different solutions, and PAES is the fastest technique with the least accurate results [6]. The results of the experiments [5,6] are indeed encouraging towards multi-objective optimization and it has been observed from the results that there is a scope for improving the quality of solutions in terms of simultaneous attainment of convergence and diversity and thus to obtain solutions with minimum cost and maximum customer satisfaction.

Intuitively, the solution to any multi-objective optimization problem with conflicting objectives produces a set of trade off solutions unlike single objective optimization. Consequently dealing with a large number of solutions is rather difficult and needs careful decision-making often associated with non-technical and qualitative aspects coupled with experience. And also from a practical perspective, any decision maker is interested in a single best solution rather than multiple alternative solutions. The purpose of the proposed techniques is to find the best potential/promising solutions while simultaneously meeting the other performance criteria of multi-objective optimization (the particular standard of the true quality of optimal solution sets) [7].

To achieve these objectives the present study has experimented with the quantum-inspired evolutionary algorithms. The motivation behind the use of quantum-inspired techniques is their ability to explore and exploit the search space effectively in such a way that they can obtain good number of non-dominated solutions which can converge better to the Pareto optimal front with a wide diversity. They have produced good results on optimization problems in different fields [8,9]. Also researchers implemented these algorithms for the solution of knapsack problem (of which NRP is an instance) and reported better results [10–12]. By applying quantum mechanical principles such as qubit representation and superposition of states, quantum-inspired algorithms can simultaneously process large numbers of quantum states. The qubit representation ensures better population diversity, as each qubit chromosome represents a linear superposition of binary states in the search space probabilistically. The developed algorithms and their features are presented in detail in Section 4.

We have also experimented with Non-dominated Sorting Genetic Algorithm-II (NSGA-II) and used it as a baseline for comparison, as it was widely experimented on MONRP in the literature and also reported better results. Moreover even though latest MOEAs are available in the literature they have either not been implemented on the said problem or have not used the same benchmark data sets so they cannot be used for comparison. The experimentation is based on the benchmark data sets as described in the literature for the sake of feasibility in estimating the quality of solutions to some extent. In order to demonstrate the practical relevance and applicability of the results the approach has been implemented on a real-world banking application obtained from a reputed multi-national software development company.

The main objective of the paper is to provide "high quality" solutions to the software engineers and to assist them in selecting the software requirements, which is a major challenging task in the present business scenario due to the radical increase in the number of requirements and their dependencies. The contributions are in terms of providing latest and most effective search techniques to support and improve requirements selection and optimization process, which is useful for decision making in the requirements engineering phase of the software development process. The potential contributions of the paper can be summarized as follows:

- Development of novel multi-objective metaheuristic search techniques with the following capabilities
  - (a) Very effective exploration and exploitation of the search space - main objective of optimization
  - (b) Maintaining the right balance between the convergence and diversity - main target of multi-objective optimization
  - (c) Providing a large number of good alternatives to the decision maker - a motive of multi-objective optimization
  - (d) Automatic systematic tweaking of the control parameters during the execution of the algorithm – to make it more generic
- Testing of the techniques on benchmark datasets (with and without interactions) and to compare their performances based on quality, distribution of the solutions and the number of solutions provided
- Demonstration of the efficacy of the developed algorithms on a real-world instance of the problem for practical relevance and applicability