Empirical Software Engineering (2022) 27: 144
https://doi.org/10.1007/s10664-022-10127-4

On the preferences of quality indicators for multi-objective search algorithms in search-based software engineering

Jiahui Wu[1] · Paolo Arcaini[2] · Tao Yue[1,3] · Shaukat Ali[3] · Huihui Zhang[4,5]

Accepted: 24 January 2022 / Published online: 6 August 2022

Abstract

Multi-Objective Search Algorithms (MOSAs) have been applied to solve diverse Search-Based Software Engineering (SBSE) problems. In most cases, SBSE users select one or more commonly used MOSAs (for instance, Nondominated Sorting Genetic Algorithm II (NSGA-II)) to solve their search problems, without any justification (i.e., not supported by any evidence) on why those particular MOSAs are selected. However, when working with a specific multi-objective SBSE problem, users typically know what kind(s) of qualities they are looking for in solutions. Such qualities are represented by one or more Quality Indicators (QIs), which are often employed to assess various MOSAs to select the best MOSA. However, users usually have limited time budgets, which prevents them from executing multiple MOSAs and consequently selecting the best MOSA in the end. Therefore, for such users, it is highly preferred to select only one MOSA since the beginning. To this end, in this paper, we aim to assist SBSE users in finding appropriate MOSAs for their experiments, given their choices of QIs or quality aspects (e.g., *Convergence*, *Uniformity*). To achieve this aim, we conduct an extensive empirical evaluation with 18 search problems from a set of real-world, industrial, and open-source case studies, to study preferences among commonly used QIs and MOSAs in SBSE. We observe that each QI has its own specific most-preferred MOSA and vice versa; NSGA-II and Strength Pareto Evolutionary Algorithm 2 (SPEA2) are the most preferred MOSAs by QIs; no QI is the most preferred by all the MOSAs; the preferences between QIs and MOSAs vary across the search problems; QIs covering the same quality aspect(s) do not necessarily have the same preference for MOSAs. Based on our results, we provide discussions and guidelines for SBSE users to select appropriate MOSAs based on experimental evidence.

Keywords Search-based software engineering · Quality indicator · Multi-objective search algorithm

Communicated by: Aldeida Aleti, Annibale Panichella and Shin Yoo

This article belongs to the Topical Collection: *Advances in Symposium on Search-Based Software Engineering (SSBSE)*

✉ Tao Yue
taoyue@ieee.org

Extended author information available on the last page of the article.

1 Introduction

Search-Based Software Engineering (SBSE) (Harman et al. 2012) has been proposed to solve optimization problems for finding a suitable balance among multiple competing and potentially conflicting objectives in the software engineering domain (e.g., test generation (McMinn 2004), requirement prioritization (Achimugu et al. 2014), and test optimization (McMinn 2011)). Thus, SBSE problems are often multi-objective and Multi-Objective Search Algorithms (MOSAs) are widely used to solve them. Since the solutions computed by different MOSAs are not directly comparable, Quality Indicators (QIs) are often used to evaluate them with a numerical score and so make them comparable (Li and Yao 2019). Since many MOSAs and QIs are available, SBSE users (i.e., practitioners and researchers) often wonder which MOSA is appropriate for solving their specific search problem (Durillo et al. 2009b; Ramírez et al. 2014; Wang et al. 2015; Zhang et al. 2020).

In order to select a MOSA to solve a search problem, SBSE practitioners have two options. One option is to execute multiple MOSAs, compare their solutions with a QI, and then select the best MOSA according to the used QI. The survey of Sayyad and Ammar (2013) reports that, in some of the investigated publications, SBSE practitioners indeed compared MOSAs against each other with certain QIs. In some cases, they choose QIs on the basis of the *quality aspects* (e.g., *Convergence* and *Uniformity*) the QIs represent. A classification of such QIs is reported by Li and Yao (2019). So, as the first observation, we can claim that "*SBSE practitioners are often aware of qualities they desire from solutions produced by a MOSA*".

However, an SBSE practitioner usually has a limited time budget, which does not always allow experimenting with multiple MOSAs. Therefore, SBSE practitioners often end up in selecting only one MOSA for their experiments. Such a choice is usually not justified, and it is only based on the popularity of the MOSA. This phenomenon is also reported in the same survey of Sayyad and Ammar (2013), which shows that most of the publications included in the survey do not provide justifications on why a given MOSA has been chosen for their experiments. So, as second observation, we can claim that "*SBSE practitioners often do not have time to run multiple experiments with different MOSAs, and so end up in selecting a MOSA without a proper justification*".

Based on these observations, we conclude that it would be desirable that SBSE practitioners could directly select a single MOSA that produces solutions entailing the qualities they desire. So, in this paper, we aim to guide SBSE practitioners to select a MOSA on the basis of their preferences. More specifically, our application context can be summarized as follows: Given a QI or a quality aspect, we suggest a MOSA that is highly likely to give desired solutions according to the selected QI or quality aspect. This means that SBSE practitioners do not need to run large-scale experiments to find the best MOSA for their specific problem, but they can select a MOSA that is highly likely to give solutions with the desired qualities.

In the literature, there are studies investigating relationships of QIs and their characteristics. For instance, in our previous paper (Ali et al. 2020), we analyzed agreements among QIs commonly used in SBSE, with the aim of providing users with a set of guidelines on selecting QIs for their SBSE applications. In comparison, in this paper, we study preferences of QIs for MOSAs with the aim of suggesting which MOSA to choose given a QI (Ali et al. 2020) (see more detailed comparison in Section 7). Similarly, Li and Yao (2019) surveyed 100 QIs that have been used in the evolutionary computation domain with the aim of studying their strengths and weaknesses. In the current SBSE literature, however, relationships between QIs and MOSAs are still insufficiently studied.

In this paper, we present an extensive empirical evaluation to study relationships between QIs and MOSAs, with 18 search problems from a set of industrial, real-world, and open-source case studies. By studying such relationships, we provide evidence about which QIs prefer which MOSAs and vice versa. More specifically, we observe that each QI has its own specific most-preferred MOSA (e.g., Hypervolume (HV) prefers NSGA-II the most), and vice versa. Moreover, SPEA2 and NSGA-II are the most preferred MOSAs by QIs, whereas no QI is the most preferred by all the MOSAs. Besides, we find that the preferences between QIs and MOSAs vary across the search problems, and QIs covering the same quality aspect(s) do not necessarily have the same preference for MOSAs. NSGA-II is the most preferred for *Spread*, *Uniformity*, and *Cardinality*, whereas SPEA2 is the most preferred by all the quality aspects. Based on such observations, we also present guidelines on choosing a MOSA for a given QI or quality aspect.

This paper is an extension of our conference paper (Ali et al. 2020). With respect to the conference version, our key contributions include: 1) Extending the experiments with seven new real-world search problems from two new SBSE applications (Zhang et al. 2020; Zhang et al. 2019); these two applications are related to uncertainty-wise requirements prioritization, and test case generation/minimization; 2) Adding a new research question that studies the preference relationships between QIs and MOSAs by also taking into account MOSAs' preferences on QIs. This analysis allows us to further refine our findings and understand better whether a MOSA suggested for an SBSE user with our recommendation also covers other quality aspects (that may or may not be desirable for the SBSE user); 3) Explaining the results in detail–wherever possible–, also giving evidence from the literature to support the explanation; and 4) Providing new insights and guidelines based on the extended experiments. For instance, each MOSA has its own specific most-preferred QI (e.g., SPEA2 prefers Generational Distance (GD) the most); no QI is the most preferred by all the MOSAs; the preferences between QIs and MOSAs vary across the search problems.

The rest of the paper is organized as follows: Section 2 introduces the relevant background. Section 3 presents the design of our empirical evaluation, and Section 4 describes our experiment results and analyses. Section 5 presents the discussion and our recommendations, while the threats to validity are presented in Section 6. Section 7 relates our work with the literature. Finally, Section 8 concludes the paper.

2 Background

In this section, we discuss multi-objective optimization in Section 2.1, and introduce the selected QIs and MOSAs in Sections 2.2 and 2.3, respectively.

2.1 Multi-Objective Optimization

SBSE approaches (Harman et al. 2012) use search algorithms (e.g., a genetic algorithm (GA)) to solve different software engineering problems (e.g., test case selection, requirement prioritization). Guided by a fitness function, a search algorithm selects the best solutions from the entire search space of candidate solutions in a cost-effective manner, instead of doing an exhaustive search. Thus, to solve an SBSE problem, the key is to formulate the problem as an optimization (or search) problem, design an appropriate fitness function, and solve it with a suitable search algorithm.

In practice, many software engineering problems have multiple goals, i.e., a multiple objective problem must be solved. Suppose that there is a multi-objective optimization problem (e.g., ρ) consisting of m objectives to be maximized or minimized. ρ is defined over an n-dimensional decision variable vector over a universe X as:

$$\mathbf{x} = (x_1, \dots, x_n)$$

Moreover, we define an objective function as:

$$f_i(\mathbf{x}), \quad i = 1, \dots, m$$

We then define the mapping of the decision variable vector $\mathbf{x}$ to the objective function vector as:

$$F(\mathbf{x}) = (f_1(\mathbf{x}), \dots, f_m(\mathbf{x}))$$

Let us assume, without loss of generality, that all objective functions are minimization ones.[1] Let us denote with S ($S \subseteq X$) the set of returned solutions for ρ. Given two solutions $A, B \in S$, A dominates B ($A \succ B$) if and only if:

$$(\forall i \in \{1, \dots, m\}\colon f_i(A) \leq f_i(B)) \wedge (\exists j \in \{1, \dots, m\}\colon f_j(A) < f_j(B))$$

i.e., for all the optimization objectives, A is never worse than B and A is better than B for at least one objective. The set of solutions that are not dominated by others forms a *Pareto front*. Given the set of computed solutions S, we formally define a Pareto front $PF(S)$ as:

$$PF(S) = \{s_i \in S \mid (\nexists s_j \in S\colon s_j \succ s_i)\}$$

For a given multi-objective optimization problem, we can use MOSAs to find solutions. Each execution of a MOSA produces one Pareto front. Thus, for more than one repetition (e.g., k repetitions) of a MOSA, we will obtain k Pareto fronts $PF^c{}_1, \dots, PF^c{}_k$. We will call them *computed Pareto fronts* (Durillo and Nebro 2011).

To evaluate the quality of the computed Pareto fronts, it is usually required to have the *optimal Pareto front* (also called the *true Pareto front*), which includes all non-dominated solutions over the given search space (Knowles et al. 2006). However, the optimal Pareto front is often unknown for complex search problems. Hence, in practice, a *reference Pareto front* is used; see the report of Li and Yao (2019) for a discussion. One possible way is to use all Pareto fronts computed by all the MOSAs applied to the problem:

$$PF^{ref} = \{s_i \in PF^{all} \mid (\nexists s_j \in PF^{all}\colon s_j \succ s_i)\}$$

where $PF^{all} = \bigcup_{j=1}^{t} \bigcup_{i=1}^{k_j} PF^c{}_{i,j}$ is the union of the k_j Pareto fronts computed by each of the t applied MOSAs.

2.2 Quality Indicators

Usually, the quality of a computed Pareto front is evaluated from four aspects (Li and Yao 2019): *Convergence*, *Spread*, *Uniformity*, and *Cardinality*.

- ***Convergence*** indicates how close the computed Pareto front is to the optimal Pareto front;
- ***Spread*** reflects the coverage of the computed Pareto front;
- ***Uniformity*** represents how uniformly the solutions are distributed in the computed Pareto front;

[1] A maximization objective can be converted into a minimization one by negating it.

– ***Cardinality*** indicates the number of solutions in the computed Pareto front.

To evaluate and compare Pareto fronts, many Quality Indicators (QIs) have been proposed (Knowles and Corne 2002). QIs assess differences among Pareto fronts by quantifying each Pareto front with a real number (Zitzler et al. 2003). Each QI can consider one or more quality aspects (e.g., *Convergence*).

We selected 8 commonly used QIs in SBSE to conduct our experiments, i.e., Hypervolume (HV) (Shang et al. 2021; Zitzler and Thiele 1998), Inverted Generational Distance (IGD) (CoelloCoello and ReyesSierra 2004), Epsilon (EP) (Zitzler et al. 2003), Generational Distance (GD) (Van Veldhuizen and Lamont 1998), Generalized Spread (GS) (Zhou et al. 2006), Euclidean Distance (ED) (Zeleny 1973), Pareto Front Size (PFS) (VanVeldhuizen 1999), and Coverage (C) (Fieldsend et al. 2003). The brief definitions of these QIs are as follows.

HV measures the volume of the objective space covered by PF^c w.r.t. a *reference point* w; the reference point can be determined, for example, by using the worst objective function values among all the solutions in PF^{ref}. For each solution $s_i \in PF^c$, $hc(s_i)$ is a hypercube with s_i and w as diagonal corners. HV is defined as:

$$HV(PF^c) = volume\left(\bigcup_{s_i \in PF^c} hc(s_i)\right)$$

IGD is the distance from the solutions in PF^{ref} to the nearest solutions in PF^c. Given a solution s, assuming $d(s, PF^c)$ is the minimum Euclidean distance from the solution s to the PF^c, IGD can be defined as:

$$IGD(PF^c) = \frac{\sqrt{\sum_{s_i \in PF^{ref}} d(s_i, PF^c)^2}}{|PF^{ref}|}$$

EP indicates the shortest distance that every solution in PF^c should be translated to dominate PF^{ref}. We define epsilon-dominance $\succ_\epsilon$ as: given solutions $A = (a_1, \ldots, a_m)$ and $B = (b_1, \ldots, b_m)$, $A \succ_\epsilon B$ iff $\forall i \in \{1, \ldots, m\}$: $a_i < b_i + \epsilon$. Hence, EP can be defined as:

$$EP(PF^c) = inf\{\epsilon \in \mathbb{R} \mid (\forall x \in PF^{ref}, \exists y \in PF^c : y \succ_\epsilon x)\}$$

GD represents the Euclidean distance between the solutions in PF^c and the nearest solutions in PF^{ref}. Similarly as IGD, given a solution s and taking $d(s, PF^{ref})$ as the minimum Euclidean distance from the solution s to the PF^{ref}, GD is defined as:

$$GD(PF^c) = \frac{\sqrt{\sum_{s_i \in PF^c} d(s_i, PF^{ref})^2}}{|PF^c|}$$

GS reflects the extent of spread for the solutions in PF^c. Let $\mathbf{e} = (e_1, \ldots, e_m)$ be the extreme solution of the PF^{ref}, where e_i is the maximum value of objective function f_i. Also, we define $d(e_i, PF^c)$ as the minimum Euclidean distance from e_i to PF^c. Given a solution s, let $id(s, PF^c) = d(s, PF^c \setminus \{s\})$ be the minimum distance of the solution s from all the other solutions in PF^c, and $\overline{id}$ be the mean value of $id(s, PF^c)$ across all the solutions of PF^c. GS is defined as:

$$GS(PF^c) = \frac{\sum_{i=1}^{m} d(e_i, PF^c) + \sum_{s_j \in PF^c} |id(s_j, PF^c) - \overline{id}|}{\sum_{i=1}^{m} d(e_i, PF^c) + |PF^c| * \overline{id}}$$

ED introduces the Euclidean distance between the *ideal solution* and the nearest solution in PF^c. The ideal solution $\mathbf{e}_{ideal}$ consists of all the optimal values of each objective obtained from the solutions in PF^c. Let $d(\mathbf{e}, PF^c)$ be the minimum Euclidean distance from $\mathbf{e}$ to PF^c. ED can be defined as:

$$ED(PF^c) = d(\mathbf{e}_{ideal}, PF^c)$$

PFS counts the number of solutions in PF^c. It can be defined as:

$$PFS(PF^c) = |PF^c|$$

C means the solution coverage of PF^c to PF^{ref}. It is defined as:

$$C(PF^c) = \frac{|PF^c \cap PF^{ref}|}{|PF^{ref}|}$$

Table 1 reports the characteristics of the selected QIs, and the quality aspects they cover as identified by Li and Yao (2019).

Table 1 Characteristics and relevant quality aspects of the selected QIs

QI	Characteristics	Quality Aspect			
		Convergence	Spread	Uniformity	Cardinality
HV	(1) focus on knee points of a solution set (2) the settings of the reference point will affect its evaluation results	+	+	-	+
IGD	(1) need a dense and evenly distributed solution set	+	+	-	-
EP	(1) only involve the largest difference of a solution in any set (2) the distribution of the solutions depends on the shape of the Pareto front (3) focus on complementary aspects relative to other indicators	+	+	-	-
GD	(1) be more accurate in terms of measuring the closeness of solution sets to the optimal Pareto front (2) be sensitive to outliers (3) be easily affected by the size of the solution set	+			
GS	(1) emphasize distribution (2) consider extreme solutions		-	+	
ED	(1) focus on the shortest Euclidean distance between the computed Pareto front and the ideal solution	-			
PFS	(1) simply count the number of non-dominated solutions (2) unable to compare solution sets (3) easily evaluate repeated solution sets				+
C	(1) unable to account for the front difference and the uniformity distribution of front points (2) emphasize the domination of the solution sets	-			-

A cell with a ”+” signifies that a particular quality aspect is fully represented by the QI, where ”-” signifies a partial representation

2.3 Multi-Objective Search Algorithms

MOSAs are often used to solve search problems with more than one objective, not limited to a specific domain or problem (Harman et al. 2012). For our experiments, we choose six commonly used MOSAs in SBSE. Table 2 provides a summary of their features, and their brief description is as follows.

Nondominated Sorting Genetic Algorithm II (NSGA-II) (Deb et al. 2002) is a multi-objective genetic algorithm that classifies and sorts the population into several non-dominated fronts by calculating the crowding distance and applying selection, crossover, and mutation operators to generate a new population. It employs a fast non-dominated sorting procedure to reduce the computational complexity, uses an elitist-preserving approach to improve performance, and applies a parameterless niching operator for guaranteeing diversity.

Speed-constrained Multi-objective Particle Swarm Optimization (SMPSO) (Nebro et al. 2009) is a multi-objective particle swarm optimization algorithm. It calculates the crowding distance to select the non-dominated solutions, utilizes an external archive to store them, and uses polynomial mutation operators to accelerate the convergence and the velocity constriction mechanism to limit the velocity of the particles.

Strength Pareto Evolutionary Algorithm 2 (SPEA2) (Zitzler et al. 2002) is an evolutionary algorithm that measures the distance between a solution and its nearest neighbours to generate the non-dominated solutions and employs selection, crossover, and mutation operators to store the best solutions into an archive, which are combined to create the new population. SPEA2 applies an elitist approach for the convergence of the solutions, uses both a fine-grained fitness assignment strategy and a density estimation technique to strengthen the domination of solutions, and also adopts an enhanced archive truncation method to improve the spread of solutions.

Pareto Archived Evolution Strategy (PAES) (Knowles and Corne 2000) is a simple evolutionary algorithm for multi-objective optimization problems. It utilizes the dynamic mutation operator to explore the search space to find optimal solutions and applies an

Table 2 Selected MOSAs and their special mechanisms

MOSA	Special mechanism
NSGA-II	(1) a fast nondominated sorting procedure (2) an elitist-preserving approach (3) a parameterless niching operator (density estimation and crowded-comparison operator)
SMPSO	(1) an external archive (2) polynomial mutation operators (3) a velocity constriction mechanism
SPEA2	(1) an external archive (2) an elitist approach (3) a fine-grained fitness assignment strategy (4) a density estimation technique (5) an enhanced archive truncation method
PAES	(1) an archive (2) an elitist approach (3) a crowding procedure based on the adaptive grid mechanism
MOCell	(1) an external archive (2) an elitist algorithm (3) a feedback mechanism (4) the NSGA-II density estimator
CellDE	(1) an external archive (2) the MOCell search engine (3) a differential evolution reproductive mechanism (4) the SPEA2 density estimator

archive to maintain the found non-dominated solutions. PAES also employs an elitist approach to enhance the convergence and adopts a crowding procedure based on the adaptive grid mechanism to improve the diversity of the solutions.

Multi-Objective Cellular (MOCell) (Nebro et al. 2009) is a cellular genetic algorithm for multi-objective optimization. It uses an external archive to store non-dominated solutions, applies a density estimator based on the crowding distance of NSGA-II and utilizes a feedback mechanism to randomly replace population existing individuals with archive solutions. MOCell also employs an elitist algorithm to improve the solution convergence.

CellDE (Durillo et al. 2008) is a multi-objective cellular genetic algorithm. It replaces the typical genetic crossover and mutation operators with differential evolution reproductive operators. It combines the advantages of MOCell (good diversity in bi-objective optimization) by utilizing its search engine and Generalized Differential Evolution 3 (GDE3, good convergence in three-objective optimization). CellDE also adopts an external archive to store the non-dominated solutions found during the search and applies the SPEA2 density estimator when the archive becomes full.

3 Design of Empirical Evaluation

The procedure of conducting our empirical study is shown in Fig. 1. All the data, scripts, and results are available online (Wu et al. 2021). It consists of six steps. For some search problems, *Steps 1-2* (marked in gray) were performed in the works (Pradhan et al. 2016a, b, 2018, 2021; Safdar et al. 2017; Wang et al. 2015; Yue and Ali 2014) and the data of these works were already publicly available. Regarding the other search problems (Zhang et al. 2019, 2020), we re-ran them for this work. *Steps 3-6* marked in blue are new activities performed for this work.

All the data obtained from *Steps 1-2* is used in our empirical evaluation, i.e., to perform *Steps 3*, *4(a)-4(c)*, *5*, and *6(a)-6(c)*. All these steps will be described in the following subsections. We answer two research questions (RQs), described in Section 3.3. In both RQs, we use 18 search problems from 11 SBSE applications (see details in Section 3.1). The number of objectives in the search problems range from 2 to 4. All selected MOSAs and QIs are used to answer both RQs. We provide their details and settings in Section 3.2. Finally, to answer the RQs, we perform various statistical tests which are described in Section 3.4.

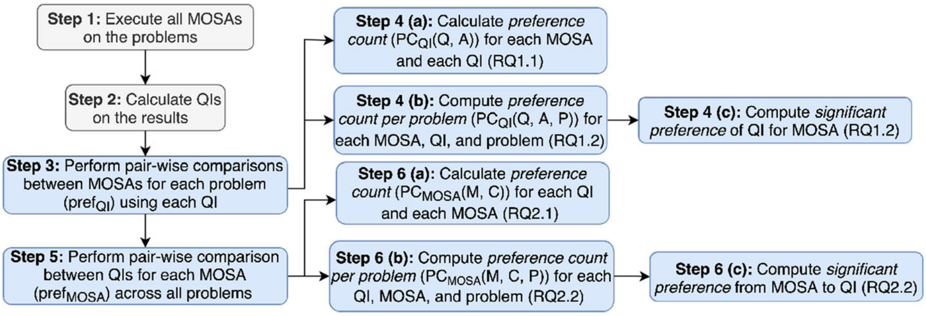

Step 1: Execute all MOSAs on the problems
Step 2: Calculate QIs on the results
Step 3: Perform pair-wise comparisons between MOSAs for each problem ($pref_{QI}$) using each QI
Step 5: Perform pair-wise comparison between QIs for each MOSA ($pref_{MOSA}$) across all problems
Step 4 (a): Calculate preference count (PC_{QI}(Q, A)) for each MOSA and each QI (RQ1.1)
Step 4 (b): Compute preference count per problem (PC_{QI}(Q, A, P)) for each MOSA, QI, and problem (RQ1.2)
Step 4 (c): Compute significant preference of QI for MOSA (RQ1.2)
Step 6 (a): Calculate preference count (PC_{MOSA}(M, C)) for each QI and each MOSA (RQ2.1)
Step 6 (b): Compute preference count per problem (PC_{MOSA}(M, C, P)) for each QI, MOSA, and problem (RQ2.2)
Step 6 (c): Compute significant preference from MOSA to QI (RQ2.2)

Fig. 1 Procedure of conducting the empirical study

3.1 Description of the Selected Case Studies

For our empirical study, we collected 18 search problems from both industrial projects and the literature, across topics of test case minimization (Wang et al. 2015), test case prioritization (Pradhan et al. 2016b, 2018), rule mining and configuration generation (Safdar et al. 2017), requirements allocation for inspection (Yue and Ali 2014), test case selection (Pradhan et al. 2016a), test case minimization (Zhang et al. 2019), uncertainty-wise requirements prioritization (Zhang et al. 2020), testing resource allocation (Pradhan et al. 2021; Wang et al. 2008), integration and test order (Guizzo et al. 2017; Pradhan et al. 2021), and software release planning (Dantas et al. 2015; Greer and Ruhe 2004). We obtained data for this empirical study by following the procedure described in Fig. 1. In the end, there were data available for 11 SBSE applications from different industrial, real-world, and open source SBSE application domains, 18 search problems with the number of optimization objectives being 2, 3, and 4.

The used search problems are summarized in Table 3 and described in the following subsections.

3.1.1 Industrial SBSE Applications

When SBSE problems and the corresponding case studies were provided by our industrial partners, these problems are classified as industrial problems.

In the past, we worked with Cisco Systems Norway to improve the cost and effectiveness of testing Video Conferencing Systems (VCSs) (Wang et al. 2015). VCSs establish video conferences between participants in different physical locations, and realize the transmission of presentations in parallel with the video conferences. However, for large-scale and complex VCSs, it is difficult to exhaustively test them due to limited time and resources; thus, their testing needs to be optimized, e.g., with test minimization and test

Table 3 Characteristics of the selected case studies

Category	SBSE Applications	# search problems	# Objectives
Industrial	Test Suite Minimization (Wang et al. 2015)	1	4
	Test Case Prioritization-1 (Pradhan et al. 2016b)	1	4
	Test Case Prioritization-2 (Pradhan et al. 2018)	3	2
	Rule Mining and Configuration Generation (Safdar et al. 2017)	1	3
Real-World	Requirements Allocation for Inspection (Yue and Ali 2014)	1	3
	Test Case Selection (Pradhan et al. 2016a)	1	4
	Test Case Minimization (Zhang et al. 2019)	4	4
	Uncertainty-Wise Requirements Prioritization (Zhang et al. 2020)	3	4
Open-Source	Testing Resource Allocation (Pradhan et al. 2021; Wang et al. 2008)	1	2
	Integration and Test Order (Guizzo et al. 2017; Pradhan et al. 2021)	1	4
	Software Release Planning (Dantas et al. 2015; Greer and Ruhe 2004)	1	3

prioritization. To this end, in our previous works, we have implemented a test suite minimization approach (Wang et al. 2015), and two test case prioritization approaches (Pradhan et al. 2016b, 2018) to improve the testing efficiency. In addition, we have implemented a search-based approach to discover faulty configurations caused by interactions among VCSs belonging to different families of VCSs (Safdar et al. 2017). We discuss these four applications below.

Test Suite Minimization (Wang et al. 2015): This problem focuses on test minimization for product lines. It aims to identify and eliminate redundant test cases from test suites to reduce the total number of test cases to be executed, thereby improving the efficiency of testing. The problem is expressed as a search problem and four cost/effectiveness objectives are defined, i.e., test minimization percentage, feature pairwise coverage, fault detection capability, and overall execution time. The corresponding evaluation was carried out on an industrial case study and 500 artificial problems of varying size and complexity.

Test Case Prioritization-1 (Pradhan et al. 2016b): This problem focuses on prioritizing test cases for testing product lines of VCSs and concentrates on achieving high coverage of configurations, test APIs, statuses, and high fault detection capability as soon as possible. For the search problem, we defined four objectives (configuration coverage, test API coverage, status coverage, and fault detection capability) and the relevant test suite consists of 211 test cases.

Test Case Prioritization-2 (Pradhan et al. 2018): For this problem, we proposed an approach for black-box dynamic test case prioritization using rule mining and multi-objective search, defined two objectives (fault detection capability and test case reliance score), and used three case studies to empirically evaluate MOSAs. Two of the three case studies include 60 and 624 test cases and the other one (consisting of 89 test cases) was from ABB Robotics for testing paint control systems of painting robots (Spieker et al. 2017).

Rule Mining and Configuration Generation (Safdar et al. 2017): To generate faulty configurations that prevent successful interactions between VCSs of different families, we identified this problem and solved it by combining multi-objective search with machine learning to mine configuration rules. We defined three objectives for search, i.e., avoiding high confidence rules with normal states, generating low confidence rules with normal states, and generating rules with abnormal states, and mined the initial set of rules based on randomly generated labeled configurations. For the case study to evaluate, we used two VCSs, 17 configuration parameters, 30 rules, and 200 initial configurations.

3.1.2 Real-World SBSE Applications

In case of the real-world SBSE problems, we identified them from the industry by working closely with our industry partners; however, the corresponding case studies were created based on real data from publicly available documents such as standards and regulations.

Requirements Allocation for Inspection (Yue and Ali 2014): Working with our industrial partner in the domain of subsea production systems, we identified this problem. The problem is about assigning requirements to different stakeholders by maximizing their familiarities to the assigned requirements while balancing the overall workload of each stakeholder. For this problem, we defined three objectives, i.e., extent of assigned requirements, familiarity of stakeholders, and overall differences of workloads, and the case study consists of 287 requirements and 10 stakeholders.

Test Case Selection (Pradhan et al. 2016a): Based on our industrial collaboration within the maritime domain, we identified a real-world and multi-objective test case selection problem for robustness testing. Within a limited time budget, it is actually not feasible to execute all test cases in the context of real-time embedded systems deployed in various maritime applications, so an effective method is needed to select cost-effective test cases. Our case study consists of 165 high-level test cases and a fitness function including one cost objective, i.e., time difference, and three effectiveness objectives, i.e., mean priority, mean probability, and mean consequence.

Test Case Minimization (Zhang et al. 2019): Considering testing of highly uncertain cyber-physical systems, we proposed four test case minimization strategies based on uncertainty theory and multi-objective search, to achieve four objectives, i.e., maximizing the average number of uncertainties, maximizing the average percentage of uncertainty space, maximizing the average uncertainty measurement, and maximizing the average percentage of unique uncertainties. With five use cases from two industrial cyber-physical system case studies, we empirically evaluated the four test case minimization strategies.

Uncertainty-wise Requirements Prioritization (Zhang et al. 2020): The stakeholders in safety-critical domains usually lack expertise related to requirements review, leading to uncertainty in cost overruns. To solve this problem, one needs to prioritize uncertain requirements by reviewing requirements with higher importance, depending more on other requirements, and higher implementation costs as early as possible. Hence, we defined four objectives: maximizing the importance of requirements, requirements dependencies, the implementation cost of requirements, and the cost overrun probability. We used three real-world datasets, i.e., the RALIC (Lim 2011), Word (Karim and Ruhe 2014; Pitangueira et al. 2016), and ReleasePlanner datasets (Karim and Ruhe 2014; Pitangueira et al. 2016) for evaluation.

3.1.3 Open Source SBSE Applications

When SBSE problems and corresponding case studies are publicly available, including the description and implementation of the problems and case studies, we consider them as open source.

Testing Resource Allocation (Pradhan et al. 2021; Wang et al. 2008): For this problem, we aimed to realize the optimal allocation of test resources to different software modules to minimize test costs (e.g., test time) and maximize the reliability of the modules. The problem has two optimization objectives, i.e., the reliability of the system and the testing cost, and adopts a system with eight modules and maximum testing resource of 10,000 hours for empirical evaluation.

Integration and Test Order (Guizzo et al. 2017; Pradhan et al. 2021): This problem generates orders for the units to be integrated and tested, and prioritizes the unit that is most needed for integration and testing by rearranging the order of the units, to minimize the stub cost. The stub refers to the simulation of a unit that has not been implemented, tested, or integrated in the software. This problem includes four objectives: number of attributes, number of operations, number of distinct return types, and number of distinct parameters. The problem uses the open source program Commons Byte Code Engineering Library (BCEL) to create, manipulate, and analyze binary Java class files, which includes 45 classes with 289 dependencies.

Software Release Planning (Dantas et al. 2015; Greer and Ruhe 2004): This problem is about deciding which features to be implemented next in the subsequent release of software. Such problem is common in incremental and iterative software development. In particular, it considers user preference to guide the search. It has three objectives, i.e., technical precedence inherent in the requirements, conflicting priorities as determined by the representative stakeholders, and balance between required and available effort. A software with 50 requirements, 5 releases, and 4 clients is used by the problem.

3.2 Settings of MOSAs and QIs

As indicated by *Step 1* in Fig. 1, we first obtained data from 100 runs of the selected MOSAs. Note that some MOSAs are not applicable to some search problems that have particular requirements on, e.g., the data type. For example, the uncertainty-wise requirements prioritization problem (Zhang et al. 2020) requires to achieve an integer permutation strategy, and this can not be encoded in CellDE. The chosen MOSAs were run using parameter settings of the MOSAs based on the previous experiments (Pradhan et al. 2016a, b, 2018, 2021, Safdar et al. 2017; Wang et al. 2015; Yue and Ali 2014; Zhang et al. 2020, 2019). These parameter settings can be found in Table 4. Moreover, the dataset used in this work also includes computed values of the eight QIs for assessing the selected MOSAs (*Step 2* in Fig. 1). These QIs are HV, IGD, EP, GD, GS, ED, PFS, and C (see Section 2.2).

As indicated in *Step 3* in Fig. 1, based on QI results, we performed relevant statistical tests to compare each pair of MOSAs using each QI. Results of these tests reveal which MOSA performed significantly better than another one with respect to a particular QI. Note that all the MOSAs performed significantly better than Random Search (RS); therefore, we did not include the results of RS. These results were then used in our empirical evaluation reported in this paper; namely, we used them to perform *Steps 3*, *4(a)-(c)*, *5*, and *6(a)-(c)* in Fig. 1, in order to answer the RQs defined in the next subsection.

3.3 Research Questions

- ***RQ1***: What is a QI's preference for a specific MOSA? This RQ aims to help SBSE users select a MOSA for solving an SBSE problem. This RQ can be addressed via the following two sub-RQs:
  - ***RQ1.1***: How frequently does a QI prefer a particular MOSA? This RQ studies the percentage of times that a QI prefers a particular MOSA by ignoring differences of the search problems when studying pairs of MOSAs, with the aim of understanding the overall preferences of a QI.

Table 4 Parameter settings for the selected MOSAs

Parameter	Settings
Population Size	100 for All but PAES
Neighborhood	MOCell and CellDE: 1-hop neighbors (8 surrounding solutions)
Parents Selection	All but PAES and SMPSO: Binary tournament + binary tournament
Recombination	PAES and SMPSO: None; CellDE: Differential evolution; Rest: Simulated binary
Crossover Rate	All but PAES and SMPSO: 0.9
Mutation	All but CellDE: polynomial, mutation rate=1.0/n
Archive Size	MOCell and PAES: 100
Max Generation	All: 25000
Times of Run	All: 100

  - ***RQ1.2***: How frequently does a QI prefer a particular MOSA across the different search problems? This RQ studies the preferences of QIs across the problems when studying pairs of MOSAs, whereas in RQ1.1 we aim to study preferences while ignoring the differences of the problems.

- ***RQ2***: What is a MOSA's preference for a specific QI? This RQ provides further insights on the relations between QIs and MOSAs, results of which could be consulted by practitioners in their selections of MOSAs. Indeed, results from RQ1 allow to select a MOSA M that better produces solutions with the desired qualities of a given QI Q. However, it does not guarantee that M only targets the qualities of Q. It could be that M also covers other qualities, which might or might not be desirable by the practitioner. A mutual preference between a MOSA and a QI, instead, demonstrates a stronger binding between solutions produced by the MOSA and the type of solutions preferred by the QI.

  This RQ can be answered via the following two sub-RQs:

  - ***RQ2.1***: How frequently does a MOSA prefer a particular QI? This RQ studies the percentage of times that a MOSA prefers a particular QI by ignoring the differences of the search problems when studying pairs of QIs. This RQ helps understanding the overall preferences of a MOSA.
  - ***RQ2.2***: How frequently does a MOSA prefer a particular QI across the different search problems? This RQ studies the preferences of MOSAs across the problems when studying pairs of QIs, whereas in RQ2.1 we aim to study preferences while ignoring the differences of the problems.

3.4 Evaluation Metrics

We define a set of evaluation metrics to answer the two RQs in Section 3.3.

3.4.1 Statistical Analysis

The *Wilcoxon signed-rank* test and the *Vargha and Delaney* $\hat{A}_{12}$ statistics are used in our analyses (*Step 4(c)* and *Step 6(c)* in Fig. 1). We choose these two statistical tests by following a well-established guide in the SBSE literature (Arcuri and Briand 2011). More specifically, since our data is in interval-scale, the guide suggests using the *Wilcoxon signed-rank* and $\hat{A}_{12}$ as the effect size measure–both are non-parametric tests for interval-scale data. The *Wilcoxon signed-rank* test determines whether a statistically significant difference exists between two distributions of matched samples (Wilcoxon 1992). In the experiments, we set its significance level to 0.05. When comparing two MOSAs (or QIs) A and B in our experiments, if the p-value computed by the *Wilcoxon signed-rank* test is < 0.05, then it means that there are significant differences between A and B. The *Vargha and Delaney* $\hat{A}_{12}$ statistics is used as the effect size measure (Vargha and Delaney 2000). It reports the magnitude of differences between two groups A and B. An $\hat{A}_{12}$ value of 0.5 indicates that A and B are not different. Moreover, on the one hand, the higher the $\hat{A}_{12}$ value than 0.5, the higher the magnitude of differences between A and B, in terms of A being better than B. On the other hand, the lower the $\hat{A}_{12}$ value than 0.5, the higher the magnitude of differences between A and B, in terms of B being better than A.

We present evaluation metrics and the statistical tests that we employ specifically for each RQ in Sections 3.4.2 and 3.4.3, respectively.

3.4.2 RQ1 – Studying QI's preference for MOSAs

To answer RQ1, first, we perform pair-wise comparisons between MOSAs for each problem using each QI (*Step 3* in Fig. 1). Let Q be a quality indicator, M_1 and M_2 two MOSAs, and P a search problem. We compare the values of QI Q for 100 runs of MOSAs M_1 and M_2 over problem P using the *Mann-Whitney U* test and *Vargha and Delaney* $\hat{A}_{12}$ statistics as the effect size measure.The tests were selected by following the guide of Arcuri and Briand (2011) on conducting experiments with randomized algorithms. The result of the statistical test is recorded with the predicate $pref_{QI}$. Specifically, if the p-value computed by the *Wilcoxon signed-rank* test is less than 0.05 and $\hat{A}_{12}$ is greater than 0.5, then it means that M_1 is significantly better than M_2 with respect to Q. In this case, $pref_{QI}(Q, M_1, M_2, P) = true$. Similarly, when a p-value is less than 0.05 and $\hat{A}_{12}$ is less than 0.5, it means that M_2 is significantly better than M_1, and so $pref_{QI}(Q, M_2, M_1, P) = true$. Finally, a p-value greater than or equal to 0.05 implies no significant differences between M_1 and M_2 with respect to Q, i.e., $pref_{QI}(Q, M_1, M_2, P) = pref_{QI}(Q, M_2, M_1, P) = false$. Note that $pref_{QI}(Q, M_1, M_2, P) = true$ implies $pref_{QI}(Q, M_2, M_1, P) = false$. If $pref_{QI}(Q, M_1, M_2, P) = false$ and $pref_{QI}(Q, M_2, M_1, P) = false$, it means that Q does not have any significant preference among the two MOSAs.

RQ1.1 In order to answer RQ1.1 (*Step 4(a)* in Fig. 1), we introduce the following measure. Let *MOSAs* be the set of *MOSAs*, *Problems* the set of search problems, and Q a quality indicator. We define the *preference count* $PC_{QI}(Q, M)$ as the percentage of times Q prefers MOSA M when compared to another MOSA in any problem, formally[2]:

$$PC_{QI}(Q, M) = \frac{\left| \bigcup_{P \in Problems} \{M' \in (MOSAs \setminus \{M\}) \mid pref_{QI}(Q, M, M', P)\} \right|}{(|MOSAs| - 1) \times |Problems|} \tag{1}$$

The rationale is that if a QI Q consistently prefers a MOSA M (when compared with another MOSAs and for different problems), it means that M tends to produce solutions that have the quality aspects assessed by Q. The higher $PC_{QI}(Q, M)$ is, the higher the probability is that, also on new problems, M will produce solutions preferred by Q.

RQ1.2 In order to answer RQ1.2, first, we compute the *preference count per problem* $PC_{QI}(Q, M, P)$ defined as follows (*Step 4(b)* in Fig. 1):

$$PC_{QI}(Q, M, P) = \frac{|\{M' \in (MOSAs \setminus \{M\}) \mid pref_{QI}(Q, M, M', P)\}|}{|MOSAs| - 1} \tag{2}$$

Second, we also perform, for each QI Q, pair-wise comparisons of $PC_{QI}(Q, M, P)$ of the selected MOSAs across the search problems (*Step 4(c)* in Fig. 1). To do this, we apply the *Wilcoxon signed-rank* test and the *Vargha and Delaney* $\hat{A}_{12}$ effect size measure as described in Section 3.4.1.

The results of these tests give a more trustworthy definition of preference between MOSAs. In order to distinguish it from the one used in RQ1.1, we will call it *significant preference*, i.e., we will say that MOSA M is *significantly preferred* over MOSA M'.

[2] Note that some MOSAs are not applicable to some of the search problems, and so the formulations of (1), (2), and (3) should be slightly more complicated. We report the simplified versions here, but we use the correct versions in the experiments.

3.4.3 RQ2 - Studying MOSA's preference for QIs

For RQ2, we can define evaluation methods similar to RQ1, to achieve a MOSA's preference for a particular QI. Intuitively, a MOSA prefers the QI which also favors the MOSA. We formally capture this intuition as follows (*Step 5* in Fig. 1). Let M be a MOSA, Q_1 and Q_2 two QIs, and P a search problem. We say that a MOSA M_1 prefers a QI Q_1 w.r.t. QI Q_2, if Q_1 prefers M_1 more times than Q_2 does when comparing M_1 with other MOSAs for a given problem P. Formally:

$$pref_{MOSA}(M_1, Q_1, Q_2, P) = \begin{array}{l} |\{M_2 \in MOSAs \setminus \{M_1\} \mid pref_{QI}(Q_1, M_1, M_2, P)\}| > \\ |\{M_2 \in MOSAs \setminus \{M_1\} \mid pref_{QI}(Q_2, M_1, M_2, P)\}| \end{array} \tag{3}$$

Note that $pref_{MOSA}(M_1, Q_1, Q_2, P) = true$ implies $pref_{MOSA}(M_1, Q_2, Q_1, P) = false$. Moreover, if $pref_{MOSA}(M_1, Q_1, Q_2, P) = false$ and $pref_{MOSA}(M_1, Q_1, Q_2, P) = false$, it means that M_1 does not have any preference between Q_1 and Q_2.

RQ2.1 In order to answer RQ2.1, we compute the related *preference count* $PC_{MOSA}(M, Q)$ (*Step 6(a)* in Fig. 1) as follows:

$$PC_{MOSA}(M, Q) = \frac{\left| \bigcup_{P \in Problems} \{Q' \in QIs \setminus \{Q\} \mid pref_{MOSA}(M, Q, Q', P)\} \right|}{(|QIs| - 1) \times |Problems|} \tag{4}$$

where QIs is the set of QIs and *Problems* is the set of search problems. It shows the percentage of the preference of M for QI Q when compared to another QI in any problem. If a MOSA M prefers a QI Q for most of the different search problems when compared with the other QIs, it means that using the QI Q to evaluate the MOSA M can achieve assessment more favorable to it than using other QIs. If $PC_{MOSA}(M, Q)$ is higher, then it means that the MOSA M prefers the QI Q more.

RQ2.2 In order to answer RQ2.2, we proceed similarly to what done for RQ1.2. Firstly, we define the similar *preference count per problem* $PC_{MOSA}(M, Q, P)$ (*Step 6(b)* in Fig. 1) as follows:

$$PC_{MOSA}(M, Q, P) = \frac{|\{Q' \in QIs \setminus \{Q\} \mid pref_{MOSA}(M, Q, Q', P)\}|}{|QIs| - 1} \tag{5}$$

Then, in order to assess the *significant preference* (*Step 6(c)* in Fig. 1), we use the *Wilcoxon signed-rank* test and the *Vargha and Delaney* $\hat{A}_{12}$ statistics as described in Section 3.4.1, when we compare $PC_{MOSA}(M, Q, P)$ for the selected QIs in pairs across search problems for each $MOSAM$.

4 Results and Analyses

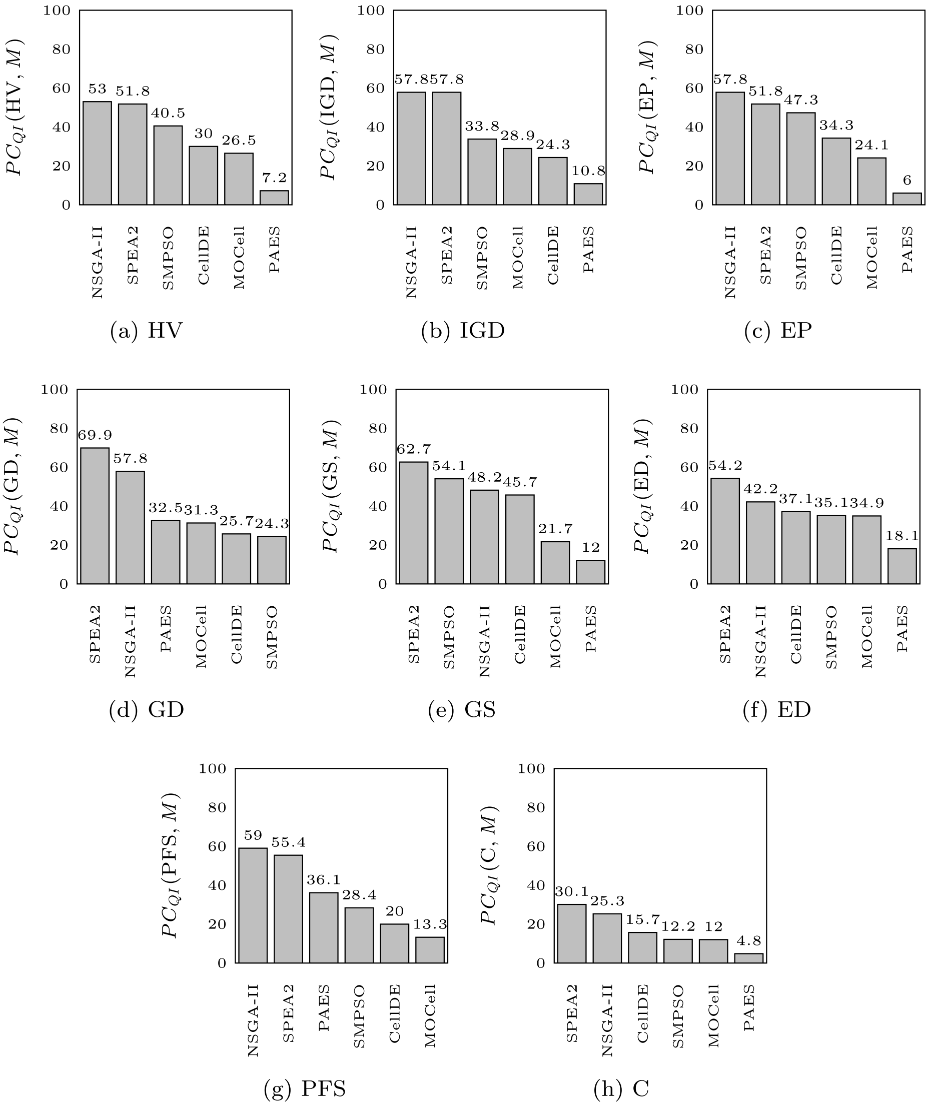
In this section, we present the results and analyses for our RQs. Section 4.1 presents the results of RQ1, whereas Section 4.2 presents the results of RQ2.

4.1 RQ1: Studying QI's preference for MOSAs

Recall from Section 3.4.2 that RQ1 studies the preference of a QI for a specific MOSA. Below, we present the results of RQ1.

4.1.1 RQ1.1: Studying QI's preferences for MOSAs with preference counts

RQ1.1 aims to study how many times a QI Q prefers a particular MOSA M across all the problems when comparing pairs of MOSAs (see (1) in Section 3.4.2). Given a QI Q, the preference count PC_{QI} for all the MOSAs is depicted in Fig. 2. We observe that some QIs prefer particular MOSAs. For instance, GD prefers SPEA2 with a preference count of 69.9%. This may be due to the reason that the elitist approach used by SPEA2 helps it in finding solutions that favor the *Convergence* quality aspect. As a result, SPEA2 is the most preferred by GD that completely covers this quality aspect. Note that the work reported by Goh and Tan (2009) also observed a similar link between the elitist-preserving approach



(a) HV

(b) IGD

(c) EP

(d) GD

(e) GS

(f) ED

(g) PFS

(h) C

Fig. 2 RQ1.1 – *Preference count* $PC_{QI}(Q, M)$ of each QI Q for each MOSA M

and convergence. Moreover, we can also observe that some QIs have low preferences for some MOSAs, e.g., EP with PAES (6.0%). This means that these MOSAs do not usually produce solutions that have the qualities preferred by these QIs.

In all the figures of Fig. 2, the preferences of each QI are sorted based on the values of the preference counts. We notice that SPEA2 is the most preferred MOSA, followed by NSGA-II. This is mainly because the elitist algorithm used by SPEA2 performs well in *Convergence* (Goh and Tan 2009) and the fine-grained fitness assignment strategy and the density estimation technique employed in SPEA2 together contribute to *Cardinality* (Zitzler et al. 2002). In addition, the enhanced archive truncation method of SPEA2 is used to strengthen the *Spread* and *Uniformity* of solution sets, as discussed in the work of Zitzler et al. (2002).

As reported by Zitzler et al. (2002), for many different problems, the performances of NSGA-II and SPEA2 are similar. From Table 1, we can see that HV, IGD, and EP cover the four quality aspects, GD and ED only fully or partially represent the *Convergence* aspect. For NSGA-II, its elitist-preserving approach has a good influence on *Convergence* as observed by Goh and Tan (2009), its special parameter-less niching operator helps ensure the diversity of the Pareto fronts, which improves its performance in the *Spread* and *Uniformity* quality aspects, as reported in the work of Deb et al. (2002). These specific mechanisms working together may cause NSGA-II to be the most or highly preferred by HV, IGD, EP, GD, and ED.

In addition, we observe that PAES is the least preferred by six QIs: HV, IGD, EP, GS, ED, and C. The plausible explanation is that the higher implicit elitism intensity employed by PAES enables it to converge faster at the beginning of the search, but, as a result, it can easily get stuck when a good solution is not found. Such observation is also reported by Goh and Tan (2009) and Zitzler et al. (2002).

For GS, which partially covers *Spread* and fully *Uniformity*, SPEA2 performed the best, followed by SMPSO. SPEA2, thanks to its enhanced archive truncation method which may positively influences *Spread* and *Uniformity*, as discussed by Zitzler et al. (2002). Moreover, SMPSO (containing the special velocity constriction mechanism) also shows the strongest influence on the *Spread* and *Uniformity* aspects, as observed by Nebro et al. (2009) and Durillo et al. (2009a).

To better answer RQ1.1, Table 5 summarizes the results in terms of which MOSAs are the most preferred across all the QIs. In the table, the rank corresponds to the percentage order of a MOSA preferred by a QI in Fig. 2. The higher the rank, the higher the percentage order. After the MOSA name, we report in parentheses the QIs for which the MOSA occupies that rank.

Table 5 RQ1.1 – Overall ranking of MOSAs preferred by QIs (for each rank position, it lists the MOSAs that have that ranking for some QIs (reported in parentheses))

Rank	Instances for each MOSA
1	SPEA2 (GD, GS, ED, C), NSGA-II (HV, EP, PFS), NSGA-II/SPEA2 (IGD)
2	SPEA2 (HV, EP, PFS), NSGA-II (GD, ED, C), SMPSO (GS)
3	SMPSO (HV, IGD, EP), PAES (GD, PFS), CellDE (ED, C), NSGA-II (GS)
4	CellDE (HV, EP, GS), SMPSO (ED, PFS, C), MOCell (IGD, GD)
5	MOCell (HV, EP, GS, ED, C), CellDE (IGD, GD, PFS)
6	PAES (HV, IGD, EP, GS, ED, C), SMPSO (GD), MOCell (PFS)

Overall, each QI has its own specific most-preferred MOSA. For instance, HV, EP, and PFS all prefer NSGA-II the most; IGD prefers NSGA-II and SPEA2 the most; and GD, GS, ED and C prefer SPEA2 the most. For all the QIs and all the search problems, the overall preference descending order of all the MOSAs is as follows: SPEA2, NSGA-II, SMPSO, CellDE, MOCell, and PAES.

4.1.2 RQ1.2: Studying the statistical significance of QI's preferences for MOSAs

RQ1.2 aims to study the MOSA preferences of all the selected QIs for each selected search problem. Table 13 in Appendix A.1 reports all detailed data. We here summarize the key findings. Overall, we observe that a given QI has different MOSA preferences for the different problems. For the same problem, different QIs prefer different MOSAs. Although there are problems (such as Integration and Test Order) for which all QIs most prefer the same MOSA (such as NSGA-II), the preference order for the other MOSAs is still different.

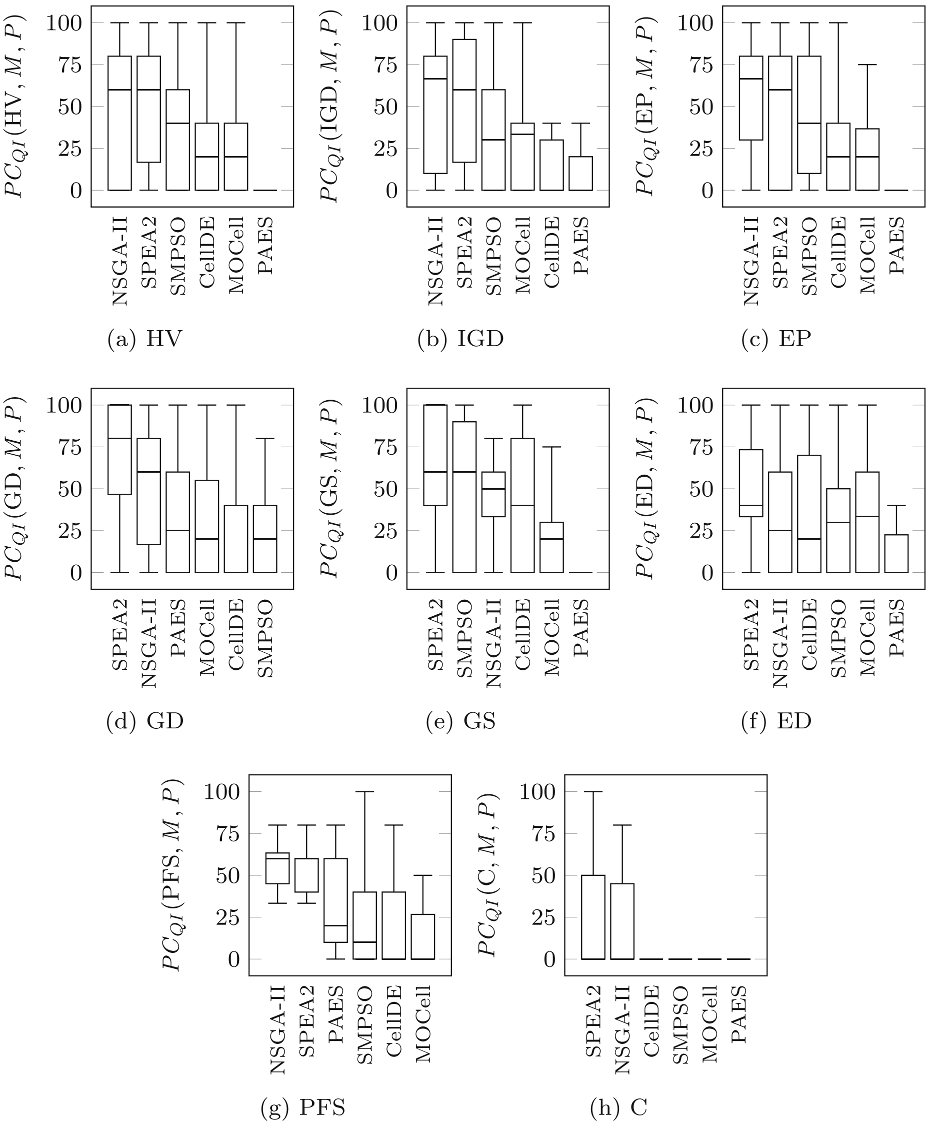
Based on the raw results in Tables 13, and 6 reports, for each QI Q and each MOSA M, how many times (i.e., for how many problems), Q prefers M the most. Since the results of each QI are not concentrated on a single MOSA, it is clear that the most preferred MOSA also depends on the search problem.

Figure 3 further reinforces the above conclusion. It reports, for each quality indicator Q and each MOSA M, the distribution of the metric *preference count per problem* $PC_{QI}(Q, M, P)$ (see (2)) across the search problems P; MOSAs are sorted as in Fig. 2. Again, we observe that Q may prefer a MOSA M on some problems, but not on others. The influence of the problem characteristics on the results of QIs has been also discovered in our previous work (Ali et al. 2020), in which we discovered that the agreement between pairs of QIs, i.e., whether they prefer the same MOSA, sometimes depends on problems solved by the MOSA. Note that we can not perform an analysis on the basis of different problem characteristics (e.g., the number of objectives), as this would require many more problems for each given characteristic. For the number of objectives, for example, we have four problems with two objectives, three problems with three objectives, and 11 problems with four objectives. This is insufficient to draw any solid conclusion about the influence of the number of objectives on QIs' preferences on MOSAs.

We further compared the results using statistical tests (*Step 4(c)* in Fig. 1), using the *Wilcoxon signed-rank* test and the $\hat{A}_{12}$ statistics as described in Section 3.4. Following the

Table 6 RQ1.2 – Number of search problems in which a QI Q prefers a MOSA M the most

QI	MOSA					
	CellDE	MOCell	NSGA-II	SMPSO	SPEA2	PAES
HV	6	6	9	7	10	4
IGD	5	5	10	5	11	4
EP	6	4	11	7	9	4
GD	3	4	6	2	12	4
GS	6	2	4	6	10	2
ED	6	10	6	4	12	3
PFS	1	3	16	5	15	8
C	6	9	10	5	12	7



(a) HV

(b) IGD

(c) EP

(d) GD

(e) GS

(f) ED

(g) PFS

(h) C

Fig. 3 RQ1.2 – Distribution of the *preference count per problem* $PC_{QI}(Q, M, P)$ of each QI Q for each MOSA M over all the search problems P

guidelines proposed in the report of Kitchenham et al. (2017), we divide the effect size magnitude of $\hat{A}_{12}$ into four levels: negligible (> 0.5 and < 0.556), small (≥ 0.556 and < 0.638), medium (≥ 0.638 and < 0.714), and large (≥ 0.714 and ≤ 1.0). Table 7 reports the overall results of the statistical tests. Namely, each cell in the table reports the number of times that a MOSA (such as NSGA-II) is significantly preferred over the other MOSAs by a given QI, and the $\hat{A}_{12}$ is at least *medium*, i.e., greater than or equal to 0.638. The MOSA(s) that is(are) most often preferred (over other MOSAs) by a given QI is(are) highlighted in gray in the table. We notice that HV, IGD, EP, PFS and C prefer NSGA-II and SPEA2 the most.

Table 7 RQ1.2 – Overall preferences of each QI on each MOSA (number of times that a QI significantly prefers a MOSA over the other MOSAs (with $\hat{A}_{12}$ at least in the *medium* category). Gray cells indicate the most often preferred MOSAs)

QI	MOSA					
	CellDE	MOCell	NSGA-II	SMPSO	SPEA2	PAES
HV	0	1	2	1	2	0
IGD	0	1	4	1	4	0
EP	1	1	2	1	2	0
GD	0	0	2	0	4	0
GS	1	0	2	2	2	0
ED	0	1	0	0	1	0
PFS	0	0	4	0	4	1
C	0	0	1	0	1	0

GD prefers SPEA2 the most, and ED prefers MOCell and SPEA2 the most. GS, instead, prefers NSGA-II, SMPSO, and SPEA2 the most. Overall, we can observe that SPEA2 is preferred by most of the QIs, followed by NSGA-II, SMPSO, and MOCell. CellDE and PAES, instead, are never the most often preferred.

Figure 4 better visualizes the significant preference relations we described above. For each QI, the figure shows which MOSAs are significantly preferred over others. An arrow from MOSA M_1 to MOSA M_2 means that M_1 is significantly preferred over M_2. We observe that some MOSAs are constantly significantly preferred over some others. For instance, NSGA-II and SPEA2 are preferred over PAES in most of the cases. The most preferred MOSAs (i.e., those with the highest numbers in Table 7) are usually preferred over the same other MOSAs. Moreover, there are some MOSAs that, although are worse than some MOSAs, are better than some others, for example, MOCell in HV, IGD, and EP, and PAES in PFS.

The preferences of QIs for MOSAs vary across the search problems suggesting that problem characteristics influence the preferences of QIs for MOSAs. Nonetheless, SPEA2 is the most preferred, either by itself or in tie with NSGA-II.

4.2 RQ2: Studying MOSA's preference for QIs

Recall from Section 3.4.3 that RQ2 studies the preference of a MOSA for a specific QI, via two sub-questions: RQ2.1 and RQ2.2. We report the results of these two sub-RQs and analyze the results according to the characteristics and quality aspects of the selected QIs (see Table 1) in the following sub-sections.

4.2.1 RQ2.1: Studying MOSA's preference for QIs with preference counts

Figure 5 presents the results of *preference count* $PC_{MOSA}(M, Q)$ (see (4) in Section 3.4.3) sorted in decreasing order. The $PC_{MOSA}(M, Q)$ indicates the preference of a MOSA M for a QI Q across all the search problems. From the results, we can observe that MOSAs have their own specific preferences for QIs. Considering the quality indicator C, we observe that almost all the MOSAs have the lowest preference percentages for it, except for MOCell in which C is the second last. This means that no MOSA is consistently preferred by C over

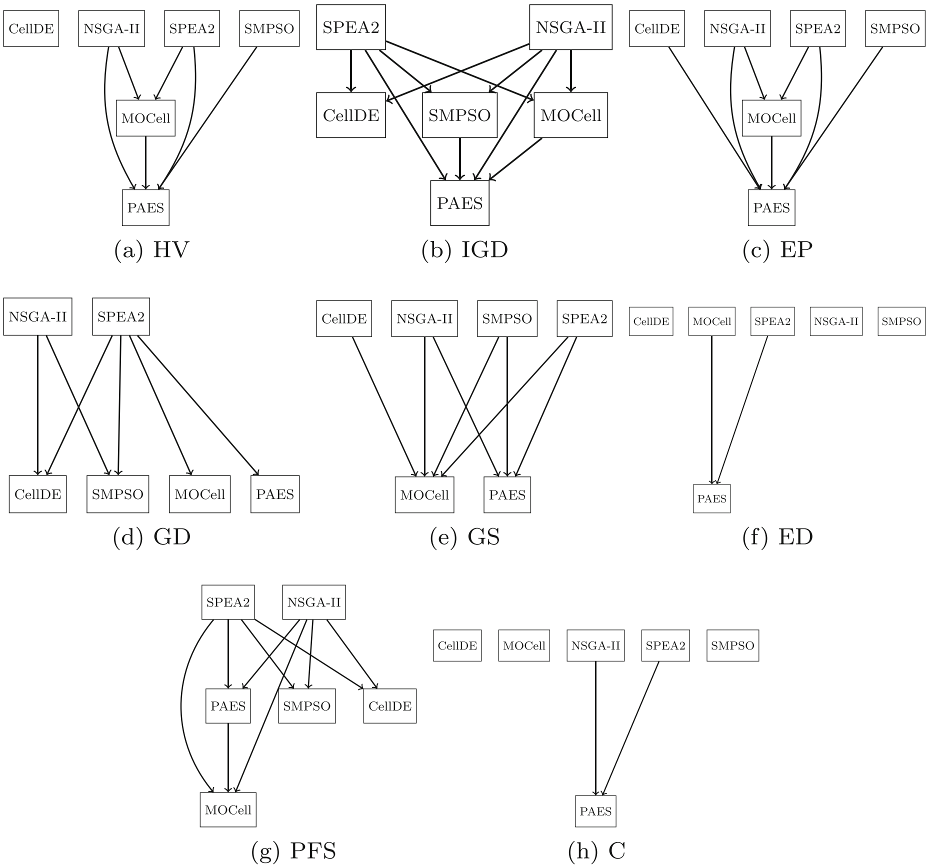
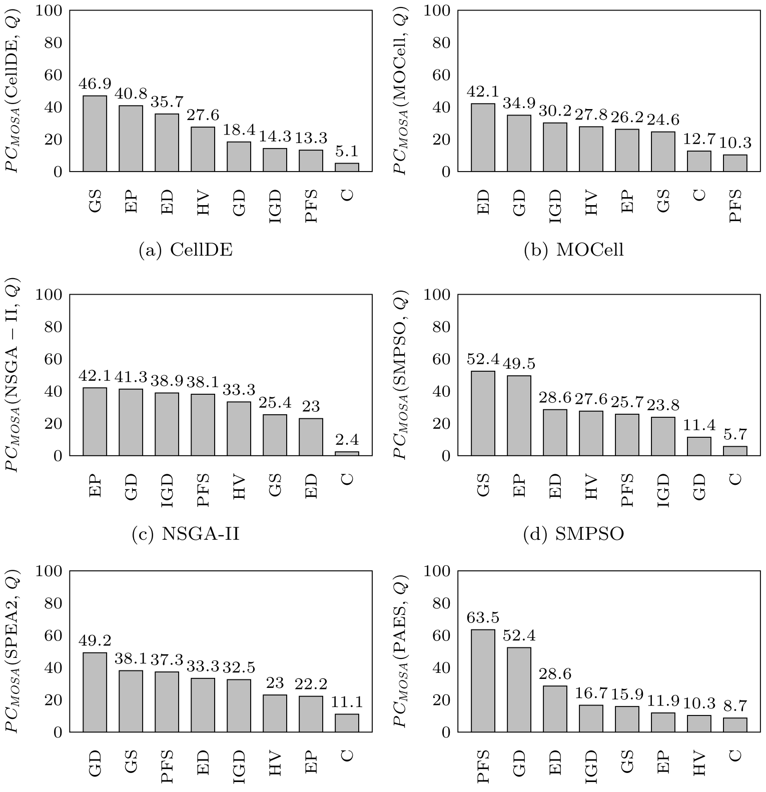


Fig. 4 RQ1.2 – Significant QI's preference relations between MOSA pairs (an arrow from MOSA M_1 to MOSA M_2 means that M_1 is significantly preferred over M_2)

the other MOSAs. Hence, in this subsection, we temporarily exclude the influence of C and analyze the other QIs.

From Fig. 5, we observe that CellDE prefers GS the most (46.9%), followed by EP (40.8%). It has lower preference for the other QIs, e.g., PFS (13.3%) and IGD (14.3%). From Table 1, we observe that GS fully covers the *Uniformity* aspect and partially covers the *Spread* aspect. EP also fully cover *Spread* aspect and partially cover *Uniformity*. Whereas, PFS does not cover these quality aspects at all. This tells us that CellDE produces solutions that have high *Spread* and *Uniformity*, because it employs the search engine of MOCell, which produces diverse solutions as mentioned in the work of Durillo et al. (2008) and Nebro et al. (2009). Regarding CellDE's low preference towards IGD, considering that IGD partially or fully covers all of the four quality aspects, our results suggest that the performance of the solutions generated by CellDE in *Convergence* and *Cardinality* may have a greater impact on the evaluation results of IGD, resulting in IGD not being preferred by CellDE.

For MOCell, we can see that it prefers ED (42.1%) the most, followed by GD (34.9%), IGD (30.2%), HV (27.8%), and EP (26.2%). Instead, MOCell prefers PFS (10.3%) much



(a) CellDE (b) MOCell

(c) NSGA-II (d) SMPSO

(e) SPEA2 (f) PAES

Fig. 5 RQ2.1 – *Preference count* $PC_{MOSA}(M, Q)$ of each MOSA M for each QI Q

less. This might be because ED, GD, IGD, HV and EP all cover the *Convergence* aspect, and MOCell produces solutions favoring *Convergence* due to the elitist algorithm it employs (Goh and Tan 2009).

NSGA-II most prefers EP, GD, IGD, PFS, and HV, with similar percentage values ranging from 33.3% to 41.3%. This is because NSGA-II performs well in generating solutions covering all the four quality aspects due to its employed elitist-preserving (Goh and Tan 2009) and parameterless diversity-preservation mechanisms (Deb et al. 2002). Note that EP, IGD, and HV cover all the four quality aspects; GD and PFS fully cover *Convergence* and *Cardinality*, respectively (Table 1).

SMPSO prefers GS the most (52.4%), and (without considering C) prefers GD the least (11.4%), meaning that SMPSO could create well-distributed solutions but also create solutions far from the reference Pareto front, because GS favors well-distributed solutions, while GD promotes the closeness to the reference Pareto front. Therefore, for SMPSO, it may be easy to create well-distributed solutions, i.e., well-performing solutions in *Spread* and *Uniformity* aspects, which is consistent with the observations reported in the works of Nebro et al. (2009) and Durillo et al. (2009a).

SPEA2 prefers GD (49.2%) the most, followed by GS (38.1%), PFS (37.3%), ED (33.3%), and IGD (32.5%). The main is that SPEA2 applies an enhanced archive truncation method to improve the *Spread* and *Uniformity* of solution sets (Zitzler et al. 2002), uses a fine-grained fitness assignment strategy and a density estimation technique to enhance the solution dominance (Zitzler et al. 2002), and adopts an elitist algorithm to strengthen the solution set *Convergence* (Goh and Tan 2009). Therefore, SPEA2 performs well in the four quality aspects.

PFS counts the number of non-dominated solutions in the computed Pareto front. PAES prefers PFS (63.5%) the best, meaning that, in its obtained solution sets, the number of non-dominated solutions is large. PAES, as a result of its premature convergence, may produce duplicated non-dominated solutions (Goh and Tan 2009; Zitzler et al. 2002). Note that repeated solutions do not dominate each other (Li et al. 2020), and so they contribute to the increase of the PFS value. This might be the reason why PAES prefers PFS. In short, PAES tends to generate solutions particularly relevant for the *Cardinality* aspect, which is fully represented by PFS.

These results can be further confirmed in Table 8. According to the results reported in Fig. 5, we set ranks from 1 to 8 to represent the QI percentage order, with 1 corresponding to the highest percentage and 8 corresponding to the lowest percentage. We fill in the QIs in the corresponding order, and report in parentheses which MOSA(s) the QI is preferred by in this rank. From the table, we can observe that for any QI, its ranking distribution is relatively uniform. For instance, GD ranks the first, second, fifth, and even seventh, across the MOSAs, which indicates that GD is unlikely to always fall in the same or simila r ranks.

No QI is consistently preferred by all the MOSAs, i.e., both CellDE and SMPSO prefer GS the most; MOCell prefers ED the most; NSGA-II prefers EP the most; SPEA2 prefers GD the most; and PAES prefers PFS the most.

4.2.2 RQ2.2: Studying statistical significance of MOSA's preference for QIs

In this RQ, we investigate the preference of MOSAs for QIs, by considering the different search problems. The detailed results are reported in Table 14 (see Appendix A.2). In summary, we observed that each MOSA prefers different QIs the most in different search problems. For example, MOCell prefers HV, IGD, and EP equally the most in the Requirements Allocation for Inspection problem, while GD is the most preferred by MOCell in the Testing Resource Allocation problem. Also, we noticed that for the same search problem, different MOSAs can most prefer different QIs, e.g., for the Test Case Selection problem,

Table 8 RQ2.1 – Overall ranking of QIs preferred by MOSAs (for each rank position, it reports the QIs that have that ranking for some MOSAs (reported in parentheses))

Rank	Instances for each QI
1	GS (CellDE, SMPSO), ED (MOCell), EP (NSGA-II), GD (SPEA2), PFS (PAES)
2	GD (MOCell, NSGA-II, PAES), EP (CellDE, SMPSO), GS (SPEA2)
3	ED (CellDE, SMPSO, PAES), IGD (MOCell, NSGA-II), PFS (SPEA2)
4	HV (CellDE, MOCell, SMPSO), PFS (NSGA-II), ED (SPEA2), IGD (PAES)
5	GD (CellDE), EP (MOCell), HV (NSGA-II), PFS (SMPSO), IGD (SPEA2), GS (PAES)
6	IGD (CellDE, SMPSO), GS (MOCell, NSGA-II), HV (SPEA2), EP (PAES)
7	PFS (CellDE), C (MOCell), ED (NSGA-II), GD (SMPSO), EP (SPEA2), HV (PAES)
8	C (CellDE, NSGA-II, SMPSO, SPEA2, PAES), PFS (MOCell)

CellDE most prefers GS, but SMPSO most prefers EP. We summarize the key results of Tables 14 in 9. For each MOSA M and each QI Q, it reports for how many problems M prefers Q the most. We observe that a MOSA does not always prefer the same QI the most, but tends to prefer different QIs the most for different problems.

We further analyze the distribution of the previous results. Figure 6 presents the distribution of *preference count per problem* $PC_{MOSA}(M, Q, P)$ over the search problems P (see (5)). Results are sorted as in Fig. 5. From the figure, we can observe that, for a specific MOSA (e.g., CellDE), the variances of QIs having high preference counts as reported in Fig. 5 (e.g., GS and EP for CellDE) are mostly high. It implies that the different search problems may affect MOSAs' preferences over QIs. More specifically, for one problem, a MOSA may prefer a QI, while on another problem this MOSA may not prefer the same QI. Note that, for QIs having a general low preference count (e.g., C), the variance of metric $PC_{MOSA}(M, Q, P)$ across all the problems is low.

Table 10 presents results of the statistical tests. The table reports, for each MOSA M and each QI Q, how many times M prefers Q rather than another QI; the preference is computed by comparing the distribution of the preference count $PC_{MOSA}(M, Q, P)$ across the search problems, using the *Wilcoxon signed-rank* test and $\hat{A}_{12}$ statistics, as described in Section 3.4.1. Note that, as done for RQ1.2, we only consider cases that the $\hat{A}_{12}$ value is at least in the *medium* category. For each MOSA, we highlight in gray the QI(s) that has(have) the highest overall preferences. We observe that CellDE prefers GS the most; MOCell prefers GD and ED the most; NSGA-II prefers all the QIs except C; SMPSO prefers GS the most; SPEA2 prefers GD the most; and PAES prefers PFS the most. We can further observe that no QI is consistently preferred by all the MOSAs.

Figure 7 better visualizes the significant preference relations between QI pairs. A link with a pointing arrow shows a preference relation between two QIs. For instance, for CellDE (see Fig. 7a), EP points to C implying that EP is significantly preferred by CellDE over C. Similarly, HV, ED and GS are all preferred over C, without significant differences observed among themselves; GS is preferred over IGD, GD, and PFS. For MOCell, IGD, ED, and GD are all preferred over PFS, ED and GD are both preferred over C; there are instead no significant preferences for HV, IGD, ED, GD, EP and GS when comparing among themselves and with the other QIs. For NSGA-II, HV, IGD, EP, GD, GS, ED and PFS are all preferred over C, but there is no significant preference among them. For SMPSO, we observe a three-hierarchical-level significant preference relation. For instance, GS is preferred over IGD, which is preferred over C. For SPEA2, we observe that GD is preferred over HV, EP

Table 9 RQ2.2 – Number of search problems that a MOSA M prefers a QI Q the most

MOSA	QI							
	HV	IGD	EP	GD	GS	ED	PFS	C
CellDE	6	4	7	4	9	7	3	3
MOCell	9	11	9	12	7	11	3	4
NSGA-II	5	8	7	9	3	3	6	0
SMPSO	5	4	8	2	9	4	5	2
SPEA2	4	8	5	13	9	6	4	4
PAES	1	2	0	7	3	4	9	2

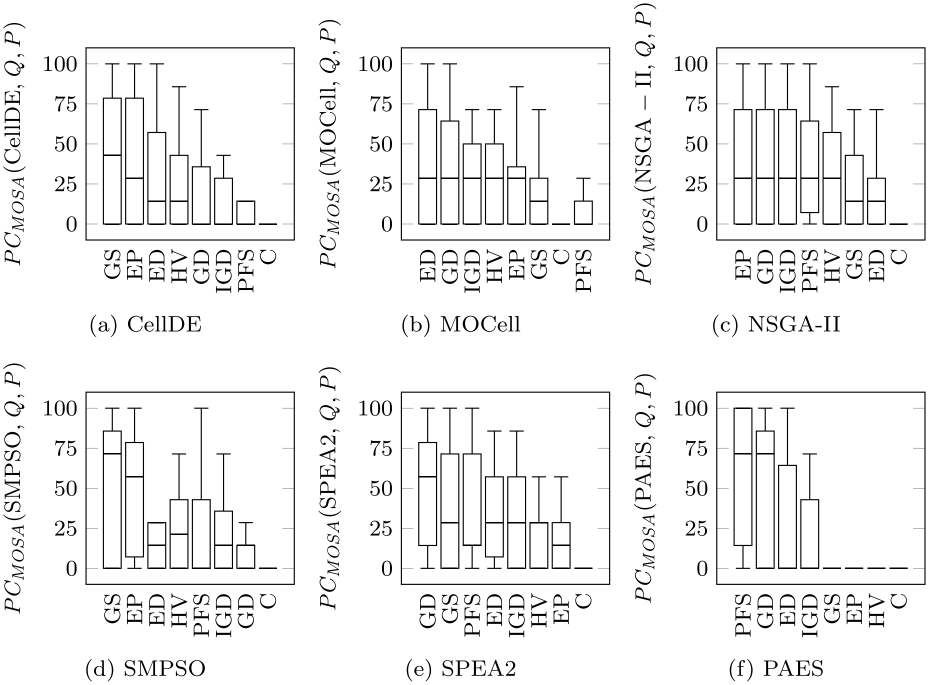


Fig. 6 RQ2.2 – Distribution of the *preference count per problem* $PC_{MOSA}(M, Q, P)$ of each MOSA M for each QI Q over all the search problems P

and C; IGD, GS, ED, and PFS are also preferred over C; and no significant preferences are observed for them. Regarding PAES, we also observe a three-hierarchical-level significant preference relation; for example, PFS is preferred over ED, which is preferred over C.

Different search problems may have influence on MOSAs' preferences for QIs. Each selected MOSA has its mostly preferred QI, i.e., GS for CellDE; GD and ED for MOCell; all the selected QIs except C for NSGA-II; GS for SMPSO; GD for SPEA2; and PFS for PAES. No QI is preferred the most by all the selected MOSAs.

4.3 Analyses based on Quality Aspects of QIs

In this section, we present additional analyses of the preferences of the quality aspects of the QIs on the MOSAs. The analysis in Sections 4.1 and 4.2 only consider whether the preference of a QI for a particular MOSA (or of a MOSA for a QI) may be due to a particular quality aspect, but do not check the influence of a quality aspect across the different QIs. In this section, we analyze whether QIs covering the same quality aspect(s) have the same or similar preferences for MOSAs. More specifically, we want to check whether they have similar rankings in Fig. 2.

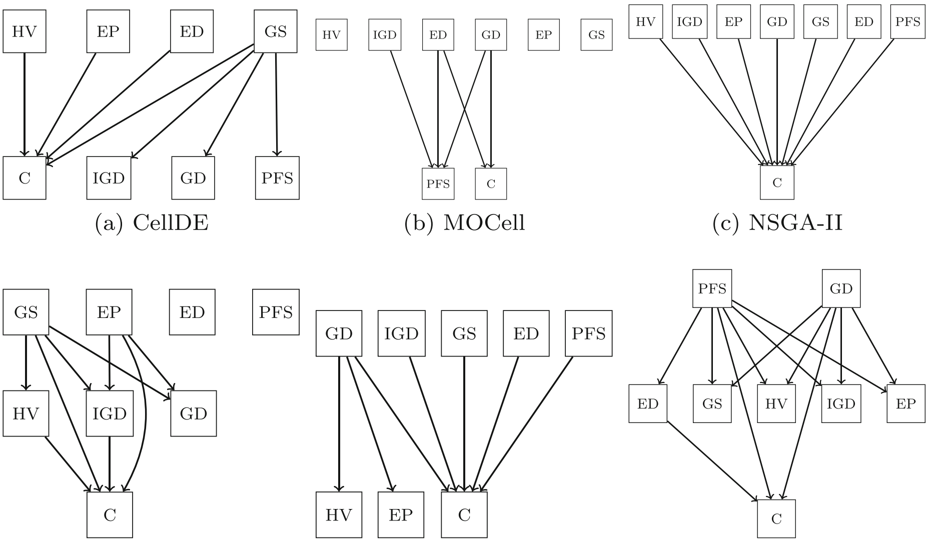
Combining Tables 1, 7, and 11 summarizes our results. For each MOSA M and each aspect, it reports how many times M is preferred (as reported in

Table 10 RQ2.2 – Overall preferences of each MOSA on each QI (number of times that a MOSA significantly prefers a QI over the other QIs (with $\hat{A}_{12}$ at least in the *medium* category). Gray cells indicate the most often preferred QIs)

MOSA	QI							
	HV	IGD	EP	GD	GS	ED	PFS	C
CellDE	1	0	1	0	4	1	0	0
MOCell	0	1	0	2	0	2	0	0
NSGA-II	1	1	1	1	1	1	1	0
SMPSO	1	1	3	0	4	1	0	0
SPEA2	0	1	0	3	1	1	1	0
PAES	0	0	0	5	0	1	6	0

Table 7) by a QI belonging to the considered aspect. For example, for *Convergence*, SPEA2 and NSGA-II achieves the highest values, i.e., 14 and 11, respectively. In general, we can see that NSGA-II and SPEA2 are the most preferred ones for all the quality aspects.

HV completely covers *Convergence*, *Spread*, and *Cardinality* aspects and partially covers *Uniformity* (Table 1). IGD and EP fully represent *Convergence* and *Spread* aspects,



(a) CellDE (b) MOCell (c) NSGA-II

(d) SMPSO (e) SPEA2 (f) PAES

Fig. 7 RQ2.2 – Significant MOSA's preference relations between QI pairs (an arrow from QI Q_1 to QI Q_2 means that Q_1 is significantly preferred over Q_2)

and partially represent *Uniformity* and *Cardinality*. When looking at our experiment results reported in Fig. 2, for HV, IGD and EP, we observe almost the same order of the MOSAs (except for IGD in which MOCell and CellDE are swapped).

ED partially and exclusively covers the *Convergence* quality aspect and GD fully and exclusively covers this aspect (Table 1), implying that these two QIs are quite similar. However, after analyzing the results of ED and GD (Fig. 2), we can observe that some MOSAs are ranked at different positions. For example, for ED, PAES is ranked at the last position, which is however at the third position for GD.

GS fully covers *Uniformity*, partially covers *Spread*, and does not cover *Convergence* and *Cardinality*, as shown in Table 1. GS has SMPSO ranked at the second place with a higher preference count among the most MOSAs: 54.1%, as shown in Fig. 2e. Similarly, we can also observe, from Fig. 5d, SMPSO prefers GS the most (i.e., 52.4%). We underline that GS is the only QI that fully covers *Uniformity*, and that SMPSO turns out to be the best MOSA only for GS (for the other QIs, it is at most the third preferred one). This shows that solutions provided by SMPSO have a good uniformity, but this is appreciated only by GS.

From Table 1, we can also see that PFS only covers *Cardinality*. When looking at the results reported in Fig. 2g, PAES is ranked at the third place, which is however not the case for almost all the other QIs in which it this the least preferred (except for GD where it is also third). This result can be also easily observed in Fig. 5f, which clearly tells that PAES prefers PFS the most.

Finally, from Table 1, we can observe that C partially represents the *Convergence* and *Cardinality* aspects. In Fig. 2h, we can see that although the percentage values of C are all low, C still most prefers SPEA2 (30.1%) and NSGA-II (25.3%). Moreover, from Fig. 5, we can observe that all the MOSAs have low preferences for C (always the least preferred, except for MOCell in which it is the second last). These results seem to show that all the MOSAs achieve similar solutions in terms of *Convergence* and *Cardinality*.

Based on the above observations, we can conclude that:

- QIs covering a comprehensive list of quality aspects (e.g., HV, IGD, and EP) tend to exhibit the same order of preferences on MOSAs;
- For QIs covering only one or two quality aspects (e.g., GD, GS, ED, PFS, and C), though some of them cover the exact same quality aspect, they do not necessarily have the same preferences for MOSAs. This observation is consistent with what has been reported by Ravber et al. (2017), who concluded that QIs with the same quality aspect(s) do not necessarily yield the same rankings of MOSAs.

Table 11 Overall preferences of quality aspects for MOSAs (how many times a MOSA *M* is preferred (see Table 7) by a QI belonging to a given aspect)

Quality Aspect	MOSA					
	CellDE	MOCell	NSGA-II	SMPSO	SPEA2	PAES
Convergence	1	4	11	3	14	0
Spread	2	3	10	5	10	0
Uniformity	2	3	10	5	10	0
Cardinality	1	3	13	3	13	1

5 Overall Discussion and Application Contexts

In this section, we first present an overall discussion both for the results based on individual QIs and the quality aspects they cover. Moreover, we provide suggestions to users for selecting a MOSA that will likely produce solutions preferred by a given QI or a given quality aspect. To this end, we have integrated the key conclusions of Section 4 (see Table 12). At the last, we discuss the relevance of the proposed study, by presenting application contexts in which users are really aware of the desired quality aspects, and so our guideline can be used.

Based on the results of RQ1.2 and RQ2.2, we plot Fig. 8 to facilitate the discussion. An arrow from a QI Q to a MOSA M means that Q has the highest number of statistical preferences for M in Table 7. Similarly, an arrow from a MOSA M to a QI Q means that M has the highest number of statistical preferences for Q in Table 10. A bidirectional arrow between a QI Q and a MOSA M means that there is a *mutual preference* between Q and M. In summary, from this figure, we can observe that NSGA-II has mutual preferences with five out of the eight QIs; SMPSO and GS mutually prefer each other; and SPEA2 mutually prefers GD. In certain cases, there might be more than one MOSAs with mutual preferences with a given QI. For example, GS has mutual preferences with two MOSAs, i.e., NSGA-II and SMPSO.

Considering the identified mutual preferences and the results reported in Tables 7, 11, and 12, we propose a guideline for SBSE users to follow when selecting a MOSA based on a given QI Q:

- When there are no ties, select the MOSA that has the largest number for the row of Q in Table 7, i.e., the third key conclusion of RQ1.2 in Table 12. Note that some results provide better assurances than others. For example, as shown in Table 7, for GD, SPEA2 has a value of 4 out of 5 (i.e., 80% of the times SPEA2 was significantly preferred over the other MOSAs), whereas, for EP, SPEA2 has a value of 2 out of 5 (i.e., 40%). Thus, in cases that a user wants solutions that are represented by GD or EP, our results

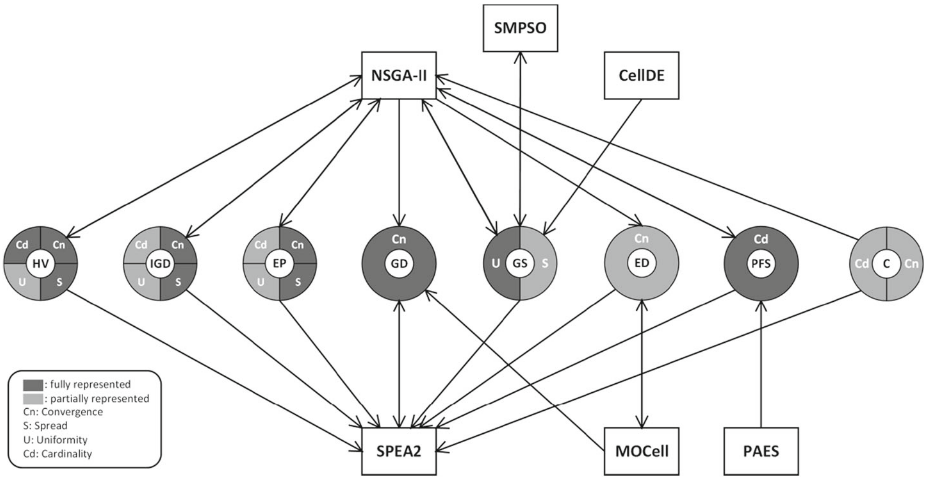


Fig. 8 Statistically significant mutual preferences between QIs and MOSAs (derived for the results of RQ1.2 and RQ2.2)

Table 12 Summary of the key observations

RQ	Analysis metrics	Key Results
RQ1.1	$PC_{QI}(Q, M)$	(1) IGD, GD, GS, ED, and C → SPEA2; HV, IGD, EP, and PFS → NSGA-II. (2) For all QIs and all search problems, the overall preference order of all MOSAs is: SPEA2 > NSGA-II > SMPSO > CellDE > MOCell > PAES.
RQ1.2	$PC_{QI}(Q, M, P)$ and *Wilcoxon signed-rank* test and *Vargha and Delaney* $\hat{A}_{12}$ effect size	(1) Different search problems would affect the preference of QIs to MOSAs. (2) For overall selected search problems, SPEA2 is the most preferred, followed by NSGA-II, SMPSO, and MOCell, CellDE and PAES have no significant difference. (3) QI has its own most preference MOSA(s): HV → NSGA-II and SPEA2; IGD → NSGA-II and SPEA2; EP → NSGA-II and SPEA2; GD → SPEA2; GS → NSGA-II, SMPSO, and SPEA2; ED → MOCell and SPEA2; PFS → NSGA-II and SPEA2; and C → NSGA-II and SPEA2.
RQ2.1	$PC_{MOSA}(M, Q)$	(1) CellDE → GS; MOCell → ED; NSGA-II → EP; SMPSO → GS; SPEA2 → GD; and PAES → PFS. (2) For all MOSAs of all search problems, no QI is the best preferred by most MOSAs.
RQ2.2	$PC_{MOSA}(M, Q_1, P)$ and *Wilcoxon signed-rank* test and *Vargha and Delaney* $\hat{A}_{12}$ effect size	(1) Different search problems may have influence on MOSA's preference for QI. (2) Each selected MOSA has its most preference QI: CellDE → GS; MOCell → GD and ED; NSGA-II → all selected QIs except C; SMPSO → GS; SPEA2 → GD; and PAES → PFS. (3) No QI is the most preferred by all the selected MOSAs.
QA		(1) NSGA-II and SPEA2 are the most preferred ones for *Spread*, *Uniformity*, and *Cardinality*; SPEA2 is the most preferred for *Convergence*. (2) QIs covering the same quality aspect(s) do not necessarily have the same preference for MOSAs.

">": more prefer than; "A → B": A prefers B the most; "QA": the analyses based on quality aspects of QIs (see in Section 4.3)

suggest to use SPEA2. However, in the former case, the user will be more certain when following the suggestion, because this is based on a stronger result; or

- When there is a tie (e.g., for HV, there is a tie between NSGA-II and SPEA2 as shown in Table 7), we have the following options:
  - selecting any MOSA;
  - selecting a MOSA that has a mutual preference (if exists) with the chosen QI. For example, for the tie between NSGA-II and SPEA2 for HV, considering that we know HV has a mutual preference with NSGA-II (see Fig. 8), we recommend NSGA-II. However, in certain cases, there can even be ties in mutual preferences. For example, for GS there is a tie between NSGA-II and SMPSO (Fig. 8). In this case, we recommend selecting any of these MOSAs;
  - selecting a MOSA based on the quality aspect table (i.e., Table 11) by checking which quality aspect(s) is(are) represented by the selected QI (i.e., Table 1). For example, as for ED, from Fig. 8, we see that there is a tie between MOCell and

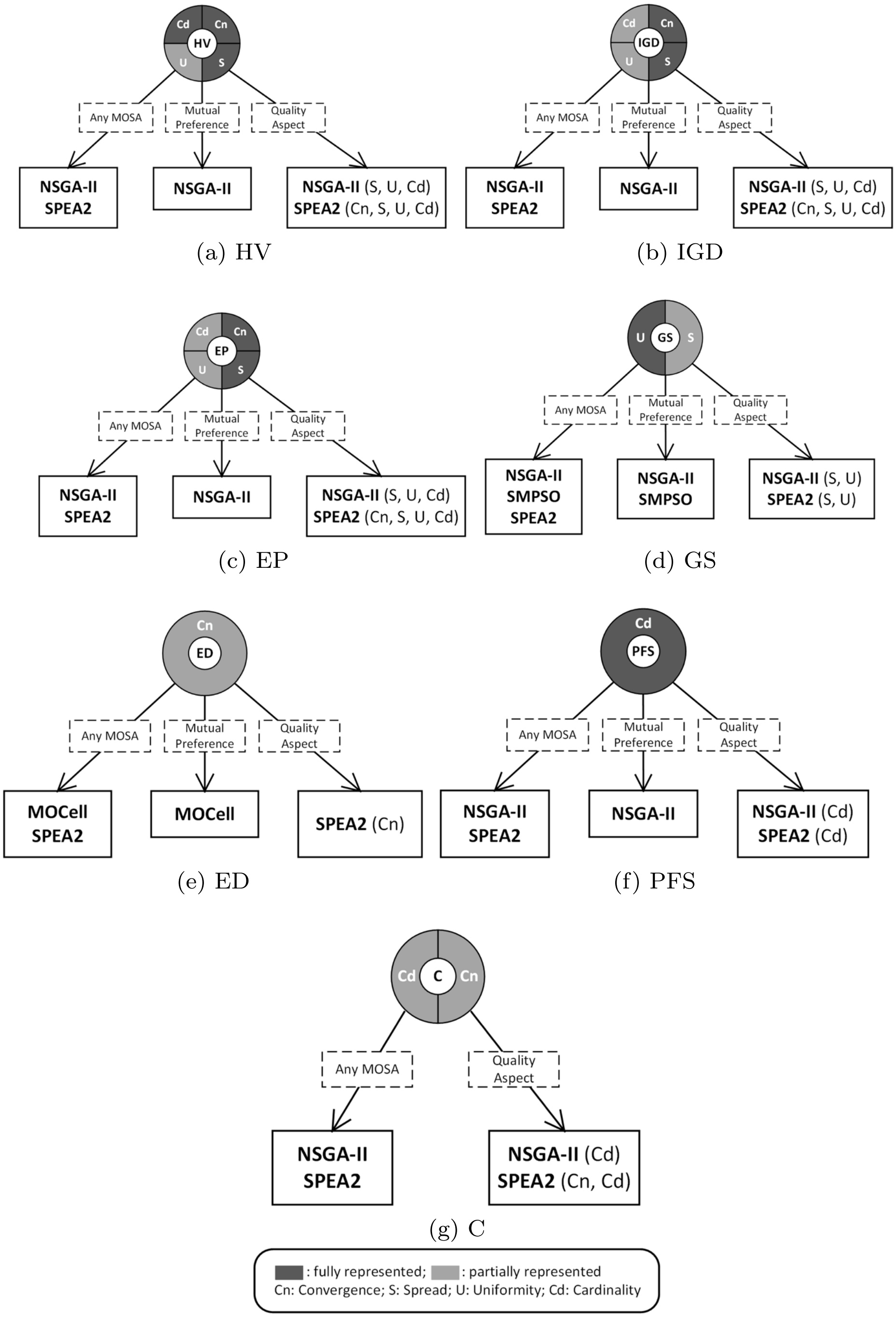
SPEA2, and ED partly represents solutions with the *Convergence* quality aspect. Then, based on Table 11, for *Convergence*, SPEA2 is more preferred than MOCell.

The user may also specify a desired quality aspect instead of a specific QI. In this case, we check Table 11 or simply refer to the quality aspect results summarized in Table 12, from which, one can easily see that NSGA-II is the most preferred for the *Spread*, *Uniformity*, and *Cardinality* aspects, and SPEA2 is the most preferred for all the four quality aspects. However, once the results are updated based on additional experimental results of other search problems, the preferences may change and we may have ties between two or more MOSAs. If such a case arises, we suggest selecting any of the preferred MOSAs of the tie or taking into account other aspects of MOSAs (e.g., their time performance). Moreover, more complicated guidelines could be provided with the availability of more data, by selecting the MOSA scoring the highest in the majority of the quality aspects, for instance.

In summary, given a QI, our recommendations, in terms of decision trees, for selecting a MOSA are shown in Fig. 9; note that the figure does not report GD, for which the recommendation is simply to use SPEA2, respectively. For example, if users prefer IGD then they can consult at Fig. 9b. According to the figure, they can choose either NSGA-II or SPEA2. However, if they care about mutual preference, then NSGA-II should be used. However, if the users prefer to resolve the tie by looking at the quality aspect table (Table 11) then NSGA-II or SPEA2 is the option with respect to *Spread*, *Uniformity*, and *Cardinality*, whereas SPEA2 is the option with respect to *Convergence*.

Note that, as also observed in Section 4.1.2, in our experiments we did not study relationships between QI preferences and characteristics of the search problems (e.g., search objective types, data distributions). Such characteristics could help us provide a better guidance for SBSE users based on different characteristics of search problems. Please note that conducting such an experiment requires a complete and well-planned experiment of its own, involving controlling various characteristics of search problems in a systematic way. Finding publicly available search problems that systematically cover various characteristics is challenging and one may resort to creating synthetic problems. We plan to conduct such an experiment in the future, where we could also study characteristics of QIs and search problems together to suggest appropriate MOSAs.

Based on our experience gained after conducting the 11 case studies, comprising 18 search problems (Table 3), we would like to argue that, in practice, a specific search problem often helps determine which quality aspect(s) a user should care. For instance, for a test case minimization problem, an SBSE practitioner often cares to have solutions containing diverse test cases, i.e., solutions with good *Spread*. When looking at the Rule Mining and Configuration Generation problem (Section 3.1.1), SBSE practitioners might care more about *Cardinality* because this quality aspect has influence on the level of the confidence of configuration rules. Regarding the Software Release Planning and the Requirements Allocation for Inspection problems, one might care more about the *Convergence*, *Spread* and *Uniformity* aspects, as close-to-optimal Pareto fronts usually imply more optimized requirements assignments or release plannings, and providing various options for decision makers (e.g., project managers who assign requirements to relevant stakeholders to review) to manually select from the returned solutions might be also important. Similarly, for all the test prioritization and ordering problems, we argue that one might want to get solutions featured with *Convergence* because optimal ordering of test cases is often considered as an important aspect. When keeping these application contexts in mind, the practical implication of our work is to provide a guideline on how to select a *preferred* MOSA for a given QI with specific expected quality aspects.



(a) HV

(b) IGD

(c) EP

(d) GS

(e) ED

(f) PFS

(g) C

Fig. 9 Recommendations for choosing a MOSA given a QI in a tie

6 Threats to Validity

We here discuss threats that may affect the validity of our experiments, namely *internal validity*, *conclusion validity*, *construct validity*, and *external validity* (Wohlin et al. 2012).

6.1 Internal validity

Many MOSAs have been proposed in the literature, and a threat to the internal validity is that we did not include all available MOSAs in our experiments. Considering it is practically impossible to include all MOSAs, to address this threat, we selected the MOSAs that are commonly used in SBSE (Ali et al. 2020; Wang et al. 2015). Note that part of our experimental data comes from the industry. For this type of data, it is it is difficult to re-execute the relevant industrial system when the contract is expired. Therefore, it is impossible to obtain the result data related to NSGA-III, MOEA/D or other MOSAs, which are popular in the SBSE now but not considered in our experiment.

Another threat is related to the settings of the parameters of the selected MOSAs. A MOSA M_1 may perform better (i.e., preferred by a given QI) than another MOSA M_2, because it has been configured better. In order to address this threat, we have configured the selected MOSAs by following the commonly applied guides (Arcuri and Briand 2011; Sheskin 2011). Note that these same settings were used in the papers from which we obtained the case studies, and in those papers these settings have been proven to give good results.

In terms of the selection of QIs, one may argue that we did not cover enough QIs, given that there exist more than 100 of QIs (Li and Yao 2019). However, note that we selected the most commonly used QIs in the SBSE literature (Sayyad and Ammar 2013), since our empirical evaluation was focused on SBSE problems. When presenting our results based on quality aspects, one may wonder why we did not choose other quality aspects, such as the ones proposed by Wang et al. (2015). We chose the quality aspects instead from a recent survey (Li and Yao 2019), which is based on the study of 100 QIs.

6.2 Conclusion validity

One possible threat to the conclusion validity is that the input data that we used in our experiment may not be sufficient to draw conclusions between the application of a MOSA M_1 on a given problem P, and its evaluation with a given QI. To mitigate such a threat, we have selected benchmarks in which each MOSA has been run 100 times, in order to reduce the effect of random variations. The conclusion whether a QI prefers a MOSA M_1 to a MOSA M_2 for a given problem P, is decided using the *Mann–Whitney U test* and $\hat{A}_{12}$ statistics over the distribution of 100 QI values for MOSA M_1 and MOSA M_2. Note that, in order to mitigate another threat related to wrong assumption for the tests, we selected such tests by following guidelines for conducting experiments in SBSE (Arcuri and Briand 2011).

6.3 Construct validity

One construct validity threat is that the measures we used for drawing our conclusions may not be adequate. As the first measure, we computed the percentage of times that a MOSA M_1 is preferred over another MOSA M_2 by a given QI Q, since our aim is to suggest a MOSA that will likely produce solutions preferred by Q. Hence, we believe that this metric is adequate. Moreover, to draw more stable conclusions, we also assessed the statistical significance of the results with the *Wilcoxon signed-rank* test and the $\hat{A}_{12}$ statistics. More specifically, we compared the *preference counts per problem* (see (2) and (5)) of the two MOSAs/QIs across the problems with the statistical tests for each QI/MOSA. Note that, we set the significance level of *Wilcoxon signed-rank* to 0.05, and divide $\hat{A}_{12}$ into four levels of negligible, small, medium, and large according to the report of Kitchenham et al. (2017) (see in Section 3.4.1) for testing the statistical significance of the results.

6.4 External validity

A major threat is that the results may not be generalizable to other case studies. In order to address such a threat, we selected as many search problems as possible and ended up with 18 problems in total, trying to cover different types of search problems, including rule mining in product line engineering, test optimization, requirements engineering. However, we are aware that such a selection is inherently partial, and we need more case studies and more search problems to generalize the results. The lack of real-world case studies to be used in empirical studies is recognized to be a common threat to external validity (Ali et al. 2010; Barros and Neto 2011). Note that the work presented in this paper does not aim at giving ultimate results, but at providing a methodology that should be followed to build a body of knowledge about the relationship between MOSAs and QIs. To this aim, we make data, scripts, and results publicly available online (Wu et al. 2021) and invite SBSE researchers to share with us their empirical studies, so to derive more reliable conclusions.

7 Related Work

Sayyad and Ammar (2013) presented a survey on SBSE papers that use MOSAs for solving software engineering optimization problems, from the perspectives of the chosen algorithms, QIs, and used tools. The paper concludes that more than half of the 51 surveyed papers do not provide justifications on the selection of a specific MOSA for a specific problem or simply state that a MOSA is selected because it is often applied by others. This observation, to a certain extent, implies that in the SBSE research community, there is no evidence showing which MOSA(s) to apply, in particular in the context in which researchers do know which QI(s) they prefer. Our current study provides evidence for guiding researchers in selecting a MOSA when they opt for a specific QI.

The most relevant work, though not in the SBSE context, was presented by Ravber et al. (2017). The work studied the impact of 11 QIs on the rating of 5 MOSAs: IBEA, MOEA/D, NSGA-II, PESA-II, and SPEA2, and concluded that QIs even with the same optimization goals (*convergence*, *uniformity*, and/or *spread*) might generate different and contradictory results in terms of ranking MOSAs. The authors analyzed the 11 QIs using a Chess Rating System for Evolutionary Algorithms (Veček et al. 2014), with 10 synthetic benchmark problems from the literature and 3 systems for a real-world problem. Based on the results of the analysis, the studied QIs were categorized into groups that had non-significant differences in ranking MOSAs. A set of guidelines were briefly discussed, considering preferred optimization aspects (e.g., *convergence*) when selecting QIs for a given search problem and selecting a robust (achieving the same rankings of MOSAs for different problems) and big enough set of QIs. To compare with our work reported in this paper, our study differentiates itself from the work of Ravber et al. (2017) in the following two aspects. First, our study focuses exclusively on SBSE problems, whereas their study was conducted in a more general context, which is not specific to SBSE problems, and therefore the sets of MOSAs and QIs used in the two studies are different. The MOSAs and QIs we selected in our study are commonly applied ones in the context of SBSE. Second, our study aims to provide evidence on selecting a MOSA for solving an SBSE problem, in the context in which the user is aware of the desired quality aspects in the final solutions, and has limited time budget (in terms of running experiments). Instead, their study aims to suggest which QI(s) to select for assessing MOSAs.

Li and Yao reported a survey (Li and Yao 2019) on 100 QIs from the literature, discussed their strengths and weaknesses, and presented application scenarios for a set of QIs. In this survey, only two studies (Li et al. 2018; Wang et al. 2016) related to SBSE were included, which are about understanding QIs from various aspects. Wang et al. (2016) proposed a guide for selecting QIs in SBSE based on the results of an experiment with 8 QIs, 6 MOSAs, and 3 industrial and real-world problems. Their guide helps determine a quality aspect of the QIs (*Convergence*, *Diversity*, *Combination of convergence and diversity*, or *Coverage*).

In our previous work (Ali et al. 2020), we conducted an extensive empirical evaluation with 11 SBSE search problems from industry, real-world ones, and open source ones, and automatically produced 22 observations based on the results of the statistical tests for studying QI agreements, by considering different ways in which SBSE researchers typically compare MOSAs. We also provided a set of guidelines in the form of a process that can be used by SBSE researchers. To compare with our previous work (Ali et al. 2020), in this paper, we aim to suggest which MOSA to select given a QI that is preferred, while previously, we aimed at suggesting which QI(s) to use for evaluating a given MOSA.

Li et al. (2020) surveyed 95 works to study whether these works employed appropriate quality evaluation methods. They aimed to investigate the possible issues with these methods to provide guidance for choosing suitable QIs. In contrast, our work guides users to select MOSAs given that they know the specific qualities they are looking for in the solutions represented by a QI. We achieve this by studying the preference relationships between QIs and MOSAs. Thus, our work provides guidance to the users from a complementary angle.

8 Conclusion and Future Work

In the Search-based Software Engineering (SBSE) domain, researchers and practitioners (i.e., SBSE users) solving multi-objective search problems often use one or more commonly used Multi-objective Search Algorithms (MOSAs) without any proper justification, followed by experimenting with the selected MOSAs to find the best MOSA. Furthermore, users always have limited time to experiment with a large number MOSAs. To this end, in this paper, we aim to provide evidence to users such that they can select a MOSA for their SBSE problem given their choice of a Quality Indicator (QI) or a quality aspect (e.g., *Convergence* or *uniformity*). The selected MOSA will be highly likely to provide desired solutions represented by the QI or quality aspect specified by the user. We built our evidence by running large-scale experiments consisting of 18 search problems of 11 SBSE application domains. Based on our findings, we observe that each QI has its own specific most-preferred MOSA (e.g., HV most prefers NSGA-II), and vice versa; SPEA2 and NSGA-II are the most preferred MOSAs by QIs, followed by SMPSO, and PAES is the least preferred. No QI is the most preferred by all the MOSAs; NSGA-II and SPEA2 win over the other MOSAs for *Spread*, *Uniformity*, and *Cardinality*, whereas SPEA2 is the most preferred for *Convergence*. Moreover, we notice that the preferences between QIs and MOSAs vary across the search problems, and QIs covering the same quality aspect(s) do not necessarily have the same preference for MOSAs. These observations contributed to the guidelines we devised for SBSE users to select a MOSA for a given QI or a quality aspect.

In the future, we would like to further extend our experiments with additional search problems, possibly from different SBSE application domains. Such an extension will bring more credible evidence for the relationships between QIs and MOSAs. Moreover, we intend to study the impact of various characteristics of search problems (e.g., complexity) on such relationships. Finally, we would also like to study the time performance of MOSAs while studying such relationships.

Acknowledgements This work is supported by the National Natural Science Foundation of China under Grant No. 61872182. It is also partially supported by the Co-evolver project (No. 286898/F20) funded by the Research Council of Norway. Paolo Arcaini is supported by ERATO HASUO Metamathematics for Systems Design Project (No. JPMJER1603), JST. Funding Reference number: 10.13039/501100009024 ERATO. Huihui Zhang is supported by the project (grant No. ZR2021MF026) funded by the Shandong Provincial Natural Science Foundation.

Declarations

Conflict of Interests The authors declare that they have no conflicts of interest.

A Appendix

A.1 Detailed Data for RQ 1.2

For answering RQ1.2, Table 13 presents all detailed results. For each QI and each search problem, it reports the ranking of all MOSA preferences.

Table 13 The preference relationship orders of selected MOSAs for each search problem of each QI

QI	Problem	The MOSA preference relationship order
HV	TM	NSGA-II = SPEA2 > MOCell = SMPSO > CellDE > PAES
	TP1	SMPSO > NSGA-II > SPEA2 > MOCell > CellDE > PAES
	TP2_1	NSGA-II = SPEA2 > SMPSO > MOCell = PAES > CellDE
	TP2_2	SMPSO > NSGA-II > SPEA2 > CellDE > MOCell > PAES
	TP2_3	CellDE > NSGA-II = SPEA2 > SMPSO > MOCell > PAES
	RM	SPEA2 > MOCell > NSGA-II > PAES > SMPSO
	RA	NSGA-II > SPEA2 > MOCell > PAES > SMPSO > CellDE
	TS	NSGA-II > SPEA2 > MOCell = SMPSO > CellDE > PAES
	UT1	CellDE = MOCell = NSGA-II = PAES = SMPSO = SPEA2
	UT2	CellDE = MOCell = NSGA-II = PAES = SMPSO = SPEA2
	UT3	CellDE = MOCell = NSGA-II = PAES = SMPSO = SPEA2
	UT4	CellDE = MOCell = NSGA-II = PAES = SMPSO = SPEA2
	RALIC	MOCell > PAES > SPEA2 > NSGA-II
	WORD	MOCell = SPEA2 > NSGA-II > PAES
	NRL	SPEA2 > NSGA-II > MOCell > PAES

Table 13 (continued)

QI	Problem	The MOSA preference relationship order
	TRA	SMPSO > CellDE > SPEA2 > NSGA-II > MOCell = PAES
	ITO	NSGA-II = SPEA2 > CellDE = SMPSO > MOCell > PAES
	RP	CellDE > NSGA-II > SMPSO > SPEA2 > MOCell = PAES
IGD	TM	SPEA2 > NSGA-II > MOCell = SMPSO > CellDE > PAES
	TP1	NSGA-II > SMPSO = SPEA2 > MOCell > PAES > CellDE
	TP2_1	SPEA2 > NSGA-II > PAES > MOCell > SMPSO > CellDE
	TP2_2	NSGA-II > SMPSO = SPEA2 > MOCell > PAES > CellDE
	TP2_3	CellDE > SPEA2 > NSGA-II > SMPSO > MOCell > PAES
	RM	SPEA2 > MOCell > NSGA-II > PAES > SMPSO
	RA	NSGA-II > SPEA2 > MOCell > PAES > SMPSO > CellDE
	TS	SPEA2 > NSGA-II > MOCell > SMPSO > CellDE > PAES
	UT1	CellDE = MOCell = NSGA-II = PAES = SMPSO = SPEA2
	UT2	CellDE = MOCell = NSGA-II = PAES = SMPSO = SPEA2
	UT3	CellDE = MOCell = NSGA-II = PAES = SMPSO = SPEA2
	UT4	CellDE = MOCell = NSGA-II = PAES = SMPSO = SPEA2
	RALIC	MOCell > NSGA-II > SPEA2 > PAES
	WORD	SPEA2 > NSGA-II > MOCell > PAES
	NRL	NSGA-II = SPEA2 > MOCell > PAES
	TRA	SMPSO > CellDE > SPEA2 > NSGA-II = PAES > MOCell
	ITO	NSGA-II = SPEA2 > SMPSO > CellDE > MOCell > PAES
	RP	NSGA-II > CellDE > SMPSO > SPEA2 > MOCell = PAES
EP	TM	NSGA-II = SPEA2 > MOCell = SMPSO > CellDE > PAES
	TP1	SMPSO > NSGA-II > SPEA2 > CellDE > MOCell > PAES
	TP2_1	NSGA-II = SPEA2 > MOCell = SMPSO > PAES > CellDE
	TP2_2	SMPSO > NSGA-II > SPEA2 > CellDE > MOCell > PAES
	TP2_3	CellDE > SPEA2 > NSGA-II > SMPSO > MOCell > PAES
	RM	SPEA2 > MOCell > NSGA-II > SMPSO > PAES
	RA	NSGA-II > SPEA2 > MOCell > PAES > SMPSO > CellDE
	TS	NSGA-II > SMPSO > SPEA2 > CellDE > MOCell > PAES
	UT1	CellDE = MOCell = NSGA-II = PAES = SMPSO > SPEA2
	UT2	CellDE = MOCell = NSGA-II = PAES = SMPSO = SPEA2
	UT3	CellDE = MOCell = NSGA-II = PAES = SMPSO = SPEA2
	UT4	CellDE = MOCell = NSGA-II = PAES = SMPSO = SPEA2
	RALIC	NSGA-II > MOCell > PAES > SPEA2
	WORD	NSGA-II = SPEA2 > MOCell > PAES
	NRL	SPEA2 > NSGA-II > MOCell > PAES
	TRA	SMPSO > CellDE > SPEA2 > NSGA-II > MOCell = PAES
	ITO	NSGA-II = SPEA2 > SMPSO > CellDE > MOCell > PAES
	RP	CellDE > SMPSO > NSGA-II > SPEA2 > MOCell = PAES
GD	TM	SPEA2 > NSGA-II > MOCell > SMPSO > CellDE > PAES
	TP1	MOCell = SPEA2 > NSGA-II > PAES > SMPSO > CellDE
	TP2_1	SPEA2 > NSGA-II > PAES > MOCell > SMPSO > CellDE
	TP2_2	MOCell > SPEA2 > NSGA-II = PAES > SMPSO > CellDE

Table 13 (continued)

QI	Problem	The MOSA preference relationship order
	TP2_3	NSGA-II = SPEA2 > CellDE > SMPSO > MOCell > PAES
	RM	SPEA2 > NSGA-II > MOCell > PAES > SMPSO
	RA	NSGA-II > SPEA2 > PAES > CellDE > SMPSO > MOCell
	TS	SPEA2 > NSGA-II > MOCell > SMPSO > CellDE > PAES
	UT1	CellDE = MOCell = NSGA-II = PAES = SMPSO = SPEA2
	UT2	SPEA2 > NSGA-II = PAES > CellDE = MOCell = SMPSO
	UT3	CellDE = MOCell = NSGA-II = PAES = SMPSO = SPEA2
	UT4	SPEA2 > NSGA-II = PAES > CellDE = MOCell = SMPSO
	RALIC	PAES > MOCell > SPEA2 > NSGA-II
	WORD	PAES > SPEA2 > MOCell > NSGA-II
	NRL	SPEA2 > PAES > NSGA-II > MOCell
	TRA	CellDE > SMPSO > NSGA-II > MOCell > PAES > SPEA2
	ITO	NSGA-II = SPEA2 > CellDE = SMPSO > MOCell > PAES
	RP	NSGA-II > CellDE > SPEA2 > SMPSO > PAES > MOCell
GS	TM	CellDE > SMPSO > SPEA2 > NSGA-II > MOCell > PAES
	TP1	SMPSO > SPEA2 > NSGA-II > MOCell > CellDE > PAES
	TP2_1	SMPSO > NSGA-II > CellDE = SPEA2 > MOCell > PAES
	TP2_2	SMPSO > NSGA-II > SPEA2 > CellDE > MOCell > PAES
	TP2_3	CellDE > SMPSO > NSGA-II > SPEA2 > MOCell > PAES
	RM	SPEA2 > MOCell > NSGA-II > SMPSO > PAES
	RA	CellDE = NSGA-II > PAES = SMPSO = SPEA2 > MOCell
	TS	SPEA2 > CellDE > SMPSO > NSGA-II > MOCell > PAES
	UT1	CellDE = MOCell = NSGA-II = PAES = SMPSO = SPEA2
	UT2	SPEA2 > NSGA-II = PAES > CellDE = MOCell = SMPSO
	UT3	CellDE = MOCell = NSGA-II = PAES = SMPSO = SPEA2
	UT4	SPEA2 > PAES > NSGA-II > CellDE = MOCell = SMPSO
	RALIC	SPEA2 > MOCell > NSGA-II > PAES
	WORD	SPEA2 > MOCell > NSGA-II > PAES
	NRL	SPEA2 > MOCell > NSGA-II > PAES
	TRA	CellDE > SMPSO > SPEA2 > PAES > MOCell > NSGA-II
	ITO	NSGA-II = SPEA2 > SMPSO > CellDE = MOCell > PAES
	RP	SMPSO > CellDE = NSGA-II > SPEA2 > MOCell = PAES
ED	TM	SPEA2 > NSGA-II > SMPSO > MOCell > CellDE > PAES
	TP1	MOCell = NSGA-II = SPEA2 > PAES > SMPSO > CellDE
	TP2_1	SPEA2 > NSGA-II = PAES > MOCell > SMPSO > CellDE
	TP2_2	MOCell > PAES > NSGA-II = SPEA2 > SMPSO > CellDE
	TP2_3	CellDE = SPEA2 > NSGA-II > SMPSO > MOCell > PAES
	RM	MOCell = NSGA-II = PAES = SPEA2 > SMPSO
	RA	SMPSO > CellDE > NSGA-II > SPEA2 > PAES > MOCell
	TS	SPEA2 > NSGA-II > MOCell > SMPSO > CellDE > PAES
	UT1	CellDE = MOCell = SMPSO = SPEA2 > NSGA-II = PAES
	UT2	CellDE = MOCell = NSGA-II = PAES = SMPSO = SPEA2
	UT3	CellDE = MOCell = SPEA2 > NSGA-II = PAES > SMPSO

Table 13 (continued)

QI	Problem	The MOSA preference relationship order
	UT4	CellDE = MOCell = NSGA-II = PAES = SMPSO = SPEA2
	RALIC	MOCell > PAES > SPEA2 > NSGA-II
	WORD	MOCell = SPEA2 > NSGA-II = PAES
	NRL	MOCell = SPEA2 > NSGA-II = PAES
	TRA	CellDE > SMPSO > SPEA2 > NSGA-II = PAES > MOCell
	ITO	NSGA-II > SPEA2 > CellDE = SMPSO > MOCell > PAES
	RP	NSGA-II > CellDE > SMPSO > SPEA2 > MOCell = PAES
PFS	TM	NSGA-II = SPEA2 > MOCell = PAES > CellDE = SMPSO
	TP1	MOCell = NSGA-II = SMPSO = SPEA2 > PAES > CellDE
	TP2_1	SMPSO > NSGA-II > SPEA2 > CellDE > PAES > MOCell
	TP2_2	NSGA-II = SMPSO = SPEA2 > MOCell > PAES > CellDE
	TP2_3	SMPSO > CellDE > NSGA-II > SPEA2 > PAES > MOCell
	RM	MOCell = NSGA-II = SPEA2 > PAES = SMPSO
	RA	NSGA-II = PAES > SPEA2 > CellDE > SMPSO > MOCell
	TS	NSGA-II = SPEA2 > CellDE = MOCell = PAES > SMPSO
	UT1	NSGA-II = PAES = SPEA2 > CellDE = MOCell = SMPSO
	UT2	NSGA-II = PAES = SPEA2 > CellDE = MOCell = SMPSO
	UT3	NSGA-II = PAES = SPEA2 > SMPSO > CellDE = MOCell
	UT4	NSGA-II = PAES = SPEA2 > CellDE = MOCell = SMPSO
	RALIC	NSGA-II = SPEA2 > MOCell > PAES
	WORD	MOCell = NSGA-II = PAES = SPEA2
	NRL	NSGA-II = PAES = SPEA2 > MOCell
	TRA	CellDE = NSGA-II = SMPSO = SPEA2 > MOCell > PAES
	ITO	NSGA-II = SPEA2 > PAES > CellDE > MOCell = SMPSO
	RP	NSGA-II = PAES = SPEA2 > CellDE > SMPSO > MOCell
C	TM	SPEA2 > NSGA-II > CellDE = MOCell = PAES = SMPSO
	TP1	MOCell > NSGA-II = PAES = SPEA2 > CellDE = SMPSO
	TP2_1	SPEA2 > CellDE = MOCell = NSGA-II = PAES = SMPSO
	TP2_2	MOCell > NSGA-II = PAES = SMPSO = SPEA2 > CellDE
	TP2_3	CellDE > MOCell = NSGA-II = PAES = SMPSO = SPEA2
	RM	SPEA2 > MOCell = NSGA-II > PAES > SMPSO
	RA	NSGA-II > CellDE = MOCell = PAES = SMPSO = SPEA2
	TS	NSGA-II = SPEA2 > CellDE = MOCell = PAES = SMPSO
	UT1	CellDE = MOCell = NSGA-II = PAES = SMPSO = SPEA2
	UT2	CellDE = MOCell = NSGA-II = PAES = SMPSO = SPEA2
	UT3	CellDE = MOCell = NSGA-II = PAES = SMPSO = SPEA2
	UT4	CellDE = MOCell = NSGA-II = PAES = SMPSO = SPEA2
	RALIC	MOCell = NSGA-II = PAES = SPEA2
	WORD	MOCell = NSGA-II = PAES = SPEA2
	NRL	MOCell = NSGA-II = PAES = SPEA2
	TRA	SMPSO > SPEA2 > CellDE > MOCell = NSGA-II > PAES

Table 13 (continued)

QI	Problem	The MOSA preference relationship order
	ITO	NSGA-II = SPEA2 > CellDE = MOCell = PAES = SMPSO
	RP	CellDE = NSGA-II = SMPSO > MOCell = PAES = SPEA2

- ">": preferred more than; "=": no significant difference

- "TM": Test Suite Minimization; "TP1": Test Case Prioritization-1; "TP2_*": the *th problem of Test Case Prioritization-2; "RM": Rule Mining and Configuration Generation; "RA": Requirements Allocation for Inspection; "TS": Test Case Selection; "UT*": the *th problem of Test Case Minimization; "RALIC": a problem of Uncertainty-wise Requirements Prioritization; "WORD": a problem of Uncertainty-wise Requirements Prioritization; "NRL": a problem of Uncertainty-wise Requirements Prioritization; "TRA": Testing Resource Allocation; "ITO": Integration and Test Order; "RP": Software Release Planning.

A.2 Detailed Data for RQ 2.2

For answering RQ2.2, Table 14 presents all detailed results. For each MOSA and each search problem, it reports the ranking of all QI preferences.

Table 14 The preference relationship orders of selected QIs for each search problem of each MOSA

MOSA	Problem	The QI preference relationship order
CellDE	TM	GS > HV = IGD = EP = GD = ED > PFS = C
	TP1	EP > HV = GS > IGD = GD = ED = PFS = C
	TP2_1	GS = PFS > HV = IGD = EP = GD = ED = C
	TP2_2	HV = EP = GS > IGD = GD = ED = PFS = C
	TP2_3	HV = IGD = EP = GS = C > ED = PFS > GD
	RA	GS = ED > GD = PFS > HV = IGD = EP = C
	TS	GS > EP > HV = IGD = GD = ED = PFS > C
	UT1	ED > EP > HV = IGD = GD = GS = PFS = C
	UT2	HV = IGD = EP = GD = GS = ED = PFS = C
	UT3	ED > HV = IGD = EP = GD = GS = PFS = C
	UT4	HV = IGD = EP = GD = GS = ED = PFS = C
	TRA	GD = GS = ED > HV = IGD = EP > C > PFS
	ITO	HV = IGD = EP = GD = ED > GS = PFS > C
	RP	HV = EP > IGD = GD = ED > GS = C > PFS
MOCell	TM	GD > HV = IGD = EP = ED = PFS > GS > C
	TP1	GD = C > ED > HV = IGD = GS = PFS > EP
	TP2_1	IGD = EP = GD = ED > HV = GS > PFS = C
	TP2_2	GD = ED > C > IGD = PFS > HV = EP = GS
	TP2_3	HV = IGD = EP = GD = GS = ED > PFS = C
	RM	HV = IGD = EP = GS > GD = PFS = C > ED
	RA	HV = IGD = EP > GD = GS = ED = PFS = C
	TS	IGD = GD = ED > HV > EP = GS = PFS > C
	UT1	ED > EP > HV = IGD = GD = GS = PFS = C
	UT2	HV = IGD = EP = GD = GS = ED = PFS = C
	UT3	ED > HV = IGD = EP = GD = GS = PFS = C

Table 14 (continued)

MOSA	Problem	The QI preference relationship order
	UT4	HV = IGD = EP = GD = GS = ED = PFS = C
	RALIC	HV = IGD = ED > EP = GD = GS > PFS > C
	WORD	HV = IGD = EP = GD > GS > ED = PFS = C
	NRL	GS > HV = IGD = EP = ED > GD = PFS = C
	TRA	GD > GS = PFS = C > HV = IGD = EP = ED
	ITO	HV = IGD = EP = GD = GS = ED > PFS = C
	RP	HV = IGD = EP = GD = GS = ED = PFS = C
NSGA-II	TM	HV = IGD = EP = GD = ED = PFS > C > GS
	TP1	IGD > HV = EP > GD = GS = ED > PFS = C
	TP2_1	HV = IGD = EP = GD = GS = PFS > ED > C
	TP2_2	IGD > HV = EP = GS > PFS > GD = ED > C
	TP2_3	GD > HV = IGD = EP = GS = ED = PFS > C
	RM	GD > HV = IGD = EP = GS = PFS = C > ED
	RA	HV = IGD = EP = GD > GS = PFS > ED = C
	TS	HV = EP > IGD = GD = ED > PFS > GS = C
	UT1	PFS > EP > HV = IGD = GD = GS = ED = C
	UT2	GD = GS = PFS > HV = IGD = EP = ED = C
	UT3	PFS > ED > HV = IGD = EP = GD = GS = C
	UT4	GD = GS = PFS > HV = IGD = EP = ED = C
	RALIC	EP > IGD = PFS > GS > HV = GD = ED = C
	WORD	IGD = EP > HV = GS > GD = ED = PFS = C
	NRL	HV = IGD = EP > GD = GS = PFS > ED = C
	TRA	GD > HV = EP = PFS > IGD = ED = C > GS
	ITO	ED > HV = IGD = EP = GD = GS = PFS = C
	RP	IGD = GD = ED > HV > EP = GS = PFS = C
SMPSO	TM	GS > ED > HV = IGD = EP = GD > PFS = C
	TP1	HV = EP = GS > IGD > PFS > GD = ED > C
	TP2_1	GS = PFS > HV > EP > IGD = GD = ED > C
	TP2_2	HV = EP = GS > IGD = PFS > GD = ED = C
	TP2_3	PFS > GS > HV = IGD = EP = GD = ED > C
	RM	EP = GS > HV = IGD = GD = ED = PFS = C
	RA	ED > HV = IGD = EP = GD = GS = PFS > C
	TS	EP > GS > HV = IGD = GD = ED > PFS = C
	UT1	ED > EP > HV = IGD = GD = GS = PFS = C
	UT2	HV = IGD = EP = GD = GS = ED = PFS = C
	UT3	PFS > HV = IGD = EP = GD = GS = ED = C
	UT4	HV = IGD = EP = GD = GS = ED = PFS = C
	TRA	HV = IGD = EP = C > GD = GS = ED > PFS
	ITO	IGD = EP = GS > HV = GD = ED > PFS = C
	RP	GS > EP > HV = IGD = ED = C > GD > PFS
SPEA2	TM	IGD = GD = ED = C > HV = EP = PFS > GS
	TP1	GD = GS > HV = IGD = EP = ED > PFS = C
	TP2_1	IGD = GD = ED > HV = EP > PFS = C > GS

Table 14 (continued)

MOSA	Problem	The QI preference relationship order
	TP2_2	GD > HV = IGD = EP = GS = PFS > ED > C
	TP2_3	IGD = EP = GD = ED > HV > GS = PFS > C
	RM	HV = IGD = EP = GD = GS = C > PFS > ED
	RA	HV = IGD = EP = GD > PFS > ED > GS > C
	TS	IGD = GD = GS = ED > HV = EP = PFS > C
	UT1	PFS > ED > HV = IGD = EP = GD = GS = C
	UT2	GD = GS > PFS > HV = IGD = EP = ED = C
	UT3	ED = PFS > HV = IGD = EP = GD = GS = C
	UT4	GD = GS > PFS > HV = IGD = EP = ED = C
	RALIC	GS > PFS > HV = IGD = GD = ED > EP = C
	WORD	IGD = GS > HV = EP = GD = ED > PFS = C
	NRL	HV = EP = GD = GS > IGD > ED = PFS > C
	TRA	C > HV = IGD = EP = GS = ED > PFS > GD
	ITO	HV = IGD = EP = GD = GS = ED = PFS = C
	RP	GD = PFS > HV = IGD = EP = GS = ED > C
PAES	TM	PFS > HV = IGD = EP = GD = GS = ED = C
	TP1	GD = ED = C > IGD = PFS > HV = EP = GS
	TP2_1	IGD = GD = ED > HV = EP = PFS > GS = C
	TP2_2	ED > GD > IGD = PFS = C > HV = EP = GS
	TP2_3	PFS > HV = IGD = EP = GD = GS = ED = C
	RM	HV = IGD = GD = ED = C > EP = GS = PFS
	RA	PFS > GD > HV = IGD = EP > GS = ED > C
	TS	PFS > HV = IGD = EP = GD = GS = ED = C
	UT1	PFS > EP > HV = IGD = GD = GS = ED = C
	UT2	GD = GS = PFS > HV = IGD = EP = ED = C
	UT3	PFS > ED > HV = IGD = EP = GD = GS = C
	UT4	GS > GD = PFS > HV = IGD = EP = ED = C
	RALIC	GD > HV = ED > EP > IGD = GS = PFS = C
	WORD	GD > HV = IGD = EP = GS = ED = PFS = C
	NRL	GD > PFS > HV = IGD = EP = GS = ED = C
	TRA	GS > IGD = GD = ED > HV = EP = PFS = C
	ITO	PFS > HV = IGD = EP = GD = GS = ED = C
	RP	PFS > GD > HV = IGD = EP = GS = ED = C

- ">": preferred more than; "=": no significant difference

- "TM": Test Suite Minimization; "TP1": Test Case Prioritization-1; "TP2_*": the *th problem of Test Case Prioritization-2; "RM": Rule Mining and Configuration Generation; "RA": Requirements Allocation for Inspection; "TS": Test Case Selection; "UT*": the *th problem of Test Case Minimization; "RALIC": a problem of Uncertainty-wise Requirements Prioritization; "WORD": a problem of Uncertainty-wise Requirements Prioritization; "NRL": a problem of Uncertainty-wise Requirements Prioritization; "TRA": Testing Resource Allocation; "ITO": Integration and Test Order; "RP": Software Release Planning

References

Achimugu P, Selamat A, Ibrahim R, Mahrin MN (2014) A systematic literature review of software requirements prioritization research. Inf Softw Technol 56(6):568–585

Ali S, Arcaini P, Pradhan D, Safdar SA, Yue T (2020) Quality indicators in search-based software engineering: An empirical evaluation. ACM Trans Softw Eng Methodol 29(2). https://doi.org/10.1145/3375636

Ali S, Arcaini P, Yue T (2020) Do quality indicators prefer particular multi-objective search algorithms in search-based software engineering?. In: Aleti A, Panichella A (eds) Search-Based Software Engineering. Springer International Publishing, Cham, pp 25–41

Ali S, Briand LC, Hemmati H, Panesar-Walawege RK (2010) A systematic review of the application and empirical investigation of search-based test case generation. IEEE Trans Softw Eng 36(6):742–762. https://doi.org/10.1109/TSE.2009.52

Arcuri A, Briand L (2011) A practical guide for using statistical tests to assess randomized algorithms in software engineering. In: Proceedings of the 33rd International Conference on Software Engineering, ICSE '11. ACM, New York, pp 1–10

Barros M, Neto A (2011) Threats to validity in search-based software engineering empirical studies. RelaTeDIA 5

CoelloCoello CA, ReyesSierra M (2004) A study of the parallelization of a coevolutionary multi-objective evolutionary algorithm. In: Monroy R, Arroyo-Figueroa G, Sucar LE, Sossa H (eds) MICAI 2004: Advances in Artificial Intelligence. Springer, Berlin, pp 688–697

Dantas A, Yeltsin I, Araújo AA, Souza J (2015) Interactive software release planning with preferences base. In: International Symposium on Search Based Software Engineering. Springer, pp 341–346

Deb K, Pratap A, Agarwal S, Meyarivan T (2002) A fast and elitist multiobjective genetic algorithm: NSGA-II. Trans Evol Comp 6(2):182–197. https://doi.org/10.1109/4235.996017

Durillo JJ, García-Nieto J, Nebro AJ, Coello Coello CA, Luna F, Alba E (2009a) Multi-objective particle swarm optimizers: An experimental comparison. In: International conference on evolutionary multi-criterion optimization. Springer, pp 495–509

Durillo JJ, Nebro AJ (2011) jMetal: A Java framework for multi-objective optimization. Adv Eng Softw 42(10):760–771

Durillo JJ, Zhang Y, Alba E, Nebro AJ (2009b) A study of the multi-objective next release problem. In: 2009 1st International Symposium on Search Based Software Engineering. IEEE, pp 49–58

Durillo JJ, Nebro AJ, Luna F, Alba E (2008) Solving three-objective optimization problems using a new hybrid cellular genetic algorithm. In: Parallel Problem Solving from Nature – PPSN X. Springer, Berlin, pp 661–670

Fieldsend JE, Everson RM, Singh S (2003) Using unconstrained elite archives for multiobjective optimization. IEEE Trans Evol Comput 7(3):305–323

Goh CK, Tan KC (2009) Evolutionary multi-objective optimization in uncertain environments. Issues Algorithm Stud Comput Intell 186:5–18

Greer D, Ruhe G (2004) Software release planning: an evolutionary and iterative approach. Inf Softw Technol 46(4):243–253

Guizzo G, Vergilio SR, Pozo ATR, Fritsche GM (2017) A multi-objective and evolutionary hyper-heuristic applied to the integration and test order problem. Appl Soft Comput 56:331–344

Harman M, Mansouri SA, Zhang Y (2012) Search-based software engineering: Trends, techniques and applications. ACM Comput Surv (CSUR) 45(1):1–61

Karim MR, Ruhe G (2014) Bi-objective genetic search for release planning in support of themes. In: International Symposium on Search Based Software Engineering. Springer, pp 123–137

Kitchenham B, Madeyski L, Budgen D, Keung J, Brereton P, Charters S, Gibbs S, Pohthong A (2017) Robust statistical methods for empirical software engineering. Empir Softw Engg 22(2):579–630. https://doi.org/10.1007/s10664-016-9437-5

Knowles J, Corne D (2002) On metrics for comparing nondominated sets. In: Proceedings of the 2002 Congress on Evolutionary Computation. CEC'02 (Cat. No. 02TH8600), vol 1. IEEE, pp 711–716

Knowles JD, Corne DW (2000) Approximating the nondominated front using the pareto archived evolution strategy. Evol Comput 8(2):149–172. https://doi.org/10.1162/106365600568167

Knowles JD, Thiele L, Zitzler E (2006) A tutorial on the performance assessment of stochastic multiobjective optimizers. TIK-report 214

Li M, Chen T, Yao X (2018) A critical review of: "a practical guide to select quality indicators for assessing Pareto-based search algorithms in search-based software engineering": Essay on quality indicator selection for SBSE. In: Proceedings of the 40th International Conference on Software Engineering:

New Ideas and Emerging Results, ICSE-NIER '18. ACM, New York, pp 17–20. https://doi.org/10.1145/3183399.3183405

Li M, Yao X (2019) Quality evaluation of solution sets in multiobjective optimisation: A survey. ACM Comput Surv 52(2). https://doi.org/10.1145/3300148

Li M, Chen T, Yao X (2020) How to evaluate solutions in pareto-basedsearch-based software engineering? a critical review and methodological guidance. IEEE Transactions on Software Engineering

Lim SL (2011) Social networks and collaborative filtering for large-scale requirements elicitation. Ph.D. Thesis, University of New South Wales

McMinn P (2004) Search-based software test data generation: a survey. Softw Test Verif Reliab 14(2):105–156

McMinn P (2011) Search-based software testing: Past, present and future. In: 2011 IEEE Fourth International Conference on Software Testing, Verification and Validation Workshops. IEEE, pp 153–163

Nebro AJ, Durillo JJ, Garcia-Nieto J, CoelloCoello CA, Luna F, Alba E (2009) SMPSO: A new PSO-based metaheuristic for multi-objective optimization. In: 2009 IEEE Symposium on Computational Intelligence in Multi-Criteria Decision-Making(MCDM), pp 66–73

Nebro AJ, Durillo JJ, Luna F, Dorronsoro B, Alba E (2009) MOCell: A cellular genetic algorithm for multiobjective optimization. Int J Intell Syst 24(7):726–746. https://doi.org/10.1002/int.v24:7

Pitangueira AM, Tonella P, Susi A, Maciel RS, Barros M (2016) Risk-aware multi-stakeholder next release planning using multi-objective optimization. In: International Working Conference on Requirements Engineering: Foundation for Software Quality. Springer, pp 3–18

Pradhan D, Wang S, Ali S, Yue T (2016a) Search-based cost-effective test case selection within a time budget: An empirical study. In: Proceedings of the Genetic and Evolutionary Computation Conference 2016, GECCO '16. ACM, New York, pp 1085–1092

Pradhan D, Wang S, Ali S, Yue T, Liaaen M (2016b) STIPI: Using search to prioritize test cases based on multi-objectives derived from industrial practice. In: IFIP International Conference on Testing Software and Systems. Springer, pp 172–190

Pradhan D, Wang S, Ali S, Yue T, Liaaen M (2018) REMAP: Using rule mining and multi-objective search for dynamic test case prioritization. In: Software Testing, Verification and Validation (ICST), 2018 IEEE 11th International Conference on. IEEE, pp 46–57

Pradhan D, Wang S, Ali S, Yue T, Liaaen M (2021) CBGA-ES+: A cluster-based genetic algorithm with non-dominated elitist selection for supporting multi-objective test optimization. IEEE Trans Softw Eng 47(1):86–107. https://doi.org/10.1109/TSE.2018.2882176

Ramírez A, Romero JR, Ventura S (2014) On the performance of multiple objective evolutionary algorithms for software architecture discovery. In: Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation, pp 1287–1294

Ravber M, Mernik M, Črepinšek M (2017) The impact of quality indicators on the rating of multi-objective evolutionary algorithms. Appl Soft Comput 55(C):265–275. https://doi.org/10.1016/j.asoc.2017.01.038

Safdar SA, Lu H, Yue T, Ali S (2017) Mining cross product line rules with multi-objective search and machine learning. In: Proceedings of the Genetic and Evolutionary Computation Conference, GECCO '17. ACM, New York, pp 1319–1326

Sayyad AS, Ammar H (2013) Pareto-optimal search-based software engineering (POSBSE): A literature survey. In: 2013 2nd International Workshop on Realizing Artificial Intelligence Synergies in Software Engineering (RAISE), pp 21–27

Shang K, Ishibuchi H, He L, Pang LM (2021) A survey on the hypervolume indicator in evolutionary multiobjective optimization. IEEE Trans Evol Comput 25(1):1–20. https://doi.org/10.1109/TEVC.2020.3013290

Sheskin DJ (2011) Handbook of Parametric and Nonparametric Statistical Procedures, 5 edn. Chapman & Hall/CRC

Spieker H, Gotlieb A, Marijan D, Mossige M (2017) Reinforcement learning for automatic test case prioritization and selection in continuous integration. In: Proceedings of the 26th ACM SIGSOFT International Symposium on Software Testing and Analysis. ACM, pp 12–22

Van Veldhuizen DA, Lamont GB (1998) Evolutionary computation and convergence to a Pareto front. In: Koza JR (ed) Late Breaking Papers at the Genetic Programming 1998 Conference. Stanford University Bookstore, University of Wisconsin, Madison, pp 221–228

VanVeldhuizen DA (1999) Multiobjective evolutionary algorithms: classifications, analyses, and new innovations. Air Force Institute of Technology

Vargha A, Delaney HD (2000) A critique and improvement of the CL common language effect size statistics of McGraw and Wong. J Educ Behav Stat 25(2):101–132

Veček N, Mernik M, Črepinšek M (2014) A chess rating system for evolutionary algorithms: A new method for the comparison and ranking of evolutionary algorithms. Inf Sci 277:656–679. https://doi.org/10.1016/j.ins.2014.02.154

Wang S, Ali S, Gotlieb A (2015) Cost-effective test suite minimization in product lines using search techniques. J Syst Softw 103:370–391

Wang S, Ali S, Yue T, Li Y, Liaaen M (2016) A practical guide to select quality indicators for assessing Pareto-based search algorithms in search-based software engineering. In: Proceedings of the 38th International Conference on Software Engineering, ICSE '16. ACM, New York, pp 631–642. https://doi.org/10.1145/2884781.2884880

Wang Z, Tang K, Yao X (2008) A multi-objective approach to testing resource allocation in modular software systems. In: 2008. CEC 2008.(IEEE World Congress on Computational Intelligence). IEEE Congress on Evolutionary Computation. IEEE, pp 1148–1153

Wilcoxon F (1992) Individual comparisons by ranking methods. In: Kotz S, Johnson NL (eds) Breakthroughs in Statistics: Methodology and Distribution. Springer, New York, pp 196–202

Wohlin C, Runeson P, Hst M, Ohlsson MC, Regnell B, Wessln A (2012) Experimentation in Software Engineering. Springer Publishing Company, Incorporated

Wu J, Arcaini P, Yue T, Ali S, Zhang H (2021) Repository for the paper "On the Preferences of Quality Indicators for Multi-Objective Search Algorithms in Search-Based Software Engineering". https://github.com/wjh-test/Quality-Indicator-2021

Yue T, Ali S (2014) Applying search algorithms for optimizing stakeholders familiarity and balancing workload in requirements assignment. In: Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation, GECCO '14. ACM, New York, pp 1295–1302

Zeleny M (1973) Compromise programming. In: Cochrane J, Zeleny M (eds) Multiple Criteria Decision Making. University of South Carolina Press, Columbia, pp 262–301

Zhang H, Zhang M, Yue T, Ali S, Li Y (2020) Uncertainty-wise requirements prioritization with search. ACM Trans Softw Eng Methodol (TOSEM) 30(1):1–54

Zhang M, Ali S, Yue T (2019) Uncertainty-wise test case generation and minimization for cyber-physical systems. J Syst Softw 153:1–21

Zhou A, Jin Y, Zhang Q, Sendhoff B, Tsang E (2006) Combining model-based and genetics-based offspring generation for multi-objective optimization using a convergence criterion. In: 2006 IEEE international conference on evolutionary computation. IEEE, pp 892–899

Zitzler E, Laumanns M, Thiele L (2002) SPEA2: Improving the strength Pareto evolutionary algorithm. In: EUROGEN 2001. Evolutionary Methods for Design, Optimization and Control with Applications to Industrial Problems, pp 95–100

Zitzler E, Thiele L (1998) Multiobjective optimization using evolutionary algorithms — a comparative case study. In: Eiben AE, Bäck T, Schoenauer M, Schwefel H-P (eds) Parallel Problem Solving from Nature — PPSN V. Springer, Berlin, pp 292–301

Zitzler E, Thiele L, Laumanns M, Fonseca CM, DaFonseca VG (2003) Performance assessment of multiobjective optimizers: An analysis and review. IEEE Trans Evol Comput 7(2):117–132

Jiahui Wu

Paolo Arcaini

Tao Yue

Shaukat Ali

Huihui Zhang

Affiliations

Jiahui Wu[1] · Paolo Arcaini[2] · Tao Yue[1,3] · Shaukat Ali[3] · Huihui Zhang[4,5]

Jiahui Wu
wjh_sx2016068@nuaa.edu.cn

Paolo Arcaini
arcaini@nii.ac.jp

Shaukat Ali
shaukat@simula.no

Huihui Zhang
huihui@wfu.edu.cn

[1] Nanjing University of Aeronautics and Astronautics, Nanjing, China

[2] National Institute of Informatics, Tokyo, Japan

[3] Simula Research Laboratory, Oslo, Norway

[4] School of Computer Science and Technology, Qilu University of Technology (Shandong Academy of Sciences), Jinan, China

[5] School of Computer Engineering, Weifang University, Weifang, China