Quality Indicators in Search-Based Software Engineering: An Empirical Evaluation

SHAUKAT ALI, Simula Research Laboratory
PAOLO ARCAINI, National Institute of Informatics
DIPESH PRADHAN, Simula Research Laboratory
SAFDAR AQEEL SAFDAR, Simula Research Laboratory
TAO YUE, Nanjing University of Aeronautics and Astronautics and Simula Research Laboratory

Search-Based Software Engineering (SBSE) researchers who apply multi-objective search algorithms (MOSAs) often assess the quality of solutions produced by MOSAs with one or more quality indicators (QIs). However, SBSE lacks evidence providing insights on commonly used QIs, especially about agreements among them and their relations with SBSE problems and applied MOSAs. Such evidence about QIs agreements is essential to understand relationships among QIs, identify redundant QIs, and consequently devise guidelines for SBSE researchers to select appropriate QIs for their specific contexts. To this end, we conducted an extensive empirical evaluation to provide insights on commonly used QIs in the context of SBSE, by studying agreements among QIs with and without considering differences of SBSE problems and MOSAs. In addition, by defining a systematic process based on three common ways of comparing MOSAs in SBSE, we present additional observations that were automatically produced based on the results of our empirical evaluation. These observations can be used by SBSE researchers to gain a better understanding of the commonly used QIs in SBSE, in particular regarding their agreements. Finally, based on the results, we also provide a set of guidelines for SBSE researchers to select appropriate QIs for their particular context.

CCS Concepts: • **Software and its engineering → Search-based software engineering**;

Additional Key Words and Phrases: Search-based Software Engineering, Quality Indicator, Agreement, Multi-Objective Search Algorithm

ACM Reference format:
Shaukat Ali, Paolo Arcaini, Dipesh Pradhan, Safdar Aqeel Safdar, and Tao Yue. 2019. Quality Indicators in Search-Based Software Engineering: An Empirical Evaluation. *ACM Trans. Softw. Eng. Methodol.* X, Y, Article Z (December 2019), 30 pages.
https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

Multi-objective optimization problems in Search-Based Software Engineering (SBSE) [18] are common (e.g., release planning, requirements prioritization, and test optimization [1, 19, 25, 27, 33]). Evidence shows that the applications of Multi-Objective Search Algorithms (MOSAs) to solve such SBSE problems have increased significantly [46]. When evaluating the performance of MOSAs for solving these problems, various Quality Indicators (QIs) have been proposed such as *Hypervolume*

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
© 2019 Association for Computing Machinery.
1049-331X/2019/12-ARTZ $15.00
https://doi.org/10.1145/1122445.1122456

(HV) [49]. Given the solutions provided by two MOSAs to an SBSE problem, a QI tells which MOSA has a better performance than the other or if there is no difference.

QIs have been designed to assess the quality of solutions produced by MOSAs from different quality aspects, i.e., *convergence*, *uniformity*, *cardinality*, or *spread* [23]. Evidence [34] has shown that even QIs that share the same quality aspects may lead to contradictory *responses*, e.g., two QIs indicating that different MOSAs are preferred for the same problem. Thus, it is important to study agreements among QIs and their relations with problems and MOSAs.

As reported in a recent survey by Li and Yao [23], there are studies focusing on understanding particular aspects of existing QIs in the evolutionary computation domain. The survey also emphasizes that it is important to understand the relationships among QIs in order to select a minimum subset of QIs that represents a larger set of QIs, i.e., identifying redundant QIs. However, the SBSE literature lacks such studies on building knowledge on QI agreements and their relations with SBSE problems and MOSAs. The survey by Li and Yao [23] also showed that there exist only two related studies in the context of SBSE. To this end, in this paper, we report an extensive empirical evaluation to give insights into QI agreements and their relations with SBSE problems and MOSAs, mainly for the SBSE researchers who want to select appropriate QIs for evaluating MOSAs to address SBSE problems. We use 8 QIs and 6 MOSAs commonly used in SBSE [12], and 9 SBSE problems (11 case studies, each of which corresponds to one search problem)[1] covering requirements inspection, test optimization, and rule mining in product line engineering.

We see three key contributions in this paper. *First*, we provide insights into how the QI agreements relate to the MOSAs and the SBSE problems by applying various agreement statistics, including *Kappa* [8], and the *Bowker* test [2]. Results showed that the differences in the SBSE problems have more impact on QI agreements, but, on the other hand, the impact of MOSAs on QI agreements is minimal. *Second*, we provide a process to automatically generate additional observations from data collected via an extensive experiment using 9 SBSE problems with 11 case studies from the industry, real-world, and open-source. Our observations are based on statistical tests for studying QI agreements. These observations are grouped for accommodating three common ways that SBSE researchers use for comparing MOSAs in their SBSE experiments: comparing all the selected MOSAs with each other, a pair of MOSAs, and comparing one MOSA with more than one chosen MOSAs. *Third*, based on the automatically generated 22 observations, we provide a set of guidelines for SBSE researchers to follow. With these guidelines, SBSE researchers can select the QIs according to one of the following two cases: a) selecting the most representative QI, or b) selecting a minimum representative set of the QIs. In both cases, an SBSE researcher could be interested in considering only one particular MOSA, a given pair of MOSAs, or all the MOSAs. Note that our process to generate observations is replicable, and thus can be applied to further enrich the observations based on new data from other experiments.[2] In addition, this process can be applied to other QIs than the ones studied in this paper to build a more comprehensive view of QI agreements.

The rest of the paper is organized as follows. Sect. 2 introduces QIs used in our experiments, and Sect. 3 presents related work. Experiments, results and analyses are reported in Sect. 4 and Sect. 5, respectively. Additional analyses, observations, and guidelines are provided in Sect. 6. Threats to validity are presented in Sect. 7. We conclude the paper in Sect. 8.

[1] One of the SBSE problems in our experiment had three case studies. Thus, in total, we have 11 search problems.

[2] Data, R scripts, Java code, and a tool for generating observations, are available online at https://github.com/ERATOMMSD/QIsAgreementMOSAs.

2 BACKGROUND ON QUALITY INDICATORS

A *multi-objective optimization problem* consists of a set of m objectives to be optimized (either maximized or minimized). It can be formally defined as $F(x) = (f_1(x), \ldots, f_m(x))$ where $x = (x_1, \ldots, x_n)$ is an n-dimensional decision variable vector defined over a universe X. Each $f_i(x)$ is an objective function. Also, there are p equality constraints and k inequality constraints that must be satisfied by any solution. Without loss of generality, we assume that all the objective functions must be minimized.[3] A solution is an element of X that satisfies the inequality and equality constraints. Let's say $\Omega \subseteq X$ be the set of solutions; given two solutions $A, B \in \Omega$, A dominates B ($A \succ B$) iff, for all the objectives, A is never worse than B, and A is better than B for at least one objective:

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

For simplicity, in the following, a solution A is considered in the objective space, i.e., $A = (f_1(x), \ldots, f_m(x))$ with $x \in \Omega$. Given a set of computed solutions S, a Pareto front $PF(S)$ is the subset of solutions that are not dominated by others:

$$PF(S) = \{s_i \in S \mid (\nexists s_j \in S : s_j \succ s_i)\} \tag{2}$$

The *optimal* (or *true*) Pareto front PF^o is the set of non-dominated solutions over all the solution space.

Let $alg_1, \ldots, alg_k$ be a set of MOSAs and $PF_1^c, \ldots, PF_k^c$ be their computed Pareto fronts for a given search problem. To assess the quality of computed Pareto fronts, various QIs (e.g., HV) were proposed [23, 39]. Most QIs require the *optimal* Pareto front PF^o, which, however, in general is computation-wise infeasible to be calculated for any complex search problem. Hence, as an approximation to the optimal Pareto front, the *reference* Pareto front is often computed from all the available solutions. Let $ALL\_PF = \cup_{i=1}^{k} PF_i^c$ be the union of all the computed Pareto fronts; the reference Pareto front is defined as:

$$PF^{ref} = \{s_i \in ALL\_PF \mid (\nexists s_j \in ALL\_PF : s_j \succ s_i)\} \tag{3}$$

Let d be the minimum Euclidean distance of a solution $A = (a_i, \ldots, a_m)$ from a Pareto front PF,

$$d((a_i, \ldots, a_m), PF) = \min_{(p_i, \ldots, p_m) \in PF} \sqrt{\sum_{i=1}^{m} (a_i - p_i)^2} \tag{4}$$

Moreover, let PF^c be the computed Pareto front that we want to evaluate. Below, we introduce the QIs that we selected for our empirical evaluation. Note that we selected these QIs since they are the most commonly used ones in SBSE and cover one or more quality aspects, i.e., convergence, spread, cardinality, and uniformity, as defined by Li and Yao [23].

Generational Distance (GD) is the distance between the solutions in PF^c and the nearest solutions in PF^{ref}:

$$GD(PF^c) = \frac{\sqrt{\sum_{s \in PF^c} d(s, PF^{ref})^2}}{|PF^c|} \tag{5}$$

GD=0 tells that all the computed solutions are optimal.

Euclidean Distance from the Ideal Solution (ED) is the Euclidean distance between the *ideal solution* and the closest solution in PF^c [15, 38]: $ED(PF^c) = d(s_{ideal}, PF^c)$. The ideal solution s_{ideal} consists of all the optimal values for each objective obtained from the solutions in PF^c.

Epsilon (EP) uses the notion of epsilon-dominance $\succ_\epsilon$. Given two solutions $A = (a_1, \ldots, a_m)$ and $B = (b_1, \ldots, b_m)$, $A \succ_\epsilon B$ iff $\forall i \in \{1, \ldots, m\} : a_i < b_i + \epsilon$. EP is defined as:

$$EP(PF^c) = \inf\{\epsilon \in \mathbb{R} \mid (\forall x \in PF^{ref}, \exists y \in PF^c : y \succ_\epsilon x)\} \tag{6}$$

[3] Maximization search problems can be converted to equivalent minimization ones.

Table 1. QIs and various Quality Aspects [22]

Quality Aspect	GD	ED	EP	GS	PFS	IGD	HV	C
Convergence	+	-	+			+	+	-
Spread			+	-		+	+	
Uniformity			+	+		+	+	
Cardinality			-		+	-	-	

A "+" means that the quality aspect is fully covered by a QI, whereas a "-" means that the quality aspect is partially covered by a QI.

EP measures the smallest distance that each solution in PF^c should be translated so that PF^c dominates PF^{ref} [10].

Generalized Spread (GS): Let $(e_1, \ldots, e_m)$ be the *extreme solution* of PF^{ref} such that each e_i provides the optimal value for objective function f_i. *GS* is then defined as:

$$GS(PF^c) = \frac{\sum_{i=1}^{m} d(e_i, PF^c) + \sum_{s \in PF^c} |id(s, PF^c) - \overline{id}|}{\sum_{i=1}^{m} d(e_i, PF^c) + |PF^c| * \overline{id}} \tag{7}$$

where $id(s, PF^c) = d(s, PF^c \setminus \{s\})$ is the minimal distance of a solution s from all the other solutions in PF^c, and $\overline{id}$ is the mean value of $id(s, PF^c)$ across the solutions s of PF^c [47]. GS aims at measuring the *spread* of solutions in PF^c.

Pareto front size (PFS) counts the number of solutions in PF^c (i.e., $PFS(PF^c) = |PF^c|$) [10].

Inverted Generational Distance (IGD) is given by the distance from the solutions in PF^{ref} to the nearest solutions in PF^c [39]. It is defined as:

$$IGD(PF^c) = \frac{\sqrt{\sum_{s \in PF^{ref}} d(s, PF^c)^2}}{|PF^{ref}|} \tag{8}$$

Hypervolume (HV) measures the volume of the objective space covered by PF^c [49]. Let w be a *reference point* computed using the worst objective function values among all solutions in PF^c. For each solution $s_i \in PF^c$, $hc(s_i)$ is the hypercube having s_i and w as diagonal corners. HV is defined as:

$$HV(PF^c) = volume(\cup_{s_i \in PF^c} hc(s_i)) \tag{9}$$

Coverage (C) measures how well PF^c covers the solutions of PF^{ref}:

$$C(PF^c) = \frac{|PF^c \cap PF^{ref}|}{|PF^{ref}|} \tag{10}$$

Li and Yao [23] described four quality aspects covered by various QIs. These quality aspects are: 1) *Convergence* measures the closeness of a set of solutions to the PF^{ref}; 2) *Spread* measures the spread of a set of solutions; 3) *Uniformity* measures how even/uniformly solutions are distributed in a set of solutions; and 4) *Cardinality* counts the number of solutions in a solution set. Li at al. [22] analyzed which of the previous four quality aspects are fully and partially covered by the QIs that we consider in this paper. Table 1 reports their classification. Since they studied the exactly same QIs as ours in the context of SBSE, we chose their classification in this paper to explain the results.

3 RELATED WORK

Recently, Li and Yao [23] surveyed and classified 100 QIs from the literature. In addition, they discussed the strengths and weaknesses of the studied QIs, which users are recommended to take into consideration when selecting QIs for their work. They also presented application scenarios for

a representative set of QIs. As found in their survey, there were only two studies related to SBSE reported in the literature that focus on understanding certain aspects of QIs [22, 42]. Moreover, Li and Yao [23] emphasized that studying relationships among QIs with the aim of selecting a minimum number of QIs representing a bigger set of QIs is important and there is a need for more studies in this area, especially in SBSE. Below, we compare our empirical evaluation with these two studies and one general study that are also closely related to our work.

Wang et al. [42] proposed a guide for selecting QIs in SBSE based on the results of an extensive experiment evaluating 8 QIs (i.e., GD, ED, EP, GS, PFS, IGD, HV, and C) with 6 MOSAs (NSGA-II, MoCell, SPEA2, PAES, SMPSO, and CellDE) with 3 industrial and real-world problems, in addition to a literature review and theory of QIs. The initial and important step of the guide is to determine a category of the QIs (*Convergence*, *Diversity*, *Combination of convergence and diversity*, or *Coverage*). Results show that (i) for *Convergence* and *Combination*, it does not matter which QI within the same category to select but it matters for *Diversity*, and (ii) it matters to select QIs across the four categories except for *Convergence* and *Coverage*. As compared to Wang et al. [42], we automatically produce observations based on an extensive empirical evaluation that was conducted with 9 SBSE problems from industry, real-world, and open sources. Moreover, we produced 22 observations based on the results of the statistical tests for studying QI agreements by considering different ways of SBSE researchers typically comparing MOSAs (i.e., comparing all the selected MOSAs with each other, a pair of MOSAs, and comparing one MOSA with the other selected MOSAs). Furthermore, we provide a set of guidelines in the form of a process that can be directly used by SBSE researchers. Our goal is to help SBSE researchers in selecting QIs without for instance knowing the classification of QIs in Wang et al. [42]. In terms of selecting QIs, Wang et al. [42] proposed a guide based on the results of an experiment for evaluating 8 QIs with three industrial and real-world problems and 6 MOSAs. Later on, Li et al. [22] critically reviewed the guide [42] by debating the classification of QIs by Wang et al. [42], and then proposed a clarified guide. But, this clarified guide [22] is not based on evidence from experiments on SBSE problems. In conclusion, to compare with [22, 42], *first*, we have more SBSE problems (i.e., 9 instead of 3 as compared to [42]). *Second*, we provide insights into agreements among QIs while studying their relations with MOSAs and SBSE problems by applying agreement statistics. *Third*, we propose a process to automatically generate observations for understanding QIs. *Fourth*, we developed a software tool implementing the process to automatically generate observations (see Sect. 6.1). This means that when more data from other experiments is available, it can be fed to our software, which can then produce more accurate observations based on a larger sample of data. Also, we automatically generated 22 observations from the current data and provided insights into the commonly used QIs in SBSE.

Ravber et al. [34] (although not on SBSE) studied the impact of 11 QIs on the rating of 5 MOSAs (i.e., IBEA, MOEA/D, NSGA-II, PESA-II, and SPEA2). They analyzed the QIs using a Chess Rating System for Evolutionary Algorithms [40] and used 10 synthetic benchmark problems from the literature and 3 systems for a real-world problem. Based on the results, the studied QIs were categorized into groups that had insignificant differences in ranking MOSAs. Inconsistent sets of groups were obtained for two scenarios: one for the 10 synthetic problems and the other for the real-world case studies. A set of guidelines were briefly discussed in the paper: 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 [34] and works reviewed in the extensive survey in [23], the key novelties of our empirical evaluation are sixfold. *First*, we study agreements among the commonly used QIs, i.e., a new way of understanding QIs via agreement statistics. *Second*, we focus particularly on SBSE problems, as compared to [22, 34], and such empirical evaluations are rare [23]. *Third*, we perform an extensive empirical evaluation based on 9 SBSE problems and 11 industrial, real-world,

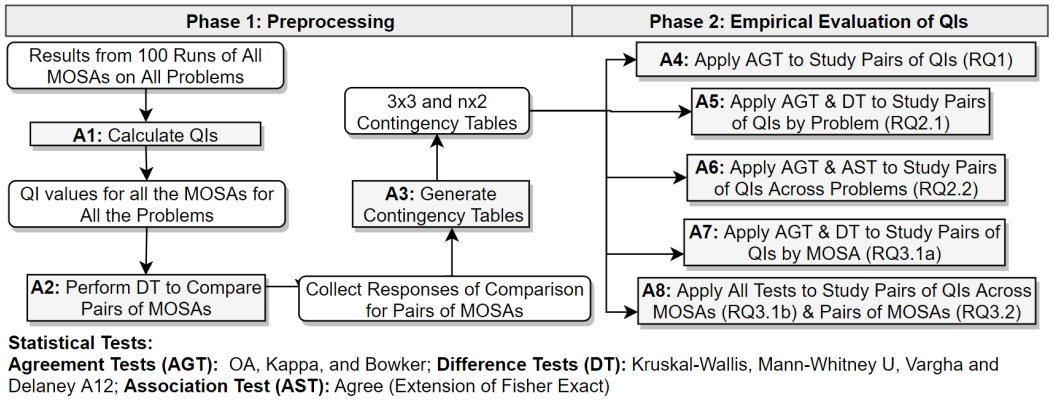


Fig. 1. Experiment Design and Execution

and open-source case studies in SBSE, which is also rarely seen in the literature [23]. *Fourth*, we provide a process to automatically generate additional observations and guidelines for selecting QIs. *Fifth*, based on the automatically generated observations, we provide a set of guidelines to guide SBSE researchers to select QIs. *Sixth*, we developed a software tool implementing the process to automatically generate observations (see Sect. 6.1).

4 EXPERIMENT DESIGN AND EXECUTION

Fig. 1 provides an overview of experimental design and execution having: research questions (RQs), activities (A1-A8) to answer the RQs, and metrics and statistical tests applied for analyzing results. Our empirical evaluation has two phases: *Phase 1* preprocesses data produced by 100 runs of the selected MOSAs for all the studied search problems such that it can be used for performing the planned analyses, and *Phase 2* performs statistical analyses to study QIs. Note that our empirical evaluation is based on SBSE problems that were available to us. Thus, the results only provide evidence for the SBSE problems. To generalize the results to non-SBSE problems, additional empirical evaluation with search problems from non-SBSE domains is needed.

4.1 Research Questions

Recall from Sect. 1 that our goal is to study agreements among the QIs with respect to SBSE problems and MOSAs with the ultimate aim of devising a set of guidelines for SBSE researchers to select QIs for their particular context. To achieve our goal, we devised three research questions (RQs) focusing on studying the overall agreements (RQ1), agreements with respect to search problems (RQ2), and agreements with respect to MOSAs (RQ3).

- **RQ1**: Do a pair of QIs agree with each other (named as *agreements*) when evaluating the quality of the MOSAs?
  This RQ aims to observe the overall trend of agreements among the QI pairs irrespective of problems and MOSAs.
- **RQ2**: How do the SBSE problems relate to QI agreements? We study the QI agreements by search problem.[4] This RQ is answered via the following two sub-RQs:
  – **RQ2.1**: How do the QI agreements vary across the search problems?

[4]Since for one SBSE problem, we had 3 case studies, i.e., 3 search problems, we will answer RQ2.1 & RQ2.2 based on search problems.

Table 2. Selected Problems, Case Studies, and Objective

Category	SBSE Problem	# Case Studies (search problems)	# Objectives
Industrial	Test Suite Minimization [41]	1	4
	Test Case Prioritization-1 [31]	1	4
	Test Case Prioritization-2 [32]	3	2
	Rule Mining and Configuration Generation [35]	1	3
Real-World	Requirements Allocation for Inspection [45]	1	3
	Test Case Selection [30]	1	4
Open-Source	Testing Resource Allocation [44]	1	2
	Integration and Test Order [17, 44]	1	4
	Software Release Planning [9, 44]	1	3

  - **RQ2.2**: How does the agreement of a QI pair vary across the search problems?
    Note that the difference between RQ2.1 and RQ2.2 is that RQ2.1 studies the overall QI agreements of all the QI pairs together, whereas RQ2.2 studies one QI pair at a time.
- **RQ3**: How do the MOSAs relate to QI agreements? We study the QI agreements by MOSA (RQ3.1, further decomposed into two sub-RQs), and by pairs of MOSAs (RQ3.2).
  - **RQ3.1a**: How do the QI agreements vary across the MOSAs?
  - **RQ3.1b**: How does the agreement of a QI pair vary across the MOSAs?
  - **RQ3.2**: How does the agreement of a QI pair vary across pairs of the MOSAs?

Note that within RQ2 and RQ3, we study all the QI pairs (i.e., 28 pairs) together (RQ2.1a, and RQ3.1a), and each QI pair in all the three RQs (RQ1, RQ2.2b, RQ3.1b, and RQ3.2). Studying all the QI pairs together helps to assess if the differences in agreements exist across all the pairs. On the other hand, studying individual QI pairs help us to identify redundant QIs (i.e., two QIs with no significant differences), that helps us to find the minimum number of QIs to be applied in a particular context.

4.2 SBSE Problems, Case Studies, and Objectives

Table 2 illustrates the 9 SBSE problems, 11 case studies (11 search problems), and their corresponding number of objectives. We classify SBSE problems into three main categories, as shown in the *Category* column, i.e., Industrial, Real-World, and Open-Source. An SBSE problem is industrial when both the SBSE problem and corresponding case studies were provided by our industrial partner. An SBSE problem is categorized as real-world when we identified the SBSE problem from the industry by working closely with our industrial partner; however, the case studies for the SBSE problem were created by ourselves based on real data based on information from various sources such as publicly available documents (e.g., standards, regulations). An SBSE problem is categorized as open-source, when an SBSE problem and the corresponding case studies are available in a publication including the description and implementation of the problem and the case studies.

Each SBSE problem has one case study except for the second Test Case Prioritization problem, which has three case studies. The 11 case studies correspond to 9 SBSE problems that we analyzed in this paper. All 9 problems have been widely investigated by the state-of-the-art. Specifically, Test Suite Minimization [41], Test Case Prioritization-1 [31], Test Case Prioritization-2 [32], Rule Mining and Configuration Generation [35], Requirements Allocation for Inspection [45], and Test Case Selection [30] were extracted based on our collaboration with the industrial partners, while Testing Resource Allocation, Integration and Test Order, and Software Release Planning were

obtained from a well-known SBSE repository hosted by the CREST center [46]. The "*SBSE Problem*" column in Table 2 provides the references to the published literature. The selected SBSE problems are representative of various software engineering problems such as in testing and requirements engineering.

4.2.1 Industrial Case Studies. Since 2007, we have established a close collaboration with Cisco Systems Norway, focusing on improving the cost and effectiveness of testing a variety of Video Conferencing Systems (VCSs) [41]. The core functionality of a VCS is to establish a video conference among participants at different physical locations. There is also a possibility of transmitting presentations in parallel to a video conference using VCSs. Each VCS has on average three million lines of embedded C code [43]. Such large-scale VCSs require thorough testing before their release in the market. However, testing the VCSs is expensive due to a large number of test cases required to be executed with different hardware and network settings. Thus, it requires optimizing the testing process to reduce the cost of testing while preserving effectiveness. Moreover, we identified another problem of learning rules on the configurations of the communicating VCSs (via information networks) belonging to different families. These rules describe how the configurations influence the run-time interactions among the VCSs.

Test Suite Minimization: This problem aims at eliminating redundant test cases while maximizing the effectiveness (e.g., fault detection ability, and feature coverage) and minimizing the cost of testing (e.g., time). For this problem, our test suite consists of 489 test cases from Cisco. Moreover, we defined four optimization objectives together with test engineers from Cisco. Details can be found in [41].

Test Case Prioritization-1 and Test Case Prioritization-2: Test Case Prioritization-1 [31] prioritizes test cases into an optimal order to meet specific criteria (e.g., higher fault detection capability) as early as possible. Our test suite consists of 211 test cases from Cisco and consisted of four optimization objectives.

Test Case Prioritization-2 [32] addresses the test case prioritization problem for black-box testing using two objectives aimed at maximizing fault detection capability and test case reliance score measuring the number of unique test cases whose results can be predicted by executing the prioritized test cases. We used three case studies. The first two case studies consist of 60 and 624 test cases from Cisco for testing various VCSs. The third case study is from ABB Robotics for testing Paint Control system in a painting robot [37] and consists of 89 test cases.[5] The paint control system is responsible for the overall process of accurate and precise painting, while at the same time optimizing the paint consumption of the robot.

Note that these two prioritization problems differ in the following ways: 1) the number and the types of prioritization objectives are different; 2) the system in the first problem was from Cisco with 211 test cases, whereas in the second problem, there were three case studies, two from Cisco (60 and 624 test cases for different systems than the first problem), and one from ABB Robotics with 89 test cases.

Rule Mining and Configuration Generation: Configuration generation aims to generate a set of configurations for a set of communicating VCSs under which some or all of these VCSs may fail to interact/communicate among each other and avoid configurations that lead to successful interactions among products [35]. To this end, we aim to also learn the rules across VCSs from different families to facilitate configuration generation. For this problem, we formulated a multi-objective configuration generation problem and applied rule mining techniques to mine the rules [35]. To achieve this goal, we defined three objectives based on the rules with normal and abnormal system states. Note that the system states indicate if the VCSs can communicate successfully. The initial set

[5]Available online at https://bitbucket.org/HelgeS/atcs-data

of rules were mined based on randomly generated labeled configurations. The case study used for the evaluation has two VCSs, 17 configuration parameters, 30 rules, and 200 initial configurations. More details can be found in [35].

4.2.2 Real World Case Studies. Requirements Allocation for Inspection: Subsea production systems are large-scale systems that manage the exploitation of oil and gas production fields by integrating hundreds of hardware components and software [7]. At the early phase of developing these systems, a large number of requirements need to be inspected by different stakeholders from different organizations or departments of the same organization. These requirements have different characteristics such as degree of importance, complexity, and dependencies among each other, thereby requiring different effort (workload) to inspect [45]. To this end, we defined a requirements allocation problem that aims to maximize stakeholders' familiarities to the assigned requirements and, at the same time, balance the overall workload of each stakeholder. Our case study consists of 287 requirements and 10 stakeholders. More details can be found in [45].

Test Case Selection: This problem was identified as part of the project with our industrial partner from the maritime domain focusing on developing a new test case execution system for optimizing robustness test execution of our industrial partner's real-time embedded systems deployed in various maritime applications, e.g., dynamic positioning, vessel control, and integrated process control. We observed that it is practically infeasible to execute all the test cases within a limited time budget (e.g., 20 hours) in their context [30]. Thus, it required an efficient approach for cost-effective test case selection, i.e., choose a subset of test cases from the existing test suite within a time budget while achieving pre-defined objectives. To address test case selection problem, we created a real-world case study in [30] to test some of the key components of their systems using different documents (e.g., standards for subsea production systems, requirements from different oil and gas companies publicly available). Our case study consists of 165 high-level test cases and four objectives identified from [30], where more details can be consulted.

4.2.3 Open Source Case Studies. Testing Resource Allocation: This problem is aimed at allocating the resources to different modules (comprising the software system) optimally for maximizing the reliability of the system while minimizing the testing resources (e.g., testing hours) [44]. In [44], a system with eight modules and maximum testing resource of 10,000 hours was used. This problem had three optimization objectives.

Integration and Test Order: This problem aims to determine an order to integrate and test the units (e.g., classes) to minimize the cost for stubbing, where a stub can be defined as an emulation of a unit that has not yet been developed or integrated into the software [17]. An open source program Commons Byte Code Engineering Library (BCEL[6]) version 5.0 was used in [17]. BCEL enables users to create, manipulate, and analyze binary Java class files, and it includes 45 classes with 289 dependencies. For this problem, four objectives were defined in [17].

Software Release Planning: This problem aims to optimally allocate requirements to a set of releases during incremental software development by considering the user preferences [9]. A software with 50 requirements, 5 releases, and 4 clients was used in [9] after identifying the three optimization objectives defined in [9].

4.3 MOSAs and Quality Indicators

We selected the six commonly used MOSAs in SBSE: NSGA-II [11], MoCell [29], SPEA2 [48], PAES [20], SMPSO [28], and CellDE [13]. CellDE was not applicable in the rule mining problem since it requires Integer type solution encoding, whereas CellDE applies to Real/RealArray type

[6]http://archive.apache.org/dist/jakarta/bcel/old/v5.0/

Table 3. Sample 3x3 Contingency Table QI2 QI1

		QI1		
		A	B	ND
QI2	A	v_{11}	v_{12}	v_{13}
	B	v_{21}	v_{22}	v_{23}
	ND	v_{31}	v_{32}	v_{33}

solutions [12]. We selected the following eight commonly used QIs in SBSE covering various quality aspects described in Sect. 2: GD, ED, EP, GS, PFS, IGD, HV, and C.

4.4 Experiment Execution, Evaluation Metrics, and Statistical Tests

We re-ran experiments with all the 11 search problems to ensure that we use all the selected MOSAs with the same parameter settings. Each MOSA was run 100 times to account for random variation. Note that our results are based on all the Pareto-front solutions across all the 100 runs. To ensure a fair comparison among the MOSAs, we used the same settings for all the algorithms (e.g., population size, termination criterion, fitness function), the same evaluation metrics and statistical tests based on well-established guidelines [4]. Moreover, the parameter settings of the MOSAs were taken from our previous published papers (see references in Table 2), where these settings have provided good results. Further details are provided below:

Based on the results of the experiments, we calculated all the eight QIs to compare the selected MOSAs (A1 in Fig. 1) based on all the Pareto-front solutions across all the 100 runs. Note that the QI values for different QIs are interpreted in different ways. For example, for HV, a higher value indicates a better performance of a MOSA, whereas for GD, a lower value indicates a better performance of MOSA. Followed by this, in the A2 activity, we used the *Difference Tests* (DT), i.e., the *Mann-Whitney U* test to determine the significance of differences in the results, and the *Vargha and Delaney A12* statistics as the effect size measure to determine the strength of the significance, as recommended in [4]. When comparing two MOSAs, i.e., A and B, with respect to a QI (e.g., *HV*), (1) If a p-value computed by the U test was < 0.05, and the $A12$ value was > 0.5, it means that A is significantly better than B regarding the QI; (2) If a p-value computed by the U test was < 0.05, and the $A12$ was < 0.5, it means that B is significantly better than A regarding the QI; and (3) If a p-value was ≥ 0.05, it means that there is *No Significant* (*ND*) difference between A and B regarding the QI.[7] Hence, we form 3x3 contingency tables (A3 in Fig. 1) for each QI pair (e.g., QI1 and QI2, Table 3).

In Table 3, the v_{ij} values count the nine cases, i.e., the combinations of *A*, *B*, and *ND* for QI1 and QI2. For example, v_{11} in Table 3 presents the number of times when both QI1 and QI2 agree that A is significantly better than B. Note that we chose such contingency tables, since they are more precise than contingency tables with only two categories, i.e., *Agree* (i.e., cells *A-A*, *B-B* and *ND-ND*) or *Not Agree* (i.e., all the other cells in the table). However, when studying each pair of QIs across search problems (RQ2.2), MOSAs (RQ3.1b), and pairs of MOSAs (RQ3.2), the sample size based on 3x3 contingency tables was small, and the *Bowker* test was not applicable. Thus, we used an additional metric called *Agree* to perform an alternative test. *Agree* is a Boolean variable: *True* when two QIs agree, *False* otherwise. Thus, we formed contingency tables of 11x2 (RQ2.2), 6x2 (for RQ3.1b), and 15x2 (RQ3.2), where 11 is the number of search problems, 6 is the number of MOSAs, and 15 is the number of pairs of MOSAs.

[7]Note that $A12$=0.5 guarantees that p-value of the U test is 1.

To answer the RQs (activities A4 to A8 in Fig. 1), the contingency tables were then inputted to the statistical tests shown in Fig. 1. In summary, we used three types of tests: (1) Three *Agreement Tests* (*AGT*), i.e., *Overall Agreement* (*OA*), *Kappa*, and *Bowker* test, as it is suggested to use more than one agreement test to ensure reliable conclusions [8]. Note that (*OA*) and *Kappa* provide degrees of agreements and do not tell the statistical significance of the agreements. Due to this reason, we also selected the *Bowker* test; (2) Three *Difference Tests* (*DT*), i.e., *Kruskal-Wallis*, *Mann-Whitney U*, *Vargha and Delaney A12*; and (3) One *Association Test* (*AST*): Extension of the *Fisher Exact* test for *n*x2 contingency tables [16]. For all the statistical tests, we chose the significance level of 0.05.

The statistical tests for each RQ are described below.

4.4.1 RQ1. RQ1 studies QI pairs across all the search problems and MOSAs together. We used the metrics below.

Overall Agreement (*OA*): it is calculated as $\frac{\sum_{i=1}^{3} v_{ii}}{\sum_{i=1}^{3}\sum_{j=1}^{3} v_{ij}}$. The numerator is the sum of v_{11}, v_{22} and v_{33} in a contingency table since these values represent the number of times that two QIs agree to each other (*A-A*, *B-B*, *ND-ND*). The denominator represents the total number of comparisons for each pair of QIs, i.e., *#search problems* × *#MOSA pairs*. In our case, it is calculated as 11*15=165. However, since for the rule mining problem, CellDE was not applicable (Sect. 4.3), the total number of comparisons is hence 160.

Kappa: Cohen's *kappa* coefficient (κ) measures the agreement of two raters (QIs in our case) for the categorical data. κ is applicable in our context as we have three categories (*A*, *B*, and *ND*) for each QI in the contingency tables. κ is more robust than simple percentages (i.e., OA) [8]. κ produces a value from -1 to 1. A value of 0 means that the agreement is by chance, whereas 1 represents the perfect agreement and -1 represents perfect disagreement.

Bowker test: It is an extension of the *McNemar* test [2], which is only applicable for 2x2 contingency tables. We applied the *Bowker* test to our 3x3 contingency tables. A p-value of < 0.05 tells that two QIs have significant disagreement.

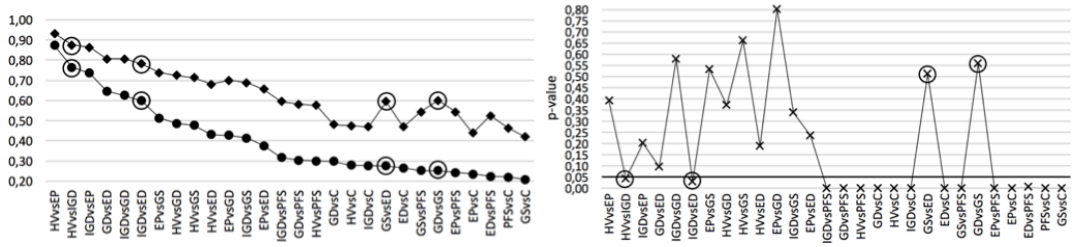
4.4.2 RQ2 and RQ3. For RQ2.1, we studied the QI agreements by search problem ignoring the differences of the MOSAs. Thus, we used the *OA* and *Kappa* values for each search problem for analyses. Similarly, in RQ3.1a, we studied the QI agreements by MOSA ignoring the differences in the search problems. Thus, we used *OA* and *Kappa* values for each MOSA. For RQ2.1, the sample size per search problem was 15 (i.e., the total pairs of MOSAs, C(6,2)) except for RM, which had the sample size of 10 since CellDE is not applicable. For RQ3.1a, the sample size per MOSA is as follows: (1) for CellDE, (*#search problems*-1)×(*#MOSAs*-1), i.e., (10×5=50) as CellDE was not applicable for RM and, therefore, the number of search problems is 10 (not 11); and (2) for the other MOSAs, the sample size is *#search problems*×*#MOSAs*-1, i.e., 11×5-1=54. Note that one is deducted since, for RM, CellDE is not applicable.

For RQ2.1 and RQ3.1a, we performed the *Kruskal-Wallis* test to determine if there are overall differences among the search problems (RQ2.1) or MOSAs (RQ3.1a) regarding the agreement. If the obtained p-value was < 0.05, we further performed the *Mann-Whitney U* test to compare each pair of MOSAs (RQ3.1a). We also used the *Vargha and Delaney's A12* statistics as the effect size measure to further discuss *Mann-Whitney U test*'s results. Note that we didn't apply the *Bowker* test in RQ2.1 since in most cases the data violated a key requirement of the test, i.e., the sum in any two symmetric cells in the contingency tables must be either 0 or should be at least 10 [2]. However, the *Bowker* test was applicable for the single MOSA (RQ3.1a).

For RQ2.2, RQ3.1b, and RQ3.2, we used *OA* and *Kappa* to compare QI pairs across search problems, MOSA, and pairs of MOSAs. We also used the *Kruskal-Wallis* test to determine if there exist significant differences among all the QI pairs regarding *OA* and *Kappa*. Moreover, to study QI

Table 4. Parameter settings for the selected MOSAs

Parameter	Settings
Population Size	100 for All but PAES; PAES Archive Size: 100
Neighborhood	MoCell & 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



(a) *OA* (diamonds) and *Kappa* (dots) (b) *Bowker* test

Fig. 2. Results of RQ1

pairs across the search problems, MOSAs, and pairs of MOSAs, regarding *Agree*, we performed an extension of the *Fisher exact* test for *n*x2 contingency tables [16], to study the differences between each QI pair.

4.5 Parameter Settings

We used the implementation of the QIs and MOSAs, and the default parameter settings of the MOSAs all from jMetal [12]. The parameter settings of the MOSAs are summarized in Table 4. Note that we chose these settings since they have shown good results for the studied problems in the experiments published in our previous works (see Table 2 for the references). An alternative would have been to tune MOSAs for each problem; however, this may result into different parameter settings for different search problems. Since we were also studying the QI agreements across different problems, we decided to keep these settings the same such that we can study the agreements without the influence of the parameter settings. Studying how the various parameter settings of MOSAs affect the agreements among the QIs is an interesting direction to follow. However, such a study requires an entirely new experiment with a carefully planned design, varying various parameter settings of MOSAs systematically. Both the original (see Table 2 for the references) and the repeated experiments reported in this paper compared the MOSAs with Random Search (RS). Results showed that all the MOSAs performed significantly better than RS and thus, we didn't include these results.

5 RESULTS AND ANALYSES

5.1 Results of RQ1

Recall from Sect. 4.1 that RQ1 focuses on studying the agreement relationships among QI pairs while ignoring the differences of MOSAs and SBSE problems. We summarize the results of RQ1 in Fig. 2a and Fig. 2b. In Fig. 2a, the QI pairs are sorted based on the *Kappa* values. In all the cases,

Table 5. Quality Aspects and QI Agreements: RQ1*

Covering QA	Statistical Test Results	#	Covering QA	Statistical Test Results	#
SameQA	Not significant	11	SameQA	Significant	8
DiffQA	Not significant	0	DiffQA	Significant	5

*QA: Quality Aspects; SameQA: Two QIs cover the same QA; DiffQA: Two QIs cover different QA; #: Number of instances

consistent results were obtained for *OA* and *Kappa*; when *OA* showed significant agreement, *Kappa* also showed the same. More specifically, all the *OA* values are above 40%, whereas all the *Kappa* values are positive.

From Fig. 2b (where QI pairs are sorted as in Fig. 2a), one can also notice that in most cases, when *Kappa/OA* gave higher agreements, the *Bowker* test revealed no significant disagreement (i.e., p-values > 0.05). When *Kappa/OA* revealed lower agreements, the *Bowker* test revealed significant differences (p-values < 0.05) suggesting that there are significant disagreements among the compared pair of QIs. Only exceptions were: HV vs. IGD, IGD vs. ED, GS vs. ED, and GD vs. GS (highlighted with small circles in Fig. 2a and Fig. 2b). In the first two pairs, although *Kappa* revealed higher agreements, *Bowker* revealed significant disagreement (p-value < 0.05). For the last two pairs, even though *Kappa* revealed lower agreements, *Bowker* revealed no significant disagreement (i.e., p-values ≥ 0.05). These exceptions can be explained by the fact that *Kappa* and *Bowker* may give different results from each other [8] mainly because Kappa focuses on a general trend, whereas *Bowker* focuses on finding strong disagreements. This further justifies the need of applying multiple agreement statistics to draw conclusions.

Moreover, we looked into the results based on the four quality aspects, i.e., *Convergence*, *Spread*, *Uniformity*, and *Cardinality* (Sect. 2). Note that each QI covers one or more of the quality aspects. The results are summarized in Table 5. Excluding the four exceptions discussed above, we observed three cases. First, we observed that there are no significant differences for 11 (out of 24) pairs of QIs that cover the same quality aspects, such as HV vs. EP and IGD vs. EP. Second, we observed that five pairs of QIs that cover different quality aspects had significant differences based on our results (e.g., GD vs. PFS). Third, we observed an unexpected case, where eight pairs of QIs (e.g., IGD vs. PFS) covering the same quality aspects, but having significant differences. The third case suggests that even though SBSE researchers shall know which QIs cover which quality aspects, it is still insufficient to select QIs only based on quality aspects.

RQ1 can be answered as: when studying the differences in agreements among the QI pairs over all the search problems and MOSAs, there are disagreements as it can be seen from the results of *OA* and *Kappa*. When looking at each pair from the results of the *Bowker* test (Fig. 2b), we can see that some pairs significantly disagree (e.g., GS vs. C, PFS vs. C) and others show no significant disagreement (e.g., HV vs. EP). Moreover, PFS and C disagree significantly with the rest of QIs, and with each other as shown in Fig. 2b (p-values below the line drawn at 0.05) that all the pairs involving PFS and C achieved p-values less than 0.05. Moreover, our assessment based on the four quality aspects suggests that there are pairs of QIs covering the same/different quality aspects that may have significant agreement/disagreement. These results suggest that there are no general agreements among the QIs, and these agreements appear to relate to the search problems and MOSAs that we further study in the next two RQs. This result is useful for SBSE researchers, since it demonstrates that there is no clear answer to which QI to use for their particular context, and there is a need for some guidance. To this aim, we designed a set of guidelines, as explained in Sect. 6.1.

5.2 Results of RQ2

Recall from Sect. 4.1 that RQ2 studies QI agreements with respect to search problems, while ignoring the differences of MOSAs.

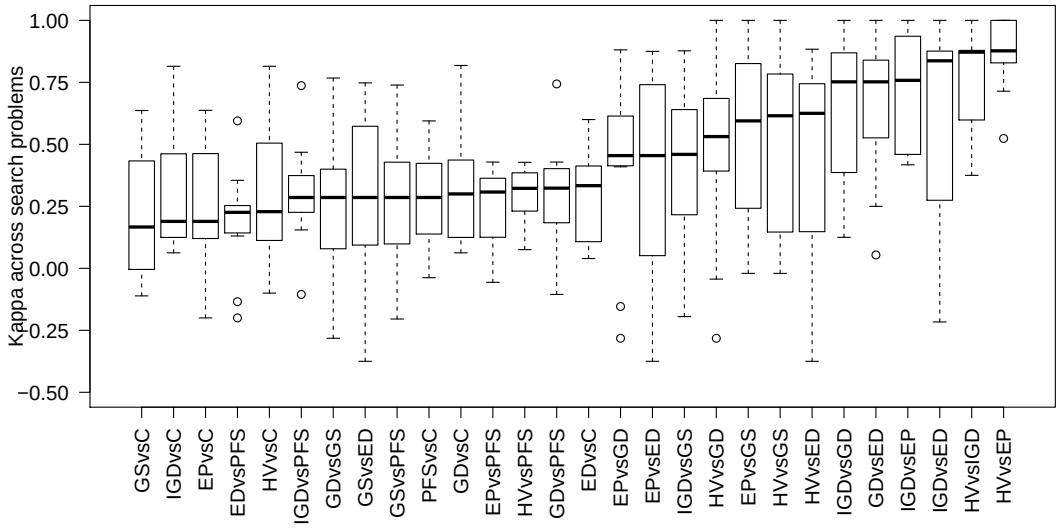
5.2.1 RQ2.1. As we described in Sect. 4.1, RQ2.1 focuses on studying the QI pairs by search problem with *OA* and *Kappa*. Note that the *Bowker* test was not applicable (Sect. 4.4). We tested, with the *Kruskal-Wallis* test, if there are differences regarding *OA* and *Kappa* among the search problems across all the QI pairs. The test revealed p-values < 0.0001 suggesting that there are significant differences among the search problems regarding *OA* and *Kappa*.

RQ2.1 can then be answered as follows: the agreements among QI pairs are strongly related to the search problems. This could be, for instance, due to the specific characteristics of the search problems, such as the number and types of objectives. However, studying such characteristics requires a carefully designed controlled experiment controlling various characteristics of the search problems. To find such industrial/real-world/open-source case studies that systematically cover all the characteristics is extremely difficult. Since our case studies do not cover various characteristics systematically, we only performed a preliminary experiment to assess whether the number of objectives of the search problem has some influence on the agreement. To study such influence, we compared the differences among the different groups (i.e., the search problems with the number of objectives 2, 3, and 4) based on the level of QI agreements. We performed the *Kruskal-Wallis* test across the problems having 2, 3, and 4 objectives (see the details on the problems in Table 2). We observed that there are no significant differences across the problems based on the number of objectives. From the results of this preliminary study, it seems that the number of objectives may not be the factor that influences agreement/disagreement. However, due to the limited number of the problems (e.g., the three subsets based on the number of objectives, see Table 2) and the absence of problems with a higher number of objectives (i.e., greater than four, see Table 2), we cannot draw any concrete conclusions. As a future work, we plan to conduct more experiments by varying (with control) the characteristics of the problems and check if they put any influence on the agreements of QIs, by devising synthetic problems.

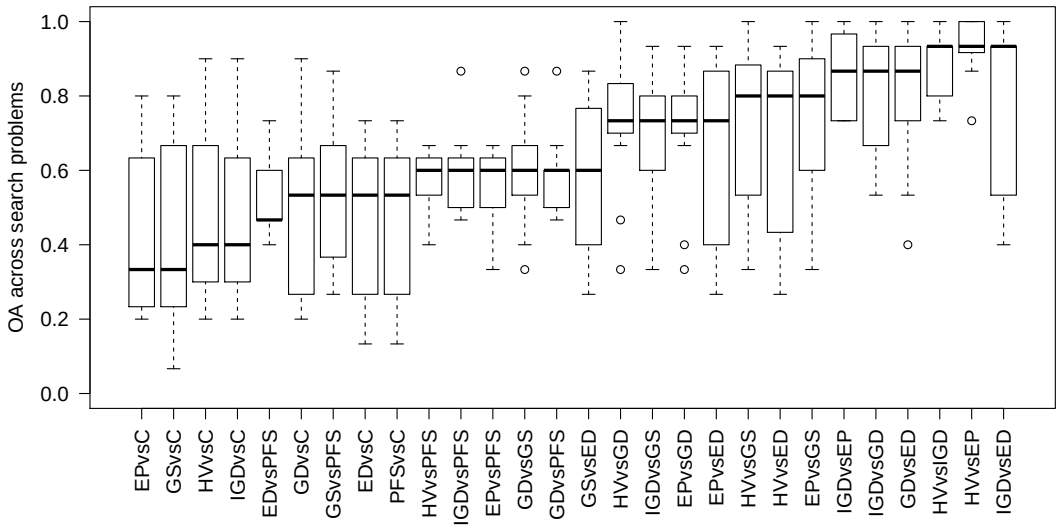
5.2.2 RQ2.2. In contrast to RQ2.1 where we study all the QI pairs together, RQ2.2 studies individual QI pairs across the search problems regarding *OA*, *Kappa*, and *Agree*. Recall from Sect. 4.4 that the *Bowker* test was not applicable for this RQ due to the small sample size. Therefore, we introduce the *Agree* metric for performing an alternative test.

Fig. 3a shows the boxplots of the distributions of the *Kappa* values for each QI pair across the search problems. We can observe that there seem to be differences among the QI pairs. We further checked such differences with the *Kruskal-Wallis* test. Results revealed a p-value $<$ of 0.001, suggesting that there are significant differences among the QI pairs across the search problems regarding *Kappa*. Similarly, Fig. 3b shows the boxplots for comparison of each QI pair for the *OA* values. We observed similar results as for *Kappa*, where the Kruskal-Wallis test revealed a p-value of < 0.001, suggesting the significant differences between each QI pair for *OA*. Thus, we can conclude that there are significant differences between each QI pair regarding *OA* and *Kappa* across the search problems.

We further studied if there exist significant differences in the agreements for each QI pair across the search problems with the *Agree* metric (Sect. 4.4) using the *Fisher exact* test. The results are summarized in Table 6. Out of the 28 pairs compared, 9 pairs had no significant differences in agreements, whereas the other 19 had significant differences. We also studied these results with respect to the four quality aspects (see Table 1) that are measured by the QIs. Table 7 summarizes the results. Among the pairs of QIs showing no significant differences, six pairs of QIs measure the



(a) Kappa



(b) Overall agreement (OA)

Fig. 3. Boxplots for each pair of QIs across search problems (RQ2.2). (For *Kappa*, a value of 0 means that the agreement is by chance, where 1 represents the perfect agreement and -1 represents perfect disagreement. For *OA*, the range is from 0-1 and a higher value means a higher agreement. All the QI pairs on the left boxplots were ordered with median *Kappa* values and the right boxplots were ordered with median *OA* values. Note that one data point corresponds to the agreement measured as Kappa/OA between a QI pair for a given search problem across all the MOSAs for this search problem)

same quality aspects (e.g., HV vs. IGD). There were four pairs of QIs (e.g., GS vs. ED) measuring different quality aspects, that showed significant differences. Moreover, we observed the following two exceptional cases: 1) three pairs of QIs (e.g., GD vs. GS) measuring different quality aspects

Table 6. Results of the *Fisher exact* test for each QI pair across the search problems (RQ2.2).*

QI Pair	Result	QI Pair	Result	QI Pair	Result	QI Pair	Result
HVvsIGD	NS	IGDvsEP	NS	EPvsGS	S	GDvsC	S
HVvsEP	NS	IGDvsGD	S	EPvsED	S	GSvsED	S
HVvsGD	S	IGDvsGS	S	EPvsPFS	NS	GSvsPFS	S
HVvsGS	S	IGDvsED	S	EPvsC	S	GSvsC	S
HVvsED	S	IGDvsPFS	NS	GDvsGS	NS	EDvsPFS	NS
HVvsPFS	NS	IGDvsC	S	GDvsED	S	EDvsC	S
HVvsC	S	EPvsGD	S	GDvsPFS	NS	PFSvsC	S

* S: Significant difference; NS: No significant difference.

Table 7. Quality Aspects and QI Agreements (RQ2.2).

CQA	Statistical Test Results	#	CQA	Statistical Test Results	#
SameQA	Not significant	6	SameQA	Significant	15
DiffQA	Not significant	3	DiffQA	Significant	4

*CQA: Covering Quality Aspects; SameQA: Two QIs cover the same QA; DiffQA: Two QIs cover different QA; #: Number of instances

showed no significant differences; and 2) 15 pairs of QIs (e.g., HV vs. GS) measuring the same quality aspects showed significant differences. These unexpected cases confirm our results from RQ1 that only selecting the QIs based on the quality aspects that they measure is not sufficient.

Based on the above results, we can answer RQ2.2 as follows: the characteristics of the search problems make most of the QI pairs disagree significantly. Furthermore, even the QIs that measure the same quality aspects do not necessarily agree. Similar to the conclusions of RQ2.1, we need to further systematically study various characteristics of the problems to derive more concrete conclusions with respect to various characteristics such as the number and types of objectives.

5.3 Results of RQ3

Recall from Sect. 4.1 that RQ3 studies the QI agreements with respect to MOSAs, while ignoring the differences of the search problems. We present the results of RQ3.1 (a-b) and RQ3.2, focusing on one MOSA at a time and on pairs of MOSAs, respectively. Note that it is possible that a given MOSA may be preferred by certain QI(s) due to the inherent working mechanisms of the MOSA. However, this aspect wasn't the focus of our empirical evaluation, since we were only interested in studying the agreements among the QIs.

5.3.1 RQ3.1a. We studied the differences in the QI agreements among all the MOSAs using *OA*, *Kappa*, and *Bowker*. Agreements for individual QI pairs are, however, studied in RQ3.1b. We performed the *Kruskal-Wallis* test to check if there are differences among the MOSAs regarding *OA* and *Kappa*. For *OA*, we observed a p-value of 0.2, i.e., there were no significant differences among the six MOSAs regarding *OA*. However, for *Kappa*, we found a p-value of 0.009 suggesting that there are significant differences among the MOSAs. Thus, we performed a pair-wise comparison among the studied MOSAs with the *Mann-Whitney U* test, and *A12* for *Kappa*; the results are summarized in Table 8. In Table 8, we provide the number of times one MOSA had significantly better/worst Kappa values than the rest of the MOSAs. We do not compare the quality of solutions produced by

Table 8. Summarized results for the *Mann-Whitney U* Test and *A12* Statistics across each pair of MOSAs for Kappa (RQ3.1a)*.

MOSA	<	=	>
CellDE	0	4	1
MoCell	0	4	1
NSGA-II	0	3	2
PAES	0	4	1
SMPSO	4	1	0
SPEA2	1	4	0

*Column "<" (">") counts the number of times that a MOSA had significantly lower (higher) *Kappa* values than the other five MOSAs (i.e., p-values < 0.05 and A12 < 0.5). Column "=" counts the number of times when there were no significant differences (i.e., p-value ≥ 0.05).

Table 9. Summarized results for the *Bowker* test for each QI pair (RQ3.1a)*.

QI Pair	S%	QI Pair	S%	QI Pair	S%	QI Pair	S%
HVvsIGD	0	IGDvsEP	0	EPvsGS	17	GDvsC	100
HVvsEP	0	IGDvsGD	0	EPvsED	0	GSvsED	17
HVvsGD	0	IGDvsGS	0	EPvsPFS	67	GSvsPFS	83
HVvsGS	17	IGDvsED	0	EPvsC	100	GSvsC	100
HVvsED	0	IGDvsPFS	33	GDvsGS	17	EDvsPFS	17
HVvsPFS	33	IGDvsC	100	GDvsED	25	EDvsC	100
HVvsC	100	EPvsGD	0	GDvsPFS	67	PFSvsC	80

*S%: ((x/y)×100), where x is the number of MOSAs (out of 6) for which a QI pair had a p-value < 0.05, i.e., the pair significantly disagrees. y is the number of MOSAs (out of 6) for which the Bowker test gave a p-value; indeed, note that, in some cases, the Bowker test couldn't be performed (Sect. 4.4.2.)

the MOSAs. From Table 8, we can observe that there weren't many differences among the MOSAs (the results presented in the "=" column). The only exception is SMPSO, which was worse than the four MOSAs (the "<" column), and there were no significant differences for one MOSA (the "=" column). This suggests that SMPSO has the lowest agreement among the QI pairs when compared with the rest of the MOSAs. The reason may be due to the fact that SMPSO produces solutions whose quality is represented by different QIs. However, we need to further investigate this with more focused experiments.

We further summarize the results of the *Bowker* test for each QI pair for all the MOSAs in Table 9. S% for a QI pair based on the Bowker test is calculated as:

$$S\% = \frac{\text{\# MOSAs for which QI1 \& QI2 significantly disagreed}}{\text{\# MOSAs for which the } \textit{Bowker} \text{ test was applicable}}$$

The numerator is the number of MOSAs for which a p-value for the *Bowker* test was < 0.05 for a QI pair. The denominator is the number of MOSAs for which the *Bowker* test was applicable (see Sect. 4.4.2). From the table, we can see that most of the times (18 out of 28), S% was either 0% or at the most 33%. The remaining 10 pairs had either PFS or C, suggesting that PFS and C are significantly different from each other and from the other QIs. Note that this was also concluded

Table 10. Quality Aspects and QI Agreements (RQ3.1a)*.

CQA	Condition	#	CQA	Condition	#
SameQA	S%=0	10	SameQA	S%>0 and S%<100	6
SameQA	S%=100	5	DiffQA	S%=0	0
DiffQA	S%>0 and S%<100	6	DiffQA	S%=100	1

*CQA: Covering Quality Aspects (CQA); SameQA: Two QIs cover the same quality aspects; DiffQA: Two QIs cover the different quality aspects; Condition: Constraints on the S% values from Table 9; #: Number of instances

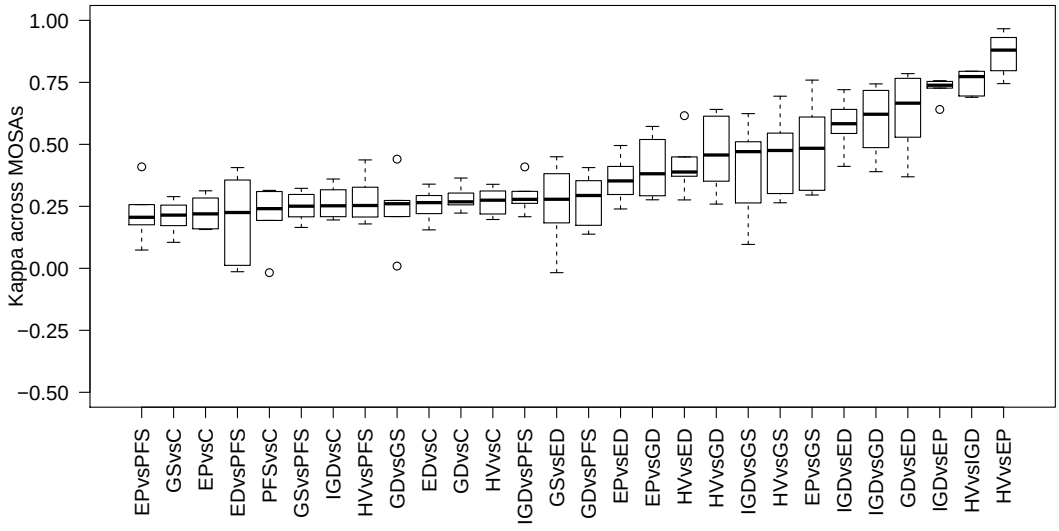
from the results of RQ1 (Sect. 5.1). Thus, we can conclude that the QI agreements between the QI pairs do not vary much across the MOSAs except for the pairs involving PFS and C.

We also investigate these results with respect to the quality aspects described in Table 1. The results are summarized in Table 10. We observed the following five cases. First, there were 10 pairs of QIs (e.g., HV vs. IGD) that measure the same quality aspects with S% of 0%, i.e., no significant differences. Second, six pairs of QIs (e.g., HV vs. GS) measured the same quality aspects with S% greater than 0% and less than 100%, e.g., 17% for HV vs. GS. Third, there were five pairs of QIs (e.g., HV vs. C) that measured the same quality aspects with S% of 100%, i.e., all significantly different. Fourth, for the pair GS vs. C, the S% is 100%. GS and C measure different quality indicators; therefore, it makes sense that S% was 100%. Fifth, among the QIs pairs for which S% was less 100% and greater than 0%, six measured different quality aspects. Once again, similarly to results from RQ1 and RQ2, we can say that selecting the QIs solely based on the quality aspects they measure is not advisable.

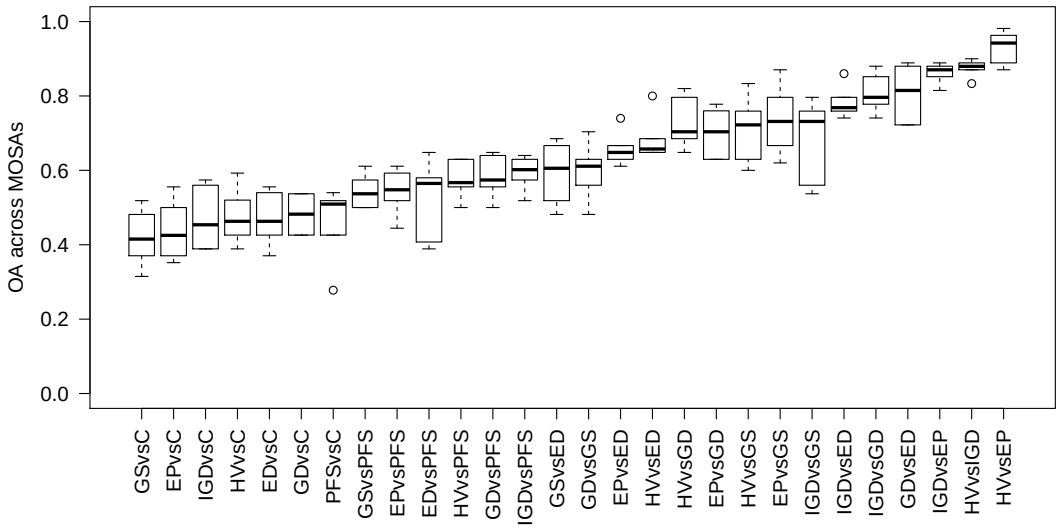
Based on the above results, we can answer RQ3.1a as follows: 1) the agreement (as *OA*) among the QI pairs does not vary across the MOSAs; 2) the agreement (as *Kappa*) does not vary much across the MOSAs, except for SMPSO; 3) based on the results of the *Bowker* test, agreements among the QI pairs do not vary much across the MOSAs except for the pairs that had PFS and/or C; and 4) the agreements between a pair of QIs may not necessarily correspond to the quality aspect measured by the pair.

At a high level, one may think that the differences between some of the pairs (e.g., the PFS and C) seem to be natural because of their inherent characteristics, e.g., due to the quality aspects they measure. However, merely suggesting SBSE researchers to consider using both PFS and C to have a diverse view on the quality of the solutions may not be sufficient. As our results showed, even the QIs measuring the same quality aspects may not agree with each other. We also found these results useful for the SBSE researchers, since the results convey that the MOSAs do not impact the agreements among the QIs. However, note that in this experiment, we didn't vary the parameter settings of the MOSAs. These fixed settings were chosen based on our previous papers, where these settings demonstrated good results. The variations of parameter settings may potentially impact the agreements, but it requires investigation of its own.

5.3.2 RQ3.1b. This RQ studies the agreement between each QI pair across the MOSAs regarding the *OA*, *Kappa*, and *Agree* metrics. Note that this RQ studies each pair of QIs, whereas RQ3.1a studies all the QI pairs at the same time. Recall from Sect. 4.4 that the *Bowker* test was not applicable in this RQ due to the small sample size. We show the boxplots for the distributions of each QI pair regarding *Kappa* and *OA* in Fig. 4 when studying each MOSA. We can observe that there seem to be differences among the QI pairs. We further checked such differences with the *Kruskal-Wallis* test. Results revealed p-values < 0.0001 for both *Kappa* and *OA*, suggesting that there are significant differences among the QIs regarding *Kappa* and *OA*. We further studied whether there



(a) Kappa



(b) Overall agreement (OA)

Fig. 4. Boxplots for each pair of QIs across MOSAs (RQ3.1b)

exist significant differences for each QI pair regarding *Agree* across MOSAs with the *Fisher exact* test. There were no significant differences, except for IGD vs. GS, EP vs. GS, and ED vs. PFS. The first two pairs of QIs (i.e., IGD vs. GS, and EP vs. GS) measure the same quality aspects (see Table 1), but still, our results showed significant differences. The third pair of the QIs (i.e., ED vs. PFS) measure different quality aspects (see Table 1), and thus it is consistent with our results, i.e., significant differences. For the rest of the QI pairs that showed no significant differences, ideally, the pairs shall cover the same quality aspects (see Table 1); however, for six QI pairs (e.g., GD vs. GS), it

wasn't true. These results once again confirm that the agreements among the QI pairs may not relate to the quality aspects measured by the QIs.

Based on the above results, we can answer RQ3.1b as follows: 1) There are significant differences among the QI pairs regarding *OA* and *Kappa* across the MOSAs; 2) QI pairs do not differ across the MOSAs based on *Agree*; and 3) the agreement between a pair of QIs measured with *Agree* does not necessarily correspond to the quality aspects measured by the pair. These results are useful for the SBSE researchers who are planning to study one specific MOSA for their experiments. Moreover, we used the results of this RQ to develop the guidelines to select QIs for one specific MOSA presented in Sect. 6.3.

5.3.3 RQ3.2. We study the agreements between each QI pair regarding *OA*, *Kappa*, and *Agree* across the pairs of MOSAs. Note that in RQ3.1, we studied one MOSA, whereas in RQ3.2, we study the pairs of MOSAs. Fig. 5 shows the boxplots for distributions of *Kappa* and *OA* values for each QI pair when studying pairs of MOSAs. We can observe that there seem to be differences among QI pairs regarding *Kappa* and *OA*. We further checked such differences with the *Kruskal-Wallis* test. Results revealed p-values < 0.001, suggesting that there are significant differences among the QI pairs. We further studied if there exist significant differences regarding the agreement in terms of *Agree* for each QI pair across the pairs of MOSAs with the *Fisher exact* test. Except for HV vs. PFS, IGD vs. GS, EP vs. GS, and EP vs. PFS, there were no significant differences among the other pairs. Each of the four pairs of QIs that showed significant differences measure the same quality aspects (see Table 1). These results indicate that when comparing a pair of MOSAs, a pair of QIs that measure the same quality aspects can significantly disagree. Seven out of the rest of the pairs that do not have significant differences (24 after excluding the four pairs from the total, i.e., 28 pairs) measured the different quality aspects. Once again, these results tell us to not rely only on the quality aspects when comparing a pair of MOSAs for selecting QIs.

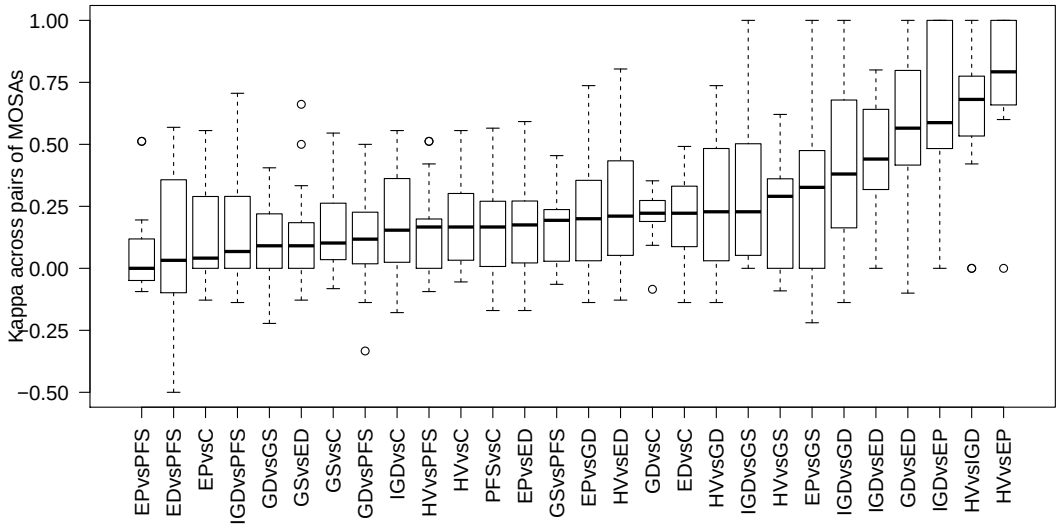
Based on the above results, we can answer RQ3.2 as: These results are useful for the SBSE researchers who are planning to use a pair of MOSAs for their experiments. Besides, the results from this RQ helped us to devise the guidelines to select QIs for pairs of MOSAs that will be presented in Sect. 6.3.

6 ADDITIONAL ANALYSES FOR GENERATING OBSERVATIONS AND GUIDELINES

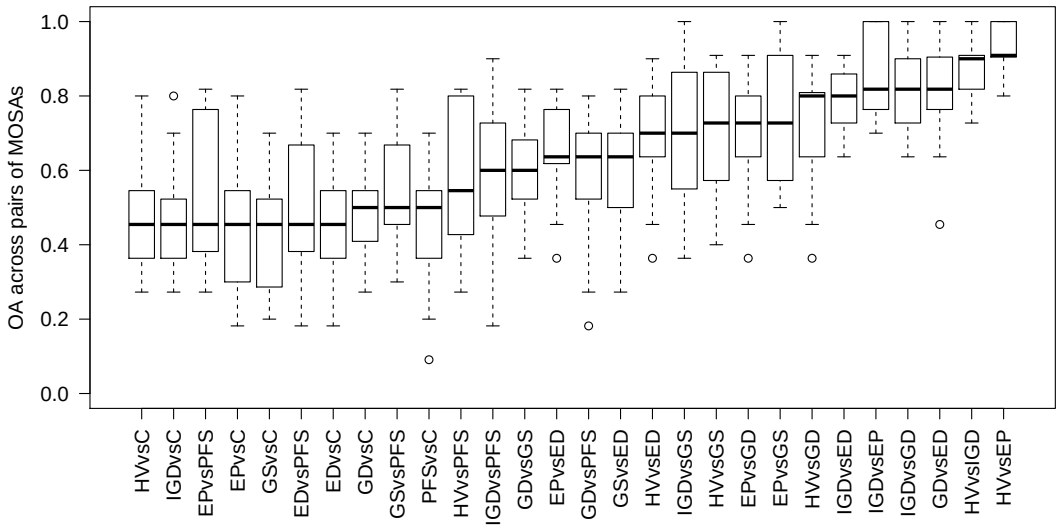
When applying MOSAs, one needs to select the best MOSA(s) among the applied ones using the QIs. Given that SBSE researchers mainly apply search algorithms from the domain of evolutionary computation, they often have limited knowledge about which QIs to select. One typically ends up with selecting the most commonly used QIs, e.g., HV [5, 26, 39]. To this end, we aim to provide a set of guidelines to help SBSE researchers to select QIs based on evidence.

Based on our empirical evaluations (Sect. 5), in this section, we provide some key observations (Sect. 6.2) obtained using an automated process (Sect. 6.1), followed by the definition of a set of guidelines (Sect. 6.3) that may help SBSE researchers in selecting the QIs for their particular contexts. First, we would like to highlight that our observations are based on two assumptions:

(1) An SBSE researcher wants to use one QI, i.e., *Single Representative* (*SR*), or the *smallest subset* of the QIs, i.e., *Minimum Representative Set* (*MRS*). In the former case, s/he wants to select a QI that has the highest agreement with the rest of the QIs, i.e., *SR* is the *representative* of the majority of the QIs. In the latter case, s/he wants to select the smallest set of the QIs that is *representative* of all the QIs (i.e., *MRS*); and
(2) An SBSE researcher is interested in selecting one or more QIs: (1) independent of any MOSA or SBSE problem characteristics (*All*); (2) caring about one specific MOSA (*SingleAlgorithm*); or (3) caring about a given pair of MOSAs (*PairAlgorithms*). These three types of observations



(a) Kappa



(b) Overall agreement

Fig. 5. Boxplots for each pair of QIs across pairs of MOSAs (RQ3.2)

were devised based on three ways in which SBSE researchers want to compare MOSAs. Regarding *All*, researchers are interested in comparing all of the selected MOSAs with each other (e.g., in [24]). In the *SingleAlgorithm* context, SBSE researchers are interested in comparing a newly proposed MOSA (e.g., a variation of NSGA-II [39]) with other MOSAs, but not with each pair of the MOSAs. Regarding *PairAlgorithms*, SBSE researchers are interested in comparing a pair of MOSAs (e.g., NSGA-II vs. MoCell in [14]). We acknowledge that there

are many other ways (e.g., comparing 3 MOSAs) and indeed more empirical studies are needed to extend our observations to a wider scope.

6.1 Automated Generation of Observations

Based on the type of observations, i.e., *All*, *SingleAlgorithm*, and *PairAlgorithms*, we select data from RQ1, RQ3.1, and RQ3.2, respectively. For *All*, for each QI pair (in total 28 pairs), we have *Kappa* values and *Bowker* test's p-values. Similarly, for *PairAlgorithms*, for each MOSA, we have 28 *Kappa* values and *Bowker* p-values. For *PairAlgorithms*, we only have 28 *Kappa* values, one for each QI pair.

Starting from the above data, we build the observations using relation *ag* saying whether two QIs agree. For *All* and *SingleAlgorithms*, it is defined in terms of the *Bowker* and *Kappa* tests as follows:

$$ag(QI_1, QI_2) \equiv Bowker(QI_1, QI_2) \geq 0.05 \wedge Kappa(QI_1, QI_2) \geq 0.4$$

For *PairAlgorithms*, it only depends on *Kappa* and is defined as:

$$ag(QI_1, QI_2) \equiv Kappa(QI_1, QI_2) \geq 0.4$$

Recall that *Bowker* p-value ≥ 0.05 means that there is no significant disagreement between a pair of QIs, whereas *Kappa* ≥ 0.4 is interpreted as at least *Moderate Agreement*[8] [21]. We assume that if two QIs agree, either one can be applied, and thus these two QIs *represent* each other.

Below, we first describe how we use *ag* to select the QI that is representative of the majority (Sect. 6.1.1), i.e., *SR*. Second, we describe how we use *ag* to select the minimal set (i.e., *MRS*) that represents all the QIs (Sect. 6.1.2).

6.1.1 Single QI representing the majority. Given a set of QIs $Q = \{q_1, \ldots, q_p\}$ (where p=8 in our case, i.e., all the QIs), we define the agreement set of q_i (in Q) as:

$$AS(q_i, Q) = \{q_j \in Q \setminus \{q_i\} \mid ag(q_i, q_j)\}$$

We define the *strength of the agreement* as the average *Kappa* value of the pairs of q_i with all the QIs in $AS(q_i, Q)$:

$$agSt(q_i, Q) = \frac{\sum_{q_j \in AS(q_i, Q)} Kappa(q_i, q_j)}{|AS(q_i, Q)|}$$

This function tells *how much* a QI agrees with the others based on *Kappa*. Using the previous definitions, we select the QI with the highest agreement with the others from Q (*SR*) with the following two steps:

a) Select QIs having the largest agreement sets:

$$maxAS(Q) = \arg\max_{q_i \in Q} |AS(q_i, Q)|$$

b) If $|maxAS(Q)| > 1$, select those having the highest *strength of agreement*:

$$maxAgSt(Q) = \arg\max_{q_i \in maxAS(Q)} |agSt(q_i, Q)|$$

In case that there are two or more QIs with the largest agreement sets and the same strength of the agreement, any of them can be picked.

[8] A *Moderate Agreement* is defined as a *Kappa* value between 0.4 and 0.6.

Table 11. Automatically Generated *All*, *SingleAlgorithm*, and *PairAlgorithms* Observations*

MOSA	SR	MRS	MOSA	SR	MRS	MOSA	SR	MRS
All (without caring about any particular MOSA)								
All	EP	C, ED, GS, PFS	-	-	-	-	-	-
Single MOSA (caring about one MOSA)								
CD	IGD	C, EP, GS, PFS	N	IGD	C, IGD, PFS	SM	IGD	C, GS, IGD, PFS
M	IGD	C, IGD, PFS	P	IGD	C, ED, PFS	SP	EP	C, ED, GD, GS, PFS
Pairs of MOSAs (caring about a pair of MOSAs)								
CD vs. M	IGD	EP, PFS	M vs. N	HV	C, ED, GS	N vs. SM	HV	C, EP, GD, GS, PFS
CD vs. N	IGD	ED, GS	M vs. P	IGD	C, EP, GD	N vs. SP	IGD	C, GD, IGD, PFS
CD vs. P	IGD	C, GD, GS, PFS	M vs. SM	EP/HV	C, EP, GD, GS, PFS	P vs. SM	HV	C, EP, GD, PFS
CD vs. SM	IGD	C, GD, GS, PFS	M vs. SP	IGD	C, EP, GD, PFS	P vs. SP	EP/HV	C, ED, EP, GD, GS, PFS
CD vs. SP	IGD	IGD/EP/HV	N vs. P	EP/HV	C, ED, EP, GD, GS, PFS	SM vs. SP	HV	ED, HV, PFS

*CellDE: CD; MoCell: M; NSGA-II: N; PAES: P; SMPSO: SM; SPEA2: SP; Single Representative: SR; Minimum Representative Set: MRS.

6.1.2 Minimal set of QIs representing all the QIs. Given a subset $\widetilde{Q} \subseteq Q$, we define the *agreement set* of $\widetilde{Q}$ as:

$$AS(\widetilde{Q}, Q) = \bigcup_{q_i \in \widetilde{Q}} AS(q_i, Q)$$

The agreement set identifies QIs in Q that agree with some $q_i \in \widetilde{Q}$. We further define the *intra-set agreement* as the average *Kappa* of the pairs of QIs in a set:

$$isetAg(\widetilde{Q}) = \begin{cases} \frac{\sum_{\{q_i, q_j\} \in \widetilde{Q}} Kappa(q_i, q_j)}{C(|\widetilde{Q}|, 2)} & |\widetilde{Q}| > 1 \\ 1 & |\widetilde{Q}| = 1 \end{cases}$$

We aim at selecting a subset of QIs representing all the QIs. To do so, we perform the following three steps:

a) Select the subsets representing all the QIs:

$$allRepr(Q) = \{\widetilde{Q} \in \mathcal{P}(Q) \mid |AS(\widetilde{Q}, Q) \cup \widetilde{Q}| = |Q|\}$$

b) Select the smallest subset from $allRepr(Q)$:

$$minAllRepr(Q) = \underset{\widetilde{Q} \in allRepr(Q)}{\arg\min} |\widetilde{Q}|$$

c) If $|minAllRepr(Q)| > 1$, select the set from $minAllRepr(Q)$ having QIs that are the most different from each other based on *Kappa*:

$$minAllReprWideAg(Q) = \underset{\widetilde{Q} \in minAllRepr(Q)}{\arg\min} isetAg(\widetilde{Q})$$

Hence, we obtain the QIs representing the most different responses (Table 11). Note that it is possible to have two or more such subsets. In this case, any subset can be picked.

All our statistical tests were performed in R[9] scripts, whereas the automated generation of observations was implemented as a Java program that takes the results produced by the R scripts and produces observations.

[9]https://www.r-project.org/

6.2 Observations and Discussion

Table 11 provides our key observation from the two perspectives described in Sect. 6.1. In total, we obtained 22 observations, i.e., 1 for *All*, 6 for *SingleAlgorithm*, and 15 for *PairAlgorithms*. All our observations were automatically devised from the results of the statistical tests (Sect. 5). In particular, *All* is built from RQ1, *SingleAlgorithm* from RQ3.1, and *PairAlgorithms* from RQ3.2. Note that in our experiments we didn't have enough case studies to conduct experiments by controlling characteristics of SBSE problems; hence, we cannot derive problem-specific observations (from RQ2).

As shown in Table 11, for *All*, the single representative QI (i.e., *SR*) is EP, whereas the minimum representative set (i.e., *MRS*) is {C, ED, GS, PFS}. We also show *SR* and *MRS* for a single algorithm in the table. For instance, for NSGA-II, *SR* is IGD, whereas MRS is {C, IGD, PFS}. Also, we show *SR* and *MRS* for pairs of the MOSAs in the table. For instance, when comparing CellDE and MoCell, *SR* is once again IGD, and *MRS* is {EP, PFS}.

From Table 11, we can also observe that IGD is the most agreed QI for *SR* in *SingleAlgorithm*. Indicator IGD is also the most agreed in *SR* for *PairAlgorithms* followed by HV. For *MRS*, indicators PFS and C are present in all the cases for *All* and *SingleAlgorithm*, whereas they are present in most cases for *PairAlgorithms*. This is due to the fact that PFS and C have unique characteristics.

In addition, we also checked if the QI pairs measuring the same quality aspects (see Table 1) agree to each other. We found that 52% of the QIs pairs measuring the same quality aspects do not agree when we studied *SingleAlgorithms*. Similarly, 63% of the QI pairs measuring the same quality aspects do not agree when studying *PairAlgorithms*. This suggests that our agreement-based classification makes sense, as one can not rely just on their category to estimate whether two QIs will give the same evaluation. Moreover, it is interesting to know that the disagreements among the QIs are not necessarily due to the fundamental differences among the QIs. Such disagreements, for instance, might be due to the differences in various characteristics of the search problems (e.g., the number of objectives, or the type of SBSE problem), as we observed in Sect. 4.4.2. To this end, it seems interesting and relevant to study such characteristics of the search problems; however, it requires a carefully designed controlled experiment controlling various characteristics of the search problems. To find such industrial/real-world/open-source case studies that systematically covers all the characteristics is extremely difficult. Another option could be to develop synthetic problems to study agreements/disagreements systematically, but it would require an entirely new experiment.

6.3 Guidelines

Fig. 6 shows the process that an SBSE researcher is recommended to follow when using our generated observations. Recall from Sect. 4.4, that a comparison between two MOSAs A and B with a QI has three responses: A is better than B; B is better than A; and no difference (*ND*). In the process shown in Fig. 6, an SBSE researcher gets two options: selecting one QI (automatically generated for *SR* (Sect. 6.1.1)) or a minimum subset (automatically generated for *MRS* (Sect. 6.1.2)).

When an SBSE researcher wants to apply only one QI, s/he can pick one from one of the *SR* columns in Table 11 based on the *All*/*SingleAlgorithm*/*PairAlgorithms* observations. For example, if one is only interested in CellDE and wants to apply just one QI, s/he can use IGD to select the response.

When an SBSE researcher wants to use a minimum subset of QIs representing all the eight QIs, s/he can pick the subset *MRS* (the *MRS* columns in Table 11). *MRS* is then applied on the results of an SBSE problem for the selected MOSAs. There are two possible outcomes:

(1) If all or the majority of the QIs in *MRS* give the same response (i.e., *A*, *B*, or *ND*), the common response should be taken. For example, in the *All* row in Table 11, *MRS* is {C, ED, GS, PFS}. If

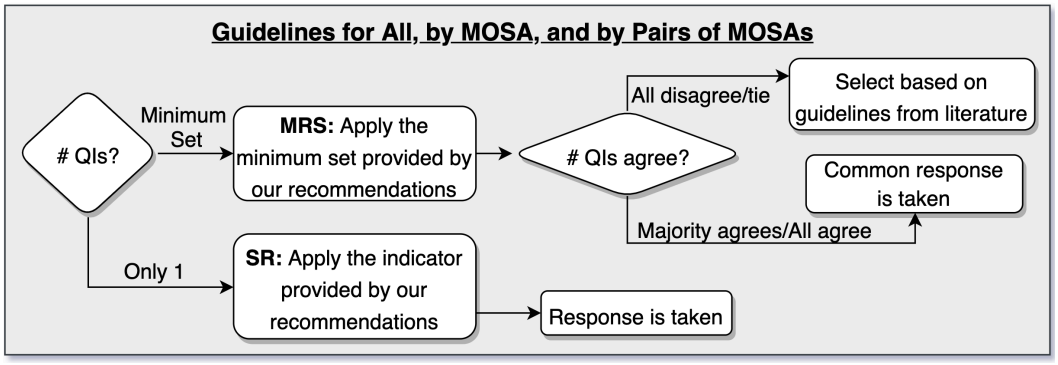


Fig. 6. Guidelines to select QIs

Table 12. Summary of Guidelines from Literature [42]*.

QI	Usage
PFS	A large number of solutions is preferred.
C	IOS* is known, and only one optimal solution is needed.
GS	A wide distribution of solutions is preferred.
GD	IOS is unknown, and the diversity of solutions is needed.
ED	IOS is known.
IGD	1) # objectives > 6, or 2) # objectives ≤ 6, and the reference point is unknown (see Sect. 2)
HV	The reference point is known.
EP	IOS is unknown.

**Ideal Objective Set* (*IOS*) refers to search problems, in which best achievable objective values for all the objectives are known [42]. Note that this summary was taken from [42] by excluding the classifications of QIs

indicators C, ED, and GS consistently tell that MOSA *A* is better, whereas indicator PFS tells that MOSA *B* is better, one should then select MOSA *A* as the champion.

(2) If all QIs in *MRS* disagree or there is a tie, one is recommended to refer to the general guidelines from the literature [42] (summarized in Table 12). For example, for PAES, in Table 11, *MRS* is {C, ED, PFS}. If indicator C says MOSA *A*, indicator ED says MOSA *B*, and indicator PFS tells *ND*, then one is recommended to refer to the literature.

It is interesting to note that all the SRs in Table 11, i.e., EP, IGD, and HV, cover the four quality aspects described in Table 1. This suggests that if one decides to use only one QI, choosing one of the QIs indicated in our guidelines (i.e., SR) can ensure covering the four quality aspects, and also ensures selecting the most representative QI. Our suggested MRSs also ensure the same, if one decides to choose a minimum set of QIs instead of just one QI. In this case, however, we observed the following exceptions. For PAES, the MRS consists of C, ED, PFS and according to Table 1, it doesn't cover the *Spread* and *Uniformity* quality aspects. For CellDE vs. NSGA-II and MoCell vs. NSGA-II, the selected MRSs do not include the *Cardinality* quality aspect.

Our guidelines are useful for SBSE researchers from the following perspectives. First, our guidelines are based on the evidence regarding the agreements among the QIs, which might be useful for SBSE researchers to select the QIs for their particular context. However, there exist at least 100

QIs as surveyed in [23], which makes it very difficult (if not impossible) for SBSE researchers to select the QIs. Second, only selecting the QIs based on the quality aspects is not sufficient as our results showed (see Sect. 5). Our guidelines ensure that an SBSE researcher selects the most diverse QIs. Third, our evidence can help QI designers to understand how the studied QIs agree to each other, e.g., to what extent a QI pair agrees. Fourth, our process can be applied to the existing QIs, in addition to the studied ones, to provide a more comprehensive view of agreements among the existing QIs. Also, given the fact that there are at least 100 QIs [23] and more QIs are appearing, our process based on empirical evaluation can be used to compare, understand, and review the characteristics of the QIs. Fifth, our observations and guidelines can help SBSE researchers to better plan their experiments right from the beginning. For instance, we build our guidelines to select the QIs based on the three common experimental settings (see Sect. 6.1) in which SBSE researchers plan their experiments to compare the MOSAs. Thus, our guidelines can help better planning and selecting QIs during the design of the experiment. Another example would be that if SBSE researchers want to perform an experiment to compare a particular pair of MOSAs, they can select the appropriate QIs based on our guidelines.

7 THREATS TO VALIDITY

One threat to the *internal validity* is the selection of MOSAs. There exist many MOSAs, and we selected the commonly applied ones in SBSE. Another threat is the parameter settings for MOSAs. As for many other SBSE works, we used one default configuration setting, which is, however, based on commonly applied guides [4, 36]. Moreover, these settings gave good results in our previous published papers where we originally solved the studied SBSE problems (see references in Table 2). Our observations are generated based on the results of running MOSAs with the default settings (Sect. 5); therefore, different parameter settings might produce different results and might lead to different observations. However, this requires a completely new experiment. However, our process of automatically generating observations from experimental results are general, i.e., independent of the selected QIs, MOSAs, and tuning of their parameter settings.

To reduce the threat to the *conclusion validity*, we ran each MOSA 100 times to reduce the effect of random variations. Based on the results of the 100 runs, we performed various statistical tests (e.g., the *Mann-Whitney U* test) to answer the targeted research questions. We selected these tests based on an existing SBSE guideline [4].

Regarding threats to the *construct validity*, we carefully selected various measures. First, the measures we defined and used for assessing the QIs (such as *Kappa*) are comparable across the QIs. Second, to be fair, we used the three agreement statistical tests (*OA*, *Kappa*, and *Bowker* test) to assess the QIs. The measures used for assessing the MOSAs are the QIs, which are again comparable across the MOSAs. One threat to the *construct validity* is the use of the threshold (i.e., 0.4) for the *Kappa* values when defining the observations. We acknowledge that different thresholds may lead to different observations and further investigation is required. Another concern is about the termination criterion used in the MOSAs. We used the same termination criterion for all the MOSAs, i.e., reaching the 25000^{th} fitness evaluation.

Regarding the generalization aspect of threats to the *external validity*, we conducted the experiments with 9 SBSE problems (11 case studies/search problems), covering challenges identified from different phases of software/system development lifecycles such as requirements, rule mining in the product line engineering context, and test optimization. We, however, understand that more real-world case studies and SBSE problems are needed, and lacking real case studies in empirical studies is a common threat to the external validity [3, 6]. Our observations are currently limited to all, one MOSA, and pairs of MOSAs (Sect. 6). Generalizing the observations to cover a wider

scope of application scenarios, however, requires further investigations on devising observations for covering three and more MOSAs for instance.

8 CONCLUSION AND FUTURE WORK

Quality indicators (QIs) help in assessing the quality of the solutions produced by multi-objective search algorithms (MOSAs). To provide insights into how QIs relate to SBSE problems and MOSAs in Search-Based Software Engineering (SBSE), we presented an extensive empirical evaluation with 8 QIs, 6 MOSAs, and 9 SBSE problems (11 case studies/search problems). Key findings are: 1) there exist significant differences among the pairs of QIs; 2) agreements among QIs differ significantly across SBSE problems, however, not across MOSAs; 3) the overall agreement among the 28 pairs of QIs differ significantly across the studied SBSE problems, but not across the selected MOSAs; 4) the significant agreement/disagreement between any given pair of QIs doesn't necessarily mean that the pair measures similar/different quality aspects. We also provided a process to automatically generate observations from the results of statistical analyses to select the QIs in three different experiment settings. Finally, we also automatically produced 22 observations for SBSE researchers to understand the selected QIs. Finally, based on our observations, we devised a set of guidelines for SBSE researchers to select the appropriate QIs for their particular context.

Our future work includes developing an automated online observations generator system for the QIs for SBSE researchers as opposed to the current implementation, where we receive data from the users and update the observations using our scripts and manually update the website with the updated observations. We also plan to perform the following empirical evaluations: 1) In-depth study of a representative set of QIs and characteristics of the SBSE problems; 2) Studying how various parameter settings of the MOSAs relate to the agreements of the QIs; 3) Extending our current empirical evaluation for more than two MOSAs. This will help us providing more detailed guidelines for any given set of MOSAs, e.g., any three, or four MOSAs. Note that currently, we only cover single, pair of algorithms, and all the MOSAs together.

ACKNOWLEDGMENTS

The work is partially supported by the National Natural Science Foundation of China under Grant No. 61872182. The work is also partially supported by the Co-evolver project (No. 286898/F20) funded by the Research Council of Norway under the FRIPRO program. Shaukat Ali was also supported by the Personal Overseas Grant of the IKTPLUSS scheme of Research Council of Norway under the MBT4CPS project. Paolo Arcaini is supported by ERATO HASUO Metamathematics for Systems Design Project (No. JPMJER1603), JST; Funding Reference number: 10.13039/501100009024 ERATO.

REFERENCES

[1] Wasif Afzal, Richard Torkar, and Robert Feldt. 2009. A Systematic Review of Search-based Testing for Non-functional System Properties. *Inf. Softw. Technol.* 51, 6 (June 2009), 957–976. https://doi.org/10.1016/j.infsof.2008.12.005

[2] Alan Agresti. 2013. *Categorical data analysis*. Wiley.

[3] Shaukat Ali, Lionel C. Briand, Hadi Hemmati, and Rajwinder Kaur Panesar-Walawege. 2010. A Systematic Review of the Application and Empirical Investigation of Search-Based Test Case Generation. *IEEE Trans. Softw. Eng.* 36, 6 (Nov. 2010), 742–762. https://doi.org/10.1109/TSE.2009.52

[4] Andrea Arcuri and Lionel Briand. 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, NY, USA, 1–10. https://doi.org/10.1145/1985793.1985795

[5] Wesley Klewerton Guez Assunção, Thelma Elita Colanzi, Silvia Regina Vergilio, and Aurora Pozo. 2013. On the Application of the Multi-Evolutionary and Coupling-Based Approach with Different Aspect-Class Integration Testing Strategies. In *Search Based Software Engineering*, Günther Ruhe and Yuanyuan Zhang (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 19–33.

[6] Márcio Barros and Arilo Neto. 2011. Threats to Validity in Search-based Software Engineering Empirical Studies. *RelaTe-DIA* 5 (01 2011).

[7] Lionel Briand, Davide Falessi, Shiva Nejati, Mehrdad Sabetzadeh, and Tao Yue. 2012. Research-based innovation: A tale of three projects in model-driven engineering. In *International Conference on Model Driven Engineering Languages and Systems*. Springer, 793–809.

[8] Jacob Cohen. 1960. A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement* 20, 1 (1960), 37–46. https://doi.org/10.1177/001316446002000104

[9] Altino Dantas, Italo Yeltsin, Allysson Allex Araújo, and Jerffeson Souza. 2015. Interactive software release planning with preferences base. In *International Symposium on Search Based Software Engineering*. Springer, 341–346.

[10] Kalyanmoy Deb and Deb Kalyanmoy. 2001. *Multi-Objective Optimization Using Evolutionary Algorithms*. John Wiley & Sons, Inc., New York, NY, USA.

[11] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and T. A. M. T. Meyarivan. 2002. A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation* 6, 2 (2002), 182–197. https://doi.org/10.1109/4235.996017

[12] Juan J. Durillo and Antonio J. Nebro. 2011. jMetal: A Java Framework for Multi-objective Optimization. *Adv. Eng. Softw.* 42, 10 (Oct. 2011), 760–771. https://doi.org/10.1016/j.advengsoft.2011.05.014

[13] Juan J. Durillo, Antonio J. Nebro, Francisco Luna, and Enrique Alba. 2008. Solving Three-Objective Optimization Problems Using a New Hybrid Cellular Genetic Algorithm. In *Proceedings of the 10th International Conference on Parallel Problem Solving from Nature — PPSN X - Volume 5199*. Springer-Verlag New York, Inc., New York, NY, USA, 661–670. https://doi.org/10.1007/978-3-540-87700-4_66

[14] Juan J. Durillo, YuanYuan Zhang, Enrique Alba, and Antonio J. Nebro. 2009. A Study of the Multi-objective Next Release Problem. In *Proceedings of the 2009 1st International Symposium on Search Based Software Engineering (SSBSE '09)*. IEEE Computer Society, Washington, DC, USA, 49–58. https://doi.org/10.1109/SSBSE.2009.21

[15] Carlos M. Fonseca and Peter J. Fleming. 1998. Multiobjective optimization and multiple constraint handling with evolutionary algorithms. I. A unified formulation. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans* 28, 1 (Jan 1998), 26–37. https://doi.org/10.1109/3468.650319

[16] G. H. Freeman and J. H. Halton. 1951. Note On an Exact Treatment Of Contingency, Goodness of Fit and Other Problems of Significance. *Biometrika* 38, 1-2 (1951), 141–149. https://doi.org/10.1093/biomet/38.1-2.141

[17] Giovani Guizzo, Silvia R Vergilio, Aurora TR Pozo, and Gian M Fritsche. 2017. A multi-objective and evolutionary hyper-heuristic applied to the integration and test order problem. *Applied Soft Computing* 56 (2017), 331–344.

[18] Mark Harman and Bryan F Jones. 2001. Search-based software engineering. *Information and Software Technology* 43, 14 (2001), 833–839. https://doi.org/10.1016/S0950-5849(01)00189-6

[19] Mark Harman, S. Afshin Mansouri, and Yuanyuan Zhang. 2012. Search-based Software Engineering: Trends, Techniques and Applications. *ACM Comput. Surv.* 45, 1, Article 11 (Dec. 2012), 61 pages. https://doi.org/10.1145/2379776.2379787

[20] Joshua D. Knowles and David W. Corne. 2000. Approximating the Nondominated Front Using the Pareto Archived Evolution Strategy. *Evol. Comput.* 8, 2 (June 2000), 149–172. https://doi.org/10.1162/106365600568167

[21] J. Richard Landis and Gary G. Koch. 1977. The Measurement of Observer Agreement for Categorical Data. *Biometrics* 33, 1 (1977).

[22] Miqing Li, Tao Chen, and Xin Yao. 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, NY, USA, 17–20. https://doi.org/10.1145/3183399.3183405

[23] Miqing Li and Xin Yao. 2019. Quality Evaluation of Solution Sets in Multiobjective Optimisation: A Survey. *Comput. Surveys* (13 12 2019).

[24] R. E. Lopez-Herrejon, J. Ferrer, F. Chicano, A. Egyed, and E. Alba. 2014. Comparative analysis of classical multi-objective evolutionary algorithms and seeding strategies for pairwise testing of Software Product Lines. In *2014 IEEE Congress on Evolutionary Computation (CEC)*. 387–396.

[25] Roberto E. Lopez-Herrejon, Lukas Linsbauer, and Alexander Egyed. 2015. A Systematic Mapping Study of Search-based Software Engineering for Software Product Lines. *Inf. Softw. Technol.* 61, C (May 2015), 33–51. https://doi.org/10.1016/j.infsof.2015.01.008

[26] Francisco Luna, David L. González-Álvarez, Francisco Chicano, and Miguel A. Vega-Rodríguez. 2014. The software project scheduling problem: A scalability analysis of multi-objective metaheuristics. *Applied Soft Computing* 15 (2014), 136–148. https://doi.org/10.1016/j.asoc.2013.10.015

[27] Phil McMinn. 2011. Search-Based Software Testing: Past, Present and Future. In *Proceedings of the 2011 IEEE Fourth International Conference on Software Testing, Verification and Validation Workshops (ICSTW '11)*. IEEE Computer Society, Washington, DC, USA, 153–163. https://doi.org/10.1109/ICSTW.2011.100

[28] Antonio J. Nebro, Juan J. Durillo, Jose Garcia-Nieto, Carlos A. Coello Coello, Francisco Luna, and Enrique Alba. 2009. SMPSO: A new PSO-based metaheuristic for multi-objective optimization. In *2009 IEEE Symposium on Computational Intelligence in Multi-Criteria Decision-Making(MCDM)*. 66–73. https://doi.org/10.1109/MCDM.2009.4938830

[29] Antonio J. Nebro, Juan J. Durillo, Francisco Luna, Bernabé Dorronsoro, and Enrique Alba. 2009. MOCell: A Cellular Genetic Algorithm for Multiobjective Optimization. *Int. J. Intell. Syst.* 24, 7 (July 2009), 726–746. https://doi.org/10.1002/int.v24:7

[30] Dipesh Pradhan, Shuai Wang, Shaukat Ali, and Tao Yue. 2016. 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, NY, USA, 1085–1092. https://doi.org/10.1145/2908812.2908850

[31] Dipesh Pradhan, Shuai Wang, Shaukat Ali, Tao Yue, and Marius Liaaen. 2016. 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, 172–190.

[32] Dipesh Pradhan, Shuai Wang, Shaukat Ali, Tao Yue, and Marius Liaaen. 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, 46–57.

[33] Aurora Ramírez, José Raúl Romero, and Christopher L. Simons. 2019. A Systematic Review of Interaction in Search-Based Software Engineering. *IEEE Transactions on Software Engineering* 45, 8 (Aug 2019), 760–781. https://doi.org/10.1109/TSE.2018.2803055

[34] Miha Ravber, Marjan Mernik, and Matej Črepinšek. 2017. The Impact of Quality Indicators on the Rating of Multi-objective Evolutionary Algorithms. *Appl. Soft Comput.* 55, C (June 2017), 265–275. https://doi.org/10.1016/j.asoc.2017.01.038

[35] Safdar Aqeel Safdar, Hong Lu, Tao Yue, and Shaukat Ali. 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, NY, USA, 1319–1326. https://doi.org/10.1145/3071178.3071261

[36] David J. Sheskin. 2011. *Handbook of Parametric and Nonparametric Statistical Procedures* (5 ed.). Chapman & Hall/CRC.

[37] Helge Spieker, Arnaud Gotlieb, Dusica Marijan, and Morten Mossige. 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, 12–22.

[38] M. Tanaka, H. Watanabe, Y. Furukawa, and T. Tanino. 1995. GA-based decision support system for multicriteria optimization. In *1995 IEEE International Conference on Systems, Man and Cybernetics. Intelligent Systems for the 21st Century*, Vol. 2. 1556–1561 vol.2. https://doi.org/10.1109/ICSMC.1995.537993

[39] David A. Van Veldhuizen and Gary B. Lamont. 2000. Multiobjective Evolutionary Algorithms: Analyzing the State-of-the-Art. *Evol. Comput.* 8, 2 (June 2000), 125–147. https://doi.org/10.1162/106365600568158

[40] Niki Veček, Marjan Mernik, and Matej Črepinšek. 2014. A chess rating system for evolutionary algorithms: A new method for the comparison and ranking of evolutionary algorithms. *Information Sciences* 277 (2014), 656–679. https://doi.org/10.1016/j.ins.2014.02.154

[41] Shuai Wang, Shaukat Ali, and Arnaud Gotlieb. 2015. Cost-effective Test Suite Minimization in Product Lines Using Search Techniques. *J. Syst. Softw.* 103, C (May 2015), 370–391. https://doi.org/10.1016/j.jss.2014.08.024

[42] Shuai Wang, Shaukat Ali, Tao Yue, Yan Li, and Marius Liaaen. 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, NY, USA, 631–642. https://doi.org/10.1145/2884781.2884880

[43] Shuai Wang, Shaukat Ali, Tao Yue, and Marius Liaaen. 2015. UPMOA: An improved search algorithm to support user-preference multi-objective optimization. In *Software Reliability Engineering (ISSRE), 2015 IEEE 26th International Symposium on*. IEEE, 393–404.

[44] Zai Wang, Ke Tang, and Xin Yao. 2008. A multi-objective approach to testing resource allocation in modular software systems. In *Evolutionary Computation, 2008. CEC 2008.(IEEE World Congress on Computational Intelligence). IEEE Congress on*. IEEE, 1148–1153.

[45] Tao Yue and Shaukat Ali. 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, NY, USA, 1295–1302. https://doi.org/10.1145/2576768.2598309

[46] Yuanyuan Zhang, Mark Harman, and Afshin Mansouri. 2017. The SBSE Repository: A repository and analysis of authors and research articles on Search Based Software Engineering. http://crestweb.cs.ucl.ac.uk/resources/sbse_repository/. (2017). Last access: 2019-10-22.

[47] Aimin Zhou, Yaochu Jin, Qingfu Zhang, Bernhard Sendhoff, and Edward Tsang. 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*. 892–899. https://doi.org/10.1109/CEC.2006.1688406

[48] E. Zitzler, M. Laumanns, and L. Thiele. 2001. SPEA2: Improving the strength Pareto evolutionary algorithm for multiobjective optimization. In *Evolutionary Methods for Design Optimization and Control with Applications to Industrial Problems* (19–21 September 2001), K. C. Giannakoglou, D. T. Tsahalis, J. Périaux, K. D. Papailiou, and T. Fogarty (Eds.). International Center for Numerical Methods in Engineering, Athens, Greece, 95–100.

[49] E. Zitzler and L. Thiele. 1999. Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach. *IEEE Transactions on Evolutionary Computation* 3, 4 (Nov 1999), 257–271. https://doi.org/10.1109/4235.797969