A Study of the Use of Multi-Objective Evolutionary Algorithms to Learn Boolean Queries: A Comparative Study

A.G. López-Herrera, E. Herrera-Viedma, and F. Herrera
Department of Computer Sciences and A.I., University of Granada, E-18071-Granada, Spain.
E-mail: {lopez-herrera, viedma, herrera}@decsai.ugr.es

In this article, our interest is focused on the automatic learning of Boolean queries in information retrieval systems (IRSs) by means of multi-objective evolutionary algorithms considering the classic performance criteria, precision and recall. We present a comparative study of four well-known, general-purpose, multi-objective evolutionary algorithms to learn Boolean queries in IRSs. These evolutionary algorithms are the *Nondominated Sorting Genetic Algorithm* (NSGA-II), the first version of the *Strength Pareto Evolutionary Algorithm* (SPEA), the second version of SPEA (SPEA2), and the *Multi-Objective Genetic Algorithm* (MOGA).

Introduction

Information retrieval (IR) may be defined as the problem of selecting documentary information from storage in response to searches provided by a user in the form of queries (Baeza-Yates & Ribeiro-Neto, 1999; Salton & McGill, 1983). Information retrieval systems (IRSs) deal with documentary databases containing textual, pictorial, or vocal information. They process user queries to allow the user to access relevant information in an appropriate time interval.

The Boolean IR model (van Rijsbergen, 1979) is frequently used to build queries in the IRSs; however, it presents some limitations: A Boolean query is defined by a set of terms joined by the logical operators AND, OR, and NOT, but to build Boolean queries is not usually easy or very intuitive (Baeza-Yates & Ribeiro-Neto, 1999). This problem becomes a more serious issue if the users do not have previous experience with the model. A possible solution to overcome this problem is to build automatic aid tools to assist users to express their information needs by means of Boolean queries. *Inductive Query By Example* (IQBE; Chen, Shankaranarayanan, She, & Iyer, 1998), where a query describing the information contexts of a set of key documents provided by the user is automatically derived or learned, is a useful paradigm to assist users to express Boolean queries.

The most well-known, in the context of Boolean IR, IQBE approach is that of Smith and Smith (1997), which is based on Genetic Programming (GP; Koza, 1992). This approach is called Boolean IQBE-GP, and it is able to derive Boolean queries. As is usual in the field (Cordón, Herrera-Viedma, López-Pujalte, Luque, & Zarco, 2003), this approach is guided by a weighted fitness function combining the classic retrieval-accuracy criteria, precision and recall (van Rijsbergen, 1979). We will use this Boolean IQBE-GP approach together with a Boolean IRS model in this article.

Given that the retrieval performance of an IRS is usually measured in terms of precision and recall criteria, the optimization of any of its components and, concretely, the automatic learning of queries, is a clear example of a multi-objective problem. Evolutionary algorithms (EAs) have been commonly used for IQBE purposes, and their application in the area has been based on combining both criteria in a single scalar fitness function by means of a weighting scheme (Chen et al., 1998). However, there is a kind of EA specially designed for multi-objective problems, multi-objective evolutionary algorithms (MOEAs), which are able to obtain nondominated solutions to the problem in a unique run (Coello, Van Veldhuizen, & Lamant, 2002; Deb, 2001). This characteristic of MOEAs, applied to IR—especially to the context of the IQBE paradigm, denoted as IQBE MOEA—allows one to derive a number of queries with a different precision–recall trade-off, which improves the aid possibilities to the users in the formulation of their Boolean queries.

Received February 1, 2008; revised January 19, 2009; accepted January 20, 2009

© 2009 ASIS&T • Published online 20 March 2009 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/asi.21060

In the literature, there are not many IQBE MOEAs to derive queries in IRSs, and besides, they are based on old-fashioned MOEAs. Three examples are:

- IQBE-MOGA-P (Cordón, Moya, & Zarco, 2004) based on MOGA (Multi-Objective Genetic Algorithm; Fonseca & Fleming, 1993) and GA-P (Genetic Algorithm-Programming; Howard & D'Angelo, 1995) to learn fuzzy queries with numerical weights,
- IQBE-SPEA (Cordón, Herrera-Viedma, & Luque, 2006a) based on SPEA (Strength Pareto Evolutionary Algorithm; Zitzler & Thiele, 1999) and GP to derive Boolean queries, and
- IQBE-SPEA-GA (Cordón, Herrera-Viedma, & Luque, 2006b) based on SPEA and Genetic Algorithms (Michalewicz, 1996) to learn fuzzy multiweighted queries with ordinal linguistic weights in a multigranular fuzzy ordinal linguistic IRS (Herrera-Viedma, Cordón, Luque, López, & Muñoz, 2003).

Although these approaches have great performance, there exist other, more advanced MOEAs in the specialized literature that never have been used in the IQBE context. Those MOEAs can improve the performance of the existing IQBE MOEAs in the context of automatic derivation of queries (Boolean queries, numerical weighted queries, or ordinal weighted queries) in IRSs.

In this work, an analysis of the performance of four well-known and successful MOEAs applied to the automatic learning of Boolean queries in the context of a Boolean IRS model is presented. The used MOEAs are: the classic *Multi-Objective Genetic Algorithm* (MOGA; Fonseca & Fleming, 1993), the *Strength Pareto Evolutionary Algorithm* (SPEA; Zitzler & Thiele, 1999), the more recent *Nondominated Sorting Genetic Algorithm* (NSGA-II; Deb, Pratap, Agrawal, & Meyarivan, 2002) and the second version of SPEA (SPEA2; Zitzler, Laumanns, & Thiele, 2002). All of them are adapted to use GP components and to optimize both precision and recall, simultaneously, extending Smith and Smith's (1997) Boolean IQBE-GP proposal into the multi-objective context.

The analysis of the proposal will include the use of the Cranfield, CACM, MEDLINE, and a subset of TREC (TREC-WSJ[1]) collections (Baeza-Yates & Ribeiro-Neto, 1999; Salton, 1989), the evaluation of the pareto fronts with *C* measure and hypervolume indicator, the analysis of the number of derived queries, the comparison with the classic Boolean IQBE-GP proposal (Smith & Smith, 1997), and the use of nonparametric statistical methods (Demsar, 2006; García & Herrera, 2008; García, Molina, Lozano, & Herrera, in press; Sheskin, 2003) to compare and analyze the experimental result to determine the best IQBE MOEA-GP approach.

To do that, this article is structured as follows. First, the IQBE paradigm and the Boolean IRS model used in this article are drawn. We then describe the four MOEAs with GP components studied. Next, the experimental framework and analysis are presented. Finally, some concluding remarks are given. In the Appendix, we include tables with some statistical tests.

Preliminaries: IQBE and Boolean IRS

In this section, we introduce the IQBE paradigm and the foundations of the Boolean IRS model used in this article, including their components and procedure of evaluation.

The IQBE Paradigm

The IQBE paradigm was proposed by Chen et al. (1998) as "a process in which searchers (users) provide documents (examples) and an algorithm induces (or learns) the key concepts of the examples with the purpose of finding other equally relevant documents" (p. 694). In this way, IQBE can be seen as a technique to assist users in the query-building process by using automatic learning methods.

Assuming a set of relevant documents (or alternatively, nonrelevant documents) which represents user information needs (which can be obtained from a preliminary query or from a browsing process through the documentary database), the IQBE technique consists of developing an automatic learning process to generate a query that describes the user information needs (see Figure 1). The learned query can be executed again in the same IRS or in other IRSs to obtain new relevant documents. In this way, it is not necessary for the user to interact with the IR process, which is mandatory in other techniques for query refinement as the *relevance feedback* (Salton, 1989).

Several IQBE EAs for different IR models have been proposed and revised in Cordón et al. (2003). The most used IQBE models are based on GP components, with queries being represented by expression syntax trees and where the algorithms are articulated on the basis of the classic operators: cross, mutation, and selection.

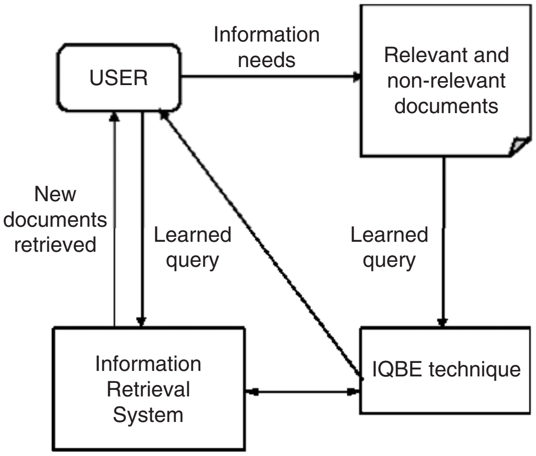


FIG. 1. Inductive Query By Example (IQBE) process.

[1] TREC-WSJ is a subset of TREC, which contains 5,000 articles published in the *Wall Street Journal* in 1990, 1991 and 1992.

DOI: 10.1002/asi

Boolean IR Systems

In the following subsections, we briefly present the components of the Boolean IRS model: the documentary database, the query subsystem, the matching subsystem and the evaluation model.

The documentary database. This component stores the documents and the representation of their contents. Textual documents representation is typically based on index terms (that can be either single terms or sequences), which work as content identifiers for the documents. We assume that the database is built like those in usual IRSs (Baeza-Yates & Ribeiro-Neto, 1999; Salton & McGill, 1983). Therefore, IRS–user interaction is unnecessary because it is built automatically. The database stores a finite set of documents $\mathcal{D} = \{d_1, \ldots, d_m\}$, a finite set of index terms $\mathcal{T} = \{t_1, \ldots, t_l\}$, and the representation R_{d_j} of each document d_j characterized by a numeric indexing function $\mathcal{F} : \mathcal{D} \times \mathcal{T} \to \{0, 1\}$. $\mathcal{F}(d_j, t_i) = 0$ implies that the document d_j contents do not deal at all with the concept(s) represented by the index term t_i, and $\mathcal{F}(d_j, t_i) = 1$ implies that the document d_j is perfectly represented by the concept(s) indicated by t_j.

The query subsystem. This subsystem allows users to formulate their information needs (i.e., queries) and presents the relevant documents retrieved by the system. To do this, each query is expressed as a combination of index terms which are connected by the Boolean operators AND (∧), OR (∨), and NOT (¬). We define a Boolean query as any legitimate Boolean expression defined by the following syntactic rules:

1. $\forall q = t_i \in \mathcal{T} \to q \in \mathcal{Q}$.
2. $\forall q, p \in \mathcal{Q} \to q \wedge p \in \mathcal{Q}$.
3. $\forall q, p \in \mathcal{Q} \to q \vee p \in \mathcal{Q}$.
4. $\forall q \in \mathcal{Q} \to \neg q \in \mathcal{Q}$.
5. All legitimate queries $q \in \mathcal{Q}$ are only those obtained by applying Rules 1 to 4, inclusive.

The matching subsystem. This system evaluates the degrees (i.e., the retrieval status value) to which the document representation satisfies the requirements expressed in the query and retrieves the documents that are judged to be relevant. The evaluation subsystem is implemented by the matching or evaluation function ε, which assesses the relationship between $\mathcal{Q}$ and $\mathcal{D}$. Therefore, the goal of ε consists of evaluating documents in terms of their relevance to a Boolean query. ε is defined by means of a constructive bottom-up evaluation process that acts in two steps:

1. The documents are evaluated according to their relevance only to the terms of the query. In this step, a partial relevance degree is assigned to each document with respect to every term in the query.
2. The documents are evaluated according to their relevance to the Boolean combination of the terms (their partial relevance degree), and so on, working in a bottom-up fashion until the whole query is processed. In this step, documents are finally classified as relevant or nonrelevant.

Evaluation of IRSs. There are several ways to measure the quality of an IRS, such as the system efficiency and effectiveness, and several subjective aspects related to user satisfaction (Baeza-Yates & Ribeiro-Neto, 1999). Traditionally, the retrieval effectiveness is based on the document relevance with respect to the user's needs. There are different criteria to measure this aspect, but precision (P) and recall (R) (van Rijsbergen, 1979) are the most used. *Precision* is the ratio between the relevant documents retrieved by the IRS in response to a query and the total number of documents retrieved while *recall* is the ratio between the number of relevant documents retrieved and the total number of relevant documents for the query that exists in the database (van Rijsbergen, 1979). The mathematical expression of each of them is:

$$P = \frac{\mathcal{D}_{rr}}{\mathcal{D}_{tr}};\ R = \frac{\mathcal{D}_{rr}}{\mathcal{D}_{rt}} \qquad (1)$$

where $\mathcal{D}_{rr}$ is the number of relevant documents retrieved, $\mathcal{D}_{tr}$ is the total number of documents retrieved, and $\mathcal{D}_{rt}$ is the total number of relevant documents for the query which exists in the database. P and R are defined in [0,1], 1 being the optimal value.

Note that the only way to know all the relevant documents existing for a query in the database (value used in the R measure) is to evaluate all documents. Due to this fact and taking into account that relevance is subjective, there are some classic documentary databases (TREC-WSJ, CACM, Cranfield, MEDLINE, ADI, CISI, etc.) available, each one with a set of queries for which the relevance judgments are known, so that they can be used to verify the new proposals in the field of the IR (Baeza-Yates & Ribeiro-Neto, 1999; Salton, 1989). In this contribution, we use the Cranfield, CACM, MEDLINE, and TREC-WSJ[2] collections.

Structure of the MOEAs With GP components

For the purposes of this research, four well-known and widely used MOEAs have been selected for performance evaluation: NSGA-II (Deb et al., 2002), SPEA (Zitzler & Thiele, 1999), SPEA2 (Zitzler et al., 2002), and MOGA (Fonseca & Fleming, 1993).

Objectives and Evaluation of MOEAs

In multi-objective optimization problems, the definition of the *quality* concept is substantially more complex than in single-objective ones since the optimization processes imply several different objectives.

The key concepts to evaluate MOEAs are the *dominance relation* and the *Pareto sets*.

The algorithms presented in this article assume the two classic criteria to evaluate IRSs, *precision* and *recall* (van Rijsbergen, 1979), whose expressions were introduced earlier. The studied IQBE MOEA approaches assume that all

[2] The results correspond to Test Query 1 of the Cranfield collection.

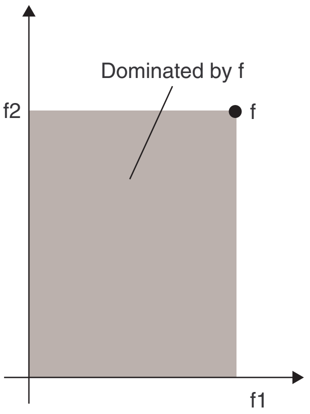


FIG. 2. Concept of dominance.

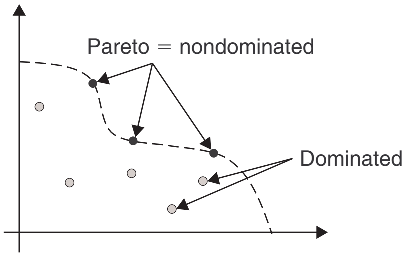


FIG. 3. Concept of pareto set.

objectives have to be maximized. The solutions are represented by objective vectors, which are compared according to the dominance relation defined next and displayed in Figure 2.

Definition 1. (Dominance relation). Let $f, g \in \mathbb{R}^m$. Then f is said to dominate g, denoted as $f \succ g$, iff

1. $\forall i \in \{1, \ldots, m\}\ :\ f_i \geq g_i$,
2. $\exists j \in \{1, \ldots, m\} : f_j > g_j$.

Based on the concept of dominance, the Pareto set can be defined as follows.

Definition 2. (Pareto set). Let $F \subseteq \mathbb{R}^m$ be a set of vectors. Then the Pareto set (see Figure 3) F^* of F is defined as follows: F^* contains all vectors $g \in F$ which are not dominated by any vector $f \in F$; that is,

$$F^* := \{g \in F | \nexists f \in F : f \succ g\}.$$

Several quantitative measures based on the pareto set concept have been proposed in the specialized literature (Coello et al., 2002; Deb, 2001; Knowles, Thiele, & Zitzler, 2006; Zitzler, Deb, & Thiele, 2000). Two of them are the C measure and the hypervolume indicator.

Definition 3. (Coverage of two sets; Zitzler & Thiele, 1998). Let A, B be two sets of objective vectors. The function C maps the ordered pair (A,B) to the interval [0,1]:

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

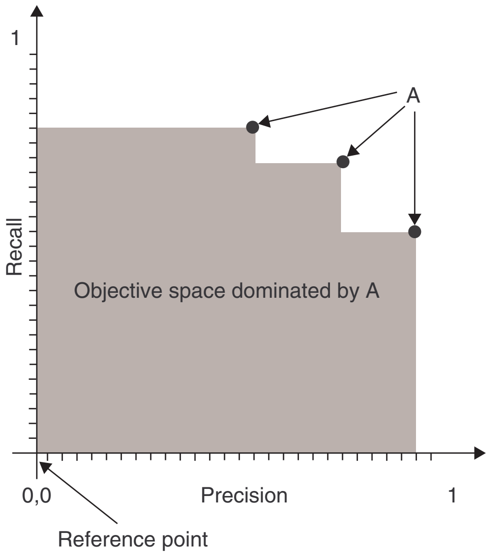


FIG. 4. The hypervolume indicator.

measures the ratio of individuals of the Pareto A that is dominated by individuals of the Pareto B. A value of 1 indicates that all individuals of the Pareto A are dominated by individuals of the Pareto B; a value of 0 indicates that none of the individuals of A are dominated by individuals of B. Note that both directions always have to be considered since $C(A,B)$ is not necessarily equal to $1 - C(B,A)$.

Definition 4. (Hypervolume; Zitzler & Thiele, 1999). This indicator measures the hypervolume of that portion of the objective space that is dominated by an approximation set A, and is to be maximized (see Figure 4). To measure this quantity, the objective space must be bounded. If it is not, a bounding reference point that is dominated by all points should be used, as shown in the Figure 4. In our case, the reference point is (0,0).

We will use both C and hypervolume in this study.

MOGA-GP

In MOGA (Fonseca & Fleming, 1993), the rank of a certain individual corresponds to the number of chromosomes in the current population by which it is dominated. Consider, for example, an individual x_i at generation t which is dominated by p_t^i individuals in the current generation. The rank of an individual is given by: $rank(x_i, t) = 1 + p_t^i$.

All nondominated individuals are assigned rank 1 while dominated ones are penalized according to the population density of the corresponding region of the trade-off surface.

Fitness assignment is performed in the following way:

1. Sort population according to rank.
2. Assign fitness to individuals by interpolating from the best (rank 1) to the worst (rank $n \leq M$, where M is the total population size) in the way proposed by Goldberg (1989), according to some function, usually, but not necessarily linear.

DOI: 10.1002/asi

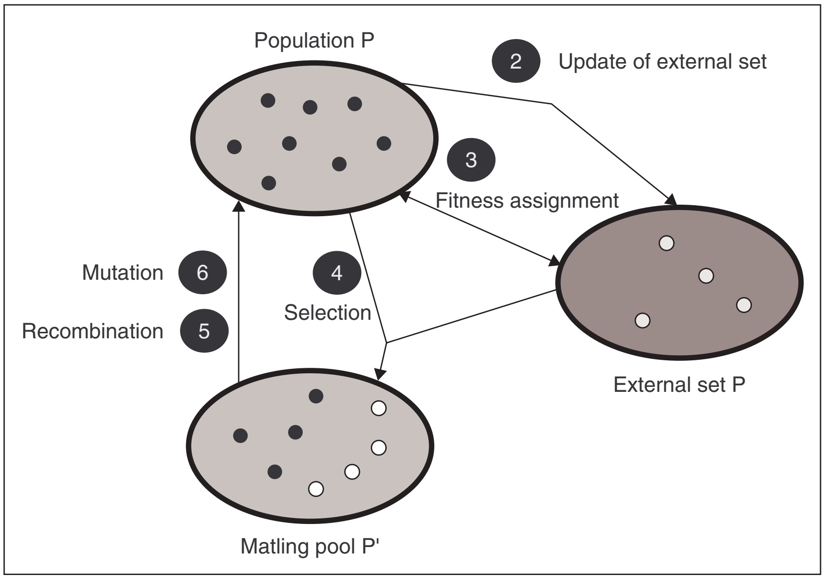


FIG. 5. The Strength Pareto Evolutionary Algorithm (SPEA) procedure.

3. Average the fitness of individuals with the same rank so that all of them are sampled at the same rate. This procedure keeps the global population fitness constant while maintaining appropriate selective pressure, as defined by the function used.

The general expression used is: $fitness(x_i, t) = 1/rank(x_i, t)$.

In this article, MOGA will be adapted to use GP components. It will be denoted as IQBE-MOGA-GP.

SPEA-GP

SPEA (Zitzler & Thiele, 1999) uses an archive containing nondominated solutions previously found (the so-called *external nondominated set*). At each generation, nondominated individuals are copied to the external nondominated set. For each individual in this external set, a *strength* value is computed. This strength is similar to the ranking value of MOGA since it is proportional to the number of solutions to which a certain individual dominates. Clearly, the external nondominated set is in this case the adopted elitist mechanism.

In SPEA, the fitness of each member of the current population is computed according to the strengths of all external nondominated solutions that dominate it. Additionally, a clustering technique called "average linkage method" (Morse, 1980) is used to keep diversity. The main loop of SPEA is outlined as follows and drawn in Figure 5.

Input:

- N (population size)
- $\overline{N}$ (maximum size of external set)
- T (maximum number of generations)
- p_c (crossover probability)
- p_m (mutation rate)

Output: A (nondominated set)

1. Initialization: Generate an initial population P_0 and create the empty external set $\overline{P}_0 = \emptyset$. Set $t = 0$.
2. Update of external set: Set the temporary external set $\overline{P}' = \overline{P}_t$.
   (a) Copy individuals whose decision vectors are nondominated regarding P_t to $\overline{P'}$.
   (b) Remove individuals from $\overline{P'}$ whose corresponding decision vectors are dominated regarding $\overline{P'}$.
   (c) Reduce the number of individuals externally stored in $\overline{P'}$ by means of clustering and assign the resulting reduced set to $\overline{P_{t+1}}$.
3. Fitness assignment: Calculate fitness values of individuals in P_t and $\overline{P_t}$. Each individual $i \in \overline{P_t}$ is assigned a real value $S(i) \in [0, 1)$, called strength, $S(i)$ is proportional to the number of population members $j \in P_t$ for which i dominates j:

$$S(i) = \frac{|\{j | j \in P_t \wedge idominatesj\}|}{N + 1}$$

The fitness of i is equal to its strength: Fitness$(i) = S(i)$.

The fitness of an individual $j \in P_t$ is calculated by summing the strengths of all externally stored individuals $i \in \overline{P_t}$ whose decision vectors dominate j. We add 1 to the total to guarantee that members of $\overline{P_t}$ have better fitness than do members of $\overline{P_t}$ (Note that fitness is to be minimized here; i.e., small fitness values correspond to high reproduction probabilities.)

4. Selection: $P' = \emptyset$. For $i = 1, \ldots, N$, do
   (a) Select two individuals $i, j \in P_t + \overline{P_t}$ at random.
   (b) If fitness$(i) >$ fitness(j), then $P' = P' + i$ else $P' = P' + j$ (Note that fitness is to be minimized here.)
5. Crossover (discussed later).
6. Mutation (discussed later).
7. Termination.

In this article, SPEA will be adapted to use GP components. It will be denoted as IQBE-SPEA-GP.

SPEA2-GP

SPEA2 (Zitzler et al., 2002) introduces elitism by explicitly maintaining an external population. This population stores a fixed number of the nondominated solutions found from the beginning of the experiment.

In each generation, the new nondominated solutions are compared with the existing external population, and the resulting nondominated solutions are preserved. In addition, SPEA2 uses these elite solutions in the genetic operations with the current population to guide the population towards good regions in the search space.

The algorithm begins with a randomly created population P_0 of size M and an external population $\overline{P}_0$ (initially empty) which has a maximum capacity $\overline{M}$. In each generation t, the best nondominated solutions (belonging to the best nondominated front) of the populations P_t and $\overline{P}_t$ are copied in the external population $\overline{P}_{t+1}$. If the size of $\overline{P}_{t+1}$ exceeds $\overline{M}$, then $\overline{P}_{t+1}$ is reduced by means of a truncate operator; on the other hand, $\overline{P}_{t+1}$ is filled up with dominated solutions from P_t and $\overline{P}_t$. This truncate operator is used to maintain the diversity of the solutions.

From $\overline{P}_{t+1}$, a pool of individuals is obtained applying a binary tournament selection operator with replacement. These individuals are crossed and mutated to obtain the new generation P_{t+1}.

In this article, SPEA2 will be adapted to use GP components. It will be denoted as IQBE-SPEA2-GP.

NSGA-II-GP

NSGA-II (Deb et al., 2002) is a MOEA that incorporates a preservation strategy of an elite population and uses an explicit mechanism (crowded comparison operator) to preserve diversity.

NSGA-II works with an offspring population Q_t, which is created using the predecessor population P_t. Both populations (Q_t and P_t) are combined to form a unique population R_t, with a size $2{\cdot}M$, that is examined to extract the front of the pareto. Then, an arrangement on the nondominated individuals is done to classify the R_t population. Although this implies a greater effort compared with the arrangement of the set Q_t, it allows a global verification of the nondominated solutions that belong as well as to the population of offspring or to the one of the predecessors.

Once the arrangement of the nondominated individuals finishes, the new generation (population) P_{t+1} is formed with solutions of the different nondominated fronts ($F_1, \ldots, F_m$), taking them alternatively from each of the fronts. It begins with the best front of nondominated individuals and continues with the solutions of the second one, and so on.

Since the R_t size is $2{\cdot}M$, it is possible that some of the front solutions have to be eliminated to form the new population.

In the last states of the execution, it is usual that the majority of the solutions is in the best front of nondominated solutions. It also is probable that the size of the best front of the combined population R_t is larger than M. It is then when the previous algorithm assures the selection of a diverse set of solutions of this front by means of the crowded comparison operator (The NSGA-II procedure is shown in Figure 6.) When the whole population converges to the pareto-optimal frontier, the algorithm continues so that the best distribution between the solutions is assured.

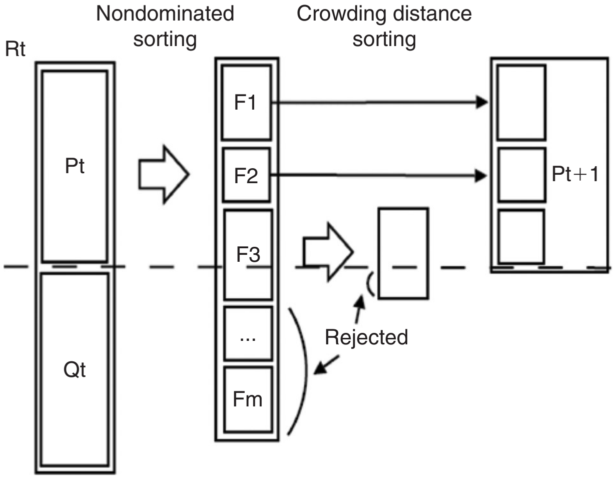


FIG. 6. The Nondominated Sorting Genetic Algorithm (NSGA-II) procedure.

In this article, NSGA-II will be adapted to use GP components. It will be denoted as IQBE-NSGA-II-GP.

GP Components for MOEAs

The four MOEAs-GP studied in this article share the following components:

- *Codification scheme:* Boolean queries are encoded in expression syntax trees, whose terminal nodes are terms and whose inner nodes are the Boolean operators *AND*, *OR*, and *NOT*. Hence, the natural representation is to encode the query within a tree and to work with a GP algorithm (Koza, 1992) to evolve it, as done by previous approaches devoted to the derivation of Boolean queries (Cordón et al., 2006a; Smith & Smith, 1997). Figure 7 shows a graphical example of this kind of query.
- *Crossover operator:* Subtrees are randomly selected and crossed-over in two randomly selected queries, as drawn in Figure 8.
- *Mutation operator:* A randomly selected term or operator is changed in a randomly selected tree. An example is shown in Figure 9.

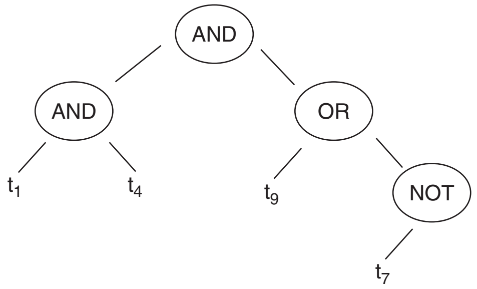


FIG. 7. An example of a Boolean query.

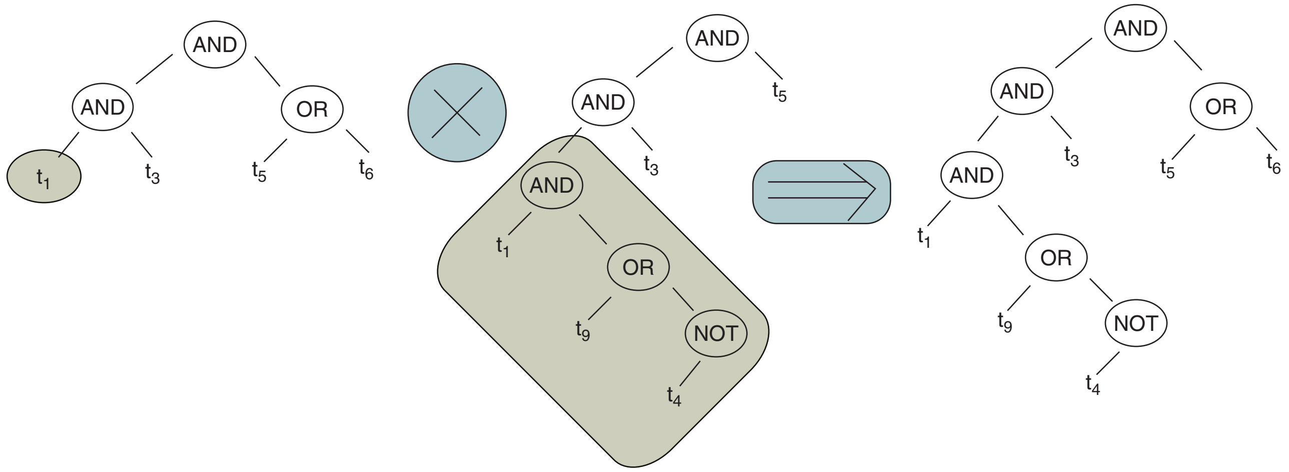


FIG. 8. An example of crossover on two queries.

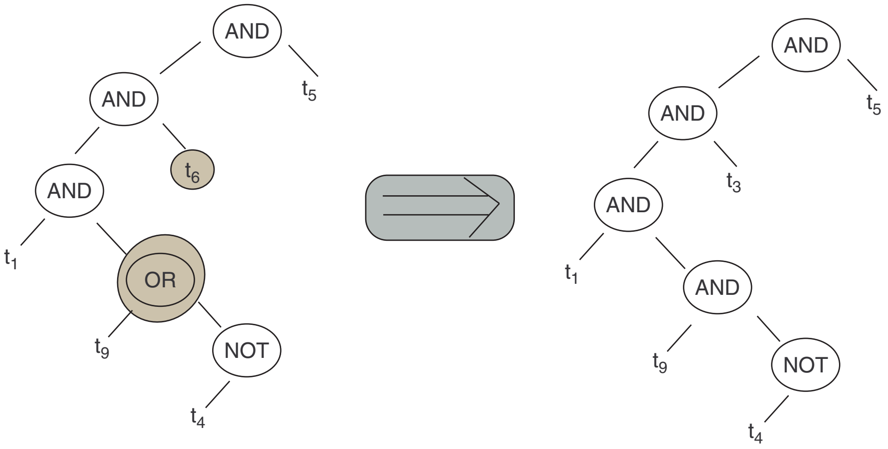


FIG. 9. An example of mutation.

- *Initial population:* All individuals of the first generation are generated in a random way. The population is created including all the terms from the relevant documents provided by the user. Those that appear in more relevant documents will have greater probability (0.8) of being selected. This is done as shown in Figure 10.

Experimental Study

This section is divided into three parts according to the following contents. The first subsection introduces the experimental framework. The next subsection briefly describes the nonparametric statistical test used for statistical analysis in this article. The last subsection presents four different studies for each used performance measure (*C* measure, hypervolume indicator, the number of derived queries, and spent time), and a summary analysis.

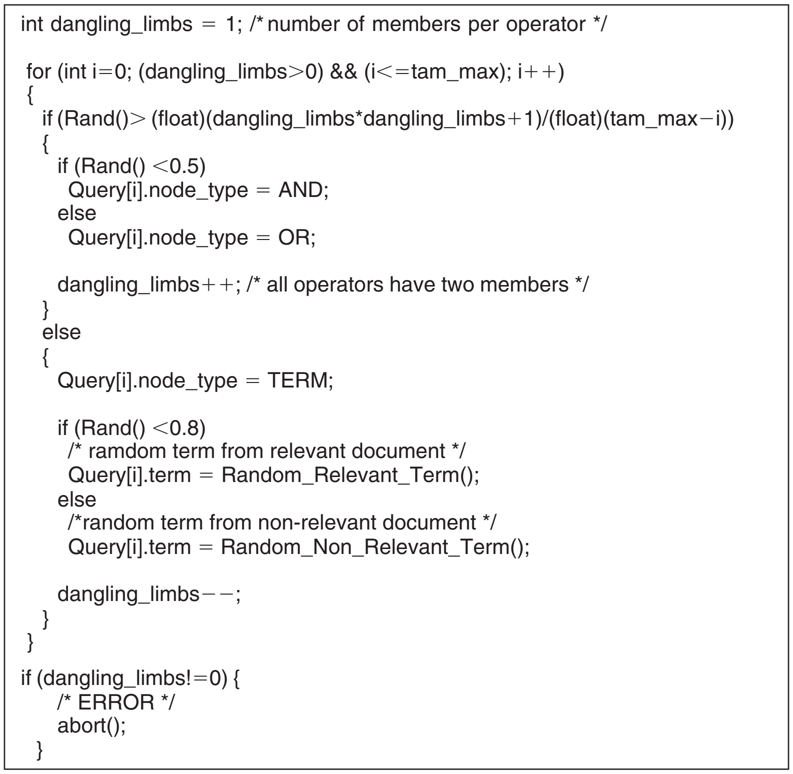
Experimental Framework

The experimental study has been developed using well-known test collections: Cranfield, CACM, MEDLINE, and TREC-WSJ (Baeza-Yates & Ribeiro-Neto, 1999; Salton, 1989):

- Crandfield is composed of 1,398 documents about aeronautics.
- CACM contains 3,204 documents published in the journal *Communications of the ACM* between 1958 and 1979.
- MEDLINE contains 1,033 documents about medicine.
- TREC-WSJ is a subset of TREC, which contains 5,000 articles published in the *Wall Street Journal* in 1990, 1991, and 1992.

In these collections, the textual documents have been automatically indexed in the usual way by using Salton's (1997) classic SMART retrieval system: First, extracting the nonstop words and performing a stemming process, thus obtaining a total number of 3,857; 7,562; 7,170; and 28,904 different indexing terms, respectively. Then, a *binary* weighing has been used to generate the term weights ({0, 1}) in the document representations, using the "bxx" weighting scheme as described in Salton and Buckley (1988).

The used test collections have a number of associated predefined test queries (225 in the Cranfield collection, 64 in the



```
int dangling_limbs = 1; /* number of members per operator */

 for (int i=0; (dangling_limbs>0) && (i<=tam_max); i++)
 {
   if (Rand()> (float)(dangling_limbs*dangling_limbs+1)/(float)(tam_max−i))
   {
     if (Rand() <0.5)
       Query[i].node_type = AND;
     else
       Query[i].node_type = OR;

     dangling_limbs++; /* all operators have two members */
   }
   else
   {
     Query[i].node_type = TERM;

     if (Rand() <0.8)
       /* ramdom term from relevant document */
       Query[i].term = Random_Relevant_Term();
     else
       /*random term from non-relevant document */
       Query[i].term = Random_Non_Relevant_Term();

     dangling_limbs−−;
   }
 }
if (dangling_limbs!=0) {
     /* ERROR */
     abort();
  }
```

FIG. 10. Procedure to build an initial query.[3]

CACM collection, 30 in the MEDLINE collection, and 67 in the TREC-WSJ collection).

The experimental study (see Figure 11) has been developed as in (Abdelmgeid, 2007; Cordón, Herrera-Viedma, & Luque, 2002a, 2002b; Cordón et al., 2003; Cordón, Moya, & Zarco, 2000; Kraft, Petry, Buckles, & Sadasivan, 1997; Smith & Smith, 1997), where the role of the user providing relevant documents is played by the sets of relevant documents grouped or given by the test queries from the Cranfield, CACM, MEDLINE, and TREC-WSJ. In our problem, each test query defines a different set of relevant documents. For us, each test query generates an experiment, and our goal is to automatically derive a set of queries that describes the information contents of the set of documents associated with it, and then to use that query on the full collection to retrieve yet more relevant documents. In this sense, test queries are used only to group documents.

The evaluation procedure to analyze the IQBE MOEAs-GP performance is based on demonstrating that the learning system is able to learn the best queries to represent user information needs provided in the system by means of relevant document(s) (given by test queries). Therefore, our way to approach the evaluation of these IQBE MOEAs-GP is different than the usual evaluation in IRSs. The basic idea of this article is to perform comparisons among four different IQBE MOEA-GP approaches. We are not interested in comparing IQBE MOEA-GP versus IRS or versus test query relevance judgements. We are just focused on analyzing the performance of four IQBE MOEA-GP approaches. We use test queries just for grouping documents in predefined lists of relevant documents. So, an IQBE MOEA-GP *A* will be better than other IQBE MOEA-GP *B* if the lists of documents retrieved by *A* are closer to those predefined lists than those retrieved by *B*.

Instead of working with the complete test query set, we have selected a representative sample that allows observing the behavior of the studied IQBE MOEA-GP approaches. In this way, for example, if there are 29 relevant documents for Test Query 1 in the Cranfield collection, this test query will mimic a situation in which the user provides 29 relevant documents related to his or her information needs. The remaining 1,369 documents $(1398 - 29)$ will be considered as nonrelevant documents for the IQBE process (see Figure 11).

The studied IQBE MOEA-GP approaches generate a set of queries from the sets of relevant and nonrelevant documents. To do so, it is necessary to consider a sufficiently representative number of positive examples (relevant documents), so test queries with more relevant documents associated have been selected:

- Among the 225 test queries associated with the Cranfield collection, those test queries presenting 20 or more relevant documents have been taken into account. The seven resulting test queries (Queries 1, 2, 23, 73, 157, 220, and 225) have 29,

[3] Where *tam_max* is the maximun number of nodes per query (see Table 1).

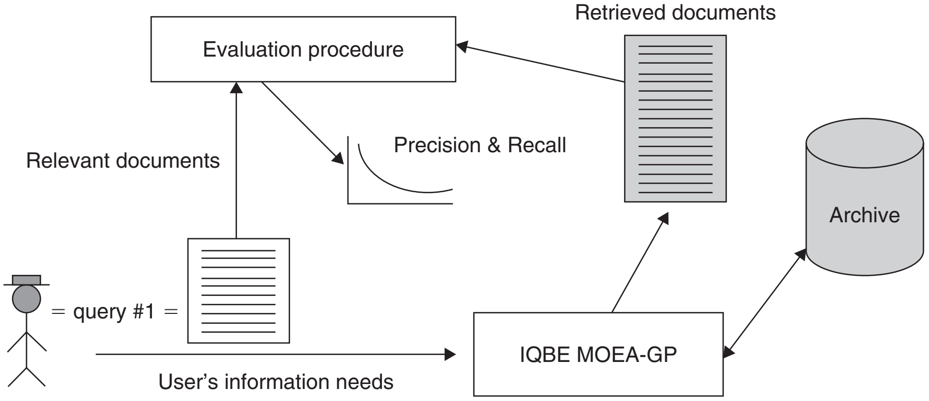


FIG. 11. Experimental framework.

TABLE 1. Common parameters.

Parameter	Value
Cross probability	0.8
Mutation probability	0.2
Maximum nodes per query	19
Population size (M)	800
Elite population	200

25, 33, 21, 40, 20, and 25, respectively, associated relevant documents.

- On the other hand, CACM provides 64 test queries, of which those that have 30 or more relevant documents have been selected. The eight resulting test queries (Queries 10, 14, 25, 26, 43, 58, 59, and 61) have 35, 44, 51, 30, 41, 30, 43, and 31, respectively, associated relevant documents.
- MEDLINE provides 30 test queries, of which those that have 30 or more relevant documents have been selected. The five resulting test queries (Queries 1, 20, 23, 28, and 29) have 37, 39, 39, 39, and 37, respectively, associated relevant documents.
- TREC-WSJ provides 67 test queries. Six queries with 130 or more relevant documents have been selected. The resulting test queries (Queries 2, 3, 8, 11, 21, and 22) have 226, 147, 138, 131, 138, and 141, respectively, associated relevant documents.

In this way, a wide range of simulated user's information needs have been developed, from Test Query 220 in Cranfield with 20 relevant documents to Test Query 2 in TREC-WSJ with 226 relevant documents.

The studied IQBE MOEAs-GP in this contribution have been run 30 times for each test query and collection (a total of 3,120 runs) with different initializations for each selected test query. The number of chromosome evaluations is 50,000 per run. We use a 2.0 GHz Pentium Core 2 Duo computer with 1 GB of RAM. The common parameter values considered are shown in Table 1.

Nonparametric Statistical Tests for Statistical Analysis

Nonparametric tests can be used for comparing the results of different EAs (García & Herrera, 2008; García et al., in press). Given that the nonparametric tests do not require explicit conditions for being conducted, it is recommended that the sample of results is obtained following the same criterion; that is, to compute the same aggregation (average, mode, etc.) over the same number of runs for each algorithm and problem.

In particular, we have considered two alternative methods based on nonparametric tests to analyze the experimental results:

1. Application of the Friedman's test and Holm's method as post hoc procedures (Demsar, 2006; García & Herrera, 2008; García et al., in press; Sheskin, 2003). The first test may be used to see whether there are significant statistical differences among the IQBE MOEAs-GP using the hypervolume measure. If differences are detected, then Holm's method is employed to compare the best IQBE MOEA-GP (control algorithm) against the remaining ones.
2. Use of the Wilcoxon matched-pairs signed-ranks test (for more detail, see Demsar, 2006; García et al., in press; Sheskin, 2003). With this test, the results of two IQBE MOEAs-GP, using the measure C, may be directly compared.

Experimental Results and Analysis

From each run, a pareto set is obtained, and a later filtration process is performed in the decision space to remove those queries that are identical. The two filtered pareto sets obtained by each run and test query are compared with the performance C measure and the hypervolume indicator. The number of different queries in the decision space also are analyzed. Finally, the time consumption analysis is presented.

C *measure.* In Table 2, we present the average results for the 30 runs of the C measure for each pair of IQBE MOEAs-GP and test query for the Cranfield, CACM, MEDLINE, and TREC-WSJ collections.

By analyzing Table 2, note that the best average results are offered by IQBE-NSGA-II-GP (see values in boldface).

We want to check if these results are statistically significant. The C measure is a performance measure which is

TABLE 2. Average results of the C measure for each test query in each pair of studied IQBE MOEAs-GP on the Cranfield, CACM, MEDLINE, and TREC-WSJ collections. X = IQBE-NSGA-II-GP, Y = IQBE-MOGA-GP, Z = IQBE-SPEA2-GP, and W = IQBE-SPEA-GP.

Query no. (collection)	*C*(X,Y)/*C*(Y,X)	*C*(X,Z)/*C*(Z,X)	*C*(X,W)/*C*(X,W)	*C*(Z,Y)/*C*(Y,Z)	*C*(Z,W)/*C*(W,Z)	*C*(W,Y)/*C*(Y,W)
1 (Cranfield)	**0.130**/0.786	**0.305**/0.915	**0.174**/0.919	0.537/0.714	0.153/0.680	0.593/0.594
2 (Cranfield)	**0.061**/0.859	**0.277**/0.963	**0.235**/0.967	0.553/0.671	0.160/0.701	0.674/0.594
23 (Cranfield)	**0.022**/0.878	**0.135**/0.918	**0.065**/0.910	0.364/0.711	0.196/0.671	0.457/0.591
73 (Cranfield)	**0.034**/0.875	**0.196**/0.932	**0.186**/0.936	0.447/0.662	0.195/0.727	0.568/0.616
157 (Cranfield)	**0.030**/0.868	**0.097**/0.910	**0.060**/0.923	0.453/0.694	0.162/0.715	0.517/0.608
220 (Cranfield)	**0.077**/0.795	**0.367**/0.913	**0.278**/0.932	0.512/0.687	0.170/0.640	0.557/0.611
225 (Cranfield)	**0.075**/0.871	**0.273**/0.950	**0.199**/0.971	0.473/0.688	0.145/0.701	0.638/0.603
10 (CACM)	**0.050**/0.838	**0.315**/0.913	**0.132**/0.939	0.653/0.759	0.166/0.709	0.836/0.656
14 (CACM)	**0.053**/0.770	**0.305**/0.918	**0.260**/0.958	0.492/0.751	0.136/0.710	0.725/0.702
25 (CACM)	**0.103**/0.825	**0.265**/0.850	**0.130**/0.855	0.709/0.751	0.120/0.728	0.729/0.666
26 (CACM)	*0.112*/0.825	**0.272**/0.933	**0.120**/0.941	0.531/0.760	0.107/0.727	0.739/0.648
43 (CACM)	*0.106*/0.812	**0.339**/0.896	**0.162**/0.958	0.625/0.814	0.125/0.778	0.683/0.751
58 (CACM)	*0.114*/0.764	**0.461**/0.887	**0.300**/0.894	0.636/0.760	0.158/0.653	0.769/0.659
59 (CACM)	*0.078*/0.831	**0.317**/0.923	**0.115**/0.936	0.542/0.799	0.145/0.663	0.664/0.716
61 (CACM)	*0.106*/0.831	**0.305**/0.922	**0.128**/0.935	0.564/0.773	0.155/0.729	0.674/0.648
1 (MEDLINE)	*0.208*/0.604	**0.246**/0.903	**0.229**/0.899	0.721/0.414	0.240/0.729	0.807/0.364
20 (MEDLINE)	*0.078*/0.744	**0.091**/0.815	**0.134**/0.879	0.477/0.595	0.300/0.740	0.684/0.579
23 (MEDLINE)	**0.211**/0.654	**0.151**/0.917	**0.120**/0.877	0.638/0.439	0.322/0.735	0.763/0.401
28 (MEDLINE)	*0.151*/0.713	**0.102**/0.925	**0.100**/0.923	0.542/0.561	0.182/0.756	0.691/0.541
29 (MEDLINE)	*0.171*/0.769	**0.070**/0.912	**0.053**/0.924	0.573/0.583	0.217/0.781	0.764/0.557
2 (TREC-WSJ)	*0.059*/0.859	**0.138**/0.862	**0.061**/0.907	0.457/0.789	0.138/0.862	0.568/0.640
3 (TREC-WSJ)	*0.076*/0.861	**0.143**/0.913	**0.094**/0.927	0.502/0.753	0.143/0.913	0.496/0.661
8 (TREC-WSJ)	*0.056*/0.887	**0.129**/0.879	**0.049**/0.909	0.517/0.776	0.129/0.879	0.566/0.693
11 (TREC-WSJ)	*0.080*/0.870	**0.086**/0.946	**0.020**/0.939	0.486/0.771	0.086/0.946	0.568/0.627
21 (TREC-WSJ)	*0.041*/0.839	**0.124**/0.842	**0.083**/0.869	0.499/0.753	0.124/0.842	0.519/0.667
22 (TREC-WSJ)	*0.069*/0.811	**0.229**/0.906	**0.117**/0.934	0.438/0.755	0.229/0.906	0.563/0.647

defined to compare two algorithms using the dominance concept between two paretos. To statistically compare the results between two IQBE MOEAs-GP and to determine which one is the best, we can perform the Wilcoxon signed-rank test for detecting differences in both means.

Table 3 summarizes the results of this procedure. The structure of the table presents $N_{alg} \times (N_{alg} + 2)$ cells to compare all the IQBE MOEAs-GP in it, with N_{alg} being the number of IQBE MOEAs-GP studied. In each of the $N_{alg} \times N_{alg}$ cells, three symbols can appear: $+$, $-$, or $=$. They show that the IQBE MOEA-GP situated in that row is better ($-$), worse ($+$), or equal ($=$) in behavior (using the C measure) to the IQBE MOEA-GP that appears in the column. The penultimate column ($\geq$) represents the number of IQBE MOEAs-GP with worse or equal behavior to the one that appears in the row (without considering the IQBE MOEA-GP itself), and the last column ($>$) represents the number of IQBE MOEAs-GP with worse behavior than the one that appears in the row.

For more detail, in the Appendix, six Wilcoxon test tables are shown. In these tables, R^+, R^-, and the critical value for $N = 26$ are presented for each pair of IQBE MOEAs-GP. The lowest values, which correspond with the best results, are in boldface.

From Table 3 and the tables in the Appendix, we clearly see that IQBE-NSGA-II-GP obtains better results than do other IQBE MOEAs-GP (i.e., the R^- values are lower than the R^+ ones). In addition, the statistical test indicates that these results are statistically significant considering the C measure (because these R^- values are lower than the critical values) with a significance level of $\alpha = 0.05$.

TABLE 3. Wilcoxon tests table for IQBE MOEAs-GP considering the C measure. X = IQBE-NSGA-II-GP, W = IQBE-SPEA-GP, Z = IQBE-SPEA2-GP, Y = IQBE-MOGA-GP.

	Y	Z	W	X	≥	>
X	−	−	−		0	0
W	=	+		+	3	2
Z	+		−	+	2	2
Y		−	=	+	2	1

Hypervolume indicator. In Table 4, we present the average results for the 30 runs of the hypervolume indicator for each of IQBE MOEAs-GP and test query for the Cranfield, CACM, MEDLINE, and TREC-WSJ collections.

Results in Table 4 show that the best average results, using the hypervolume indicator, are offered by IQBE-NSGA-II-GP (see boldface values).

We want to check if these results are statistically significant. To compare these results, we will use a multiple comparison test to find the best IQBE MOEA-GP. In a multiple comparison test, first it is necessary to check (using a test such as Friedman's) whether all the results obtained by the algorithms present any inequality. In the case of finding inequality, then we can know, by using a post hoc test

DOI: 10.1002/asi

TABLE 4. Average results of the hypervolume indicator for each test query and the four studied IQBE MOEAs-GP on the Cranfield, CACM, MEDLINE, and TREC-WSJ collections.

Query no. (collection)	IQBE-NSGA-II-GP	IQBE-SPEA2-GP	IQBE-SPEA-GP	IQBE-MOGA-GP
1 (Cranfield)	**0, 329**	0, 102	0, 080	0, 228
2 (Cranfield)	**0, 419**	0, 116	0, 084	0, 279
23 (Cranfield)	**0, 378**	0, 120	0, 087	0, 178
73 (Cranfield)	**0, 484**	0, 138	0, 094	0, 267
157 (Cranfield)	**0, 307**	0, 098	0, 074	0, 184
220 (Cranfield)	**0, 456**	0, 115	0, 094	0, 248
225 (Cranfield)	**0, 407**	0, 108	0, 084	0, 236
10 (CACM)	**0, 304**	0, 082	0, 059	0, 249
14 (CACM)	**0, 295**	0, 085	0, 057	0, 221
25 (CACM)	**0, 233**	0, 082	0, 058	0, 189
26 (CACM)	**0, 360**	0, 098	0, 065	0, 285
43 (CACM)	**0, 276**	0, 079	0, 057	0, 238
58 (CACM)	**0, 289**	0, 080	0, 066	0, 246
59 (CACM)	**0, 248**	0, 075	0, 056	0, 220
61 (CACM)	**0, 334**	0, 089	0, 064	0, 243
1 (MEDLINE)	**0, 691**	0, 361	0, 155	0, 643
20 (MEDLINE)	**0, 676**	0, 320	0, 209	0, 411
23 (MEDLINE)	**0, 756**	0, 398	0, 197	0, 656
28 (MEDLINE)	**0, 792**	0, 333	0, 189	0, 571
29 (MEDLINE)	**0, 642**	0, 244	0, 124	0, 463
2 (TREC-WSJ)	**0, 113**	0, 065	0, 057	0, 089
3 (TREC-WSJ)	**0, 112**	0, 049	0, 044	0, 080
8 (TREC-WSJ)	**0, 122**	0, 049	0, 041	0, 081
11 (TREC-WSJ)	**0, 122**	0, 046	0, 040	0, 085
21 (TREC-WSJ)	**0, 112**	0, 050	0, 043	0, 084
22 (TREC-WSJ)	**0, 114**	0, 048	0, 042	0, 083

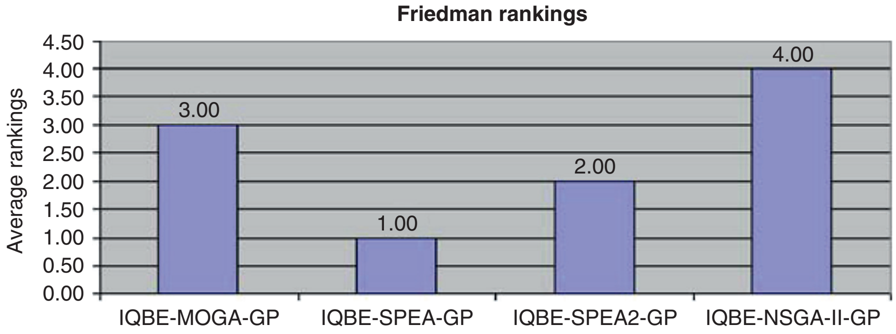


FIG. 12. Friedman rankings for the studied IQBE MOEAs-GP.

(e.g., Holm’s), which algorithm partners’ average results are dissimilar. Next, we describe the procedure used.

Friedman’s test [with Friedman value: **78.00**, $\chi^2(7.81)$, $p < .0001$] was applied to the data in Table 4 to see if there were differences in the results (using the hypervolume indicator). A χ^2 distribution with 3 *df* for $N_{ds} = 26$ was used. We emphasize in boldface the highest value between the two values being compared, and as the smallest in both cases corresponds to the value given by the statistic, it informs us of the rejection of the null hypothesis; in this manner, Friedman’s test shows the existence of significant differences among the observed results in all test queries. Given that the *p* value of Friedman’s test is lower than the level of significance considered $\alpha = 0.05$, there are significant differences among the observed results. Attending to these results, a post hoc statistical analysis could help to detect specific differences among algorithms.

In Figure 12, the values of the average rankings using Friedman’s test are drawn. Each column represents the average ranking obtained by an IQBE MOEA-GP approach; that is, if a certain IQBE MOEA-GP achieves rankings 1, 3, 1, 4, and 2 on five datasets, the average ranking is $\frac{1+3+1+4+2}{5} = \frac{11}{5}$. The height of each column is proportional to the ranking; the higher a column, the better its associated algorithm.

We now apply Holm’s test to compare the best ranking IQBE MOEA-GP with the remaining IQBE MOEAs-GP. In order to show the results of this test, we will present the table

TABLE 5. Holm's table for IQBE-NSGA-II-GP as control IQBE MOEA-GP.

I	IQBE MOEA-GP	Z	p	α/i	Hypothesis
3	IQBE-SPEA-GP	8,378544	0,000000	0,016667	R for IQBE-NSGA-II-GP
2	IQBE-SPEA2-GP	5,585696	0,000001	0,025000	R for IQBE-NSGA-II-GP
1	IQBE-MOGA-GP	2,792848	0,005225	0,050000	R for IQBE-NSGA-II-GP

TABLE 6. Average number of different queries (in the decision space) for each test query and the four studied IQBE MOEAs-GP on the Cranfield, CACM, MEDLINE, and TREC-WSJ collections.

Query no. (collection)	IQBE-NSGA-II-GP	IQBE-SPEA2-GP	IQBE-SPEA-GP	IQBE-MOGA-GP
1 (Cranfield)	**302, 33**	20, 77	14, 40	106,10
2 (Cranfield)	**299, 37**	19, 53	16, 33	97,90
23 (Cranfield)	**236, 80**	20, 20	14, 47	85,70
73 (Cranfield)	**270, 03**	14, 57	13, 30	173,57
157 (Cranfield)	**255, 57**	17, 03	15, 97	113,50
220 (Cranfield)	**242, 33**	17, 90	12, 90	151,13
225 (Cranfield)	**258, 53**	19, 00	15, 97	83,63
10 (CACM)	**335, 33**	20, 93	14, 13	46, 03
14 (CACM)	**397, 50**	20, 20	18, 60	26, 47
25 (CACM)	**370, 93**	22, 87	16, 60	27, 73
26 (CACM)	**282, 10**	19, 53	14, 43	51, 37
43 (CACM)	**344, 57**	23, 37	14, 57	27, 97
58 (CACM)	**311, 20**	21, 10	12, 47	75, 20
59 (CACM)	**330, 33**	22, 57	17, 43	24, 60
61 (CACM)	**280, 17**	22, 37	14, 27	81, 13
1 (MEDLINE)	**377, 57**	10, 03	17, 20	133, 57
20 (MEDLINE)	**328, 97**	9, 47	13, 63	38, 33
23 (MEDLINE)	**349, 80**	12, 37	14, 47	32, 87
28 (MEDLINE)	**275, 70**	10, 53	14, 87	26, 90
29 (MEDLINE)	**237, 97**	12, 07	16, 10	31, 77
2 (TREC-WSJ)	**308, 37**	25, 63	20, 00	25, 20
3 (TREC-WSJ)	**288, 70**	27, 07	19, 37	33, 37
8 (TREC-WSJ)	**306, 87**	23, 77	19, 00	36, 03
11 (TREC-WSJ)	**263, 17**	21, 63	17, 47	60, 17
21 (TREC-WSJ)	**292, 10**	24, 60	18, 37	27, 97
22 (TREC-WSJ)	**343, 83**	27, 23	19, 70	34, 03

associated with Holm's procedure, in which all the computations are shown. In this table (Table 5) the IQBE MOEAs-GP are ordered with respect to the z value obtained. Thus, by using the normal distribution, we can obtain the corresponding p-value associated with it and this can be compared with the associated α/i in the same row of the table to show whether the associated hypothesis of equal behavior is rejected in favor of the best ranking IQBE MOEA-GP (marked with an R) or not (marked with an A).

The tests reject the hypothesis of equality of means for the tree IQBE MOEAs-GP. Holm's method allow us to point out that IQBE-NSGA-II-GP is better (using the hypervolume indicator) than IQBE-SPEA2-GP, IQBE-SPEA-GP and IQBE-MOGA-GP with $\alpha = 0.05$.

Number of queries. In Table 6, we present the average number of different queries (in the decision space) for the 30 runs for each of IQBE MOEAs-GP and test query for the Cranfield, CACM, MEDLINE, and TREC-WSJ collections.

Results in Table 6 show that IQBE-NSGA-II-GP is the IQBE MOEA-GP approach that learns a larger number of different queries, in the decision space, than did the other IQBE MOEAs-GP (see boldface values).

As in the previous subsection, we want to check if these results are statistically significant. To compare these results, we will use the Friedman's test to check whether all the results obtained by the algorithms present any inequality. In the case of finding it, we will use Holm's test to know which algorithms partners average results are dissimilar. In the following, we describe the used procedure.

Friedman's test (with Friedman value: **71.769**, χ^2(7.81, $p < .0000$) was applied to the data in Table 6 to see if there are differences in the results (using the hypervolume indicator). A χ^2 distribution with 3 df for $N_{ds} = 26$ was used. We emphasize in boldface the highest value between the two values that are being compared, and as the smallest in both cases corresponds to the value given by the statistic, it informs us of the rejection of the null hypothesis; in this manner, Friedman's test shows the existence of significant differences among the observed results in all test queries. Given that the p value of Friedman's test is lower than the level of significance considered $\alpha = 0.05$, there are significant differences among the

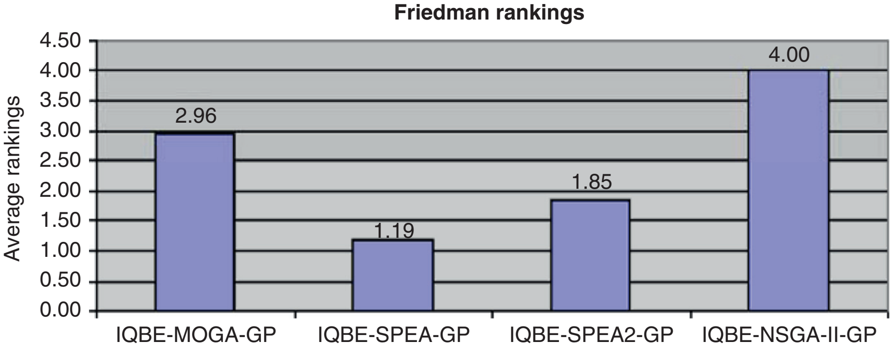


FIG. 13. Friedman rankings for the studied IQBE MOEAs-GP (considering the number of different queries).

TABLE 7. Holm's table for IQBE-NSGA-II-GP as control IQBE MOEA-GP.

I	IQBE MOEA-GP	Z	p	α/i	Hypothesis
3	IQBE-SPEA-GP	6.736097	0.000000	0.016667	R for IQBE-NSGA-II-GP
2	IQBE-SPEA2-GP	5.511352	0.000000	0.025000	R for IQBE-NSGA-II-GP
1	IQBE-MOGA-GP	2.449490	0.014306	0.050000	R for IQBE-NSGA-II-GP

observed results. Attending to these results, a post hoc statistical analysis could help detect specific differences among algorithms.

In Figure 13, the values of the average rankings using Friedman's test are drawn.

We now apply Holm's method to compare the best ranking IQBE MOEA-GP with the remaining IQBE MOEAs-GP. To show the results of this test, we will present the table associated with Holm's procedure, in which all the computations are shown. In this table (Table 7), the IQBE MOEAs-GP are ordered with respect to the z value obtained. Thus, by using the normal distribution, we can obtain the corresponding p value associated with it, which can be compared with the associated α/i in the same row of the table to show whether the associated hypothesis of equal behavior is rejected in favor of the best-ranking IQBE MOEA-GP (marked with an *R*) or not (marked with an *A*).

The tests reject the hypothesis of equality of means for the tree IQBE MOEAs-GP. Holm's method allows us to show that IQBE-NSGA-II-GP is the IQBE MOEA-GP that learns more different queries in the decision space, compared to IQBE-SPEA2-GP, IQBE-SPEA-GP, and IQBE-MOGA-GP with $\alpha = 0.05$.

Time-consumption analysis. In Table 8, the spent time average for each IQBE-MOEA-GP on each test query is presented.

Results suggest that IQBE-NSGA-II-GP is the fastest IQBE-MOEA-GP on small and big collections (Cranfield and TREC-WSJ with 3,857 and 28,904 index terms, respectively); on medium test collections (CACM and MEDLINE with 7,562 and 7,170 index terms, respectively), IQBE-NSGA-II-GP has a very good performance (i.e., low time consumption), with IQBE-MOGA-GP the fastest on MEDLINE and IQBE-SPEA-GP faster on CACM.

Summary analysis. The experimental results show that IQBE-NSGA-II-GP is the IQBE MOEA-GP approach that achieves the best performance; that is, it achieves better nondominated solutions sets, using the *C* measure, the hypervolume indicator, and time consumption (see boldface values in Tables 2, 4, and 8), in the process of learning Boolean queries than other studied IQBE MOEAs-GP.

Results also show that IQBE-NSGA-II-GP is the IQBE MOEA-GP approach that achieves more different queries, in both the decision and the objective space for a unique run, compared to the other studied IQBE MOEAs-GP.

All these results also were statistically supported using nonparametric statistical tests.

Finally, we compared the best IQBE MOEA-GP approach, IQBE-NSGA-II-GP, with the other IQBE MOEAs-GP and with the classic Boolean IQBE-GP approach. In Figure 14, we graphically presented the queries achieved[4] in a unique run, by IQBE-NSGA-II-GP, IQBE-MOGA-GP, IQBE-SPEA-GP, IQBE-SPEA2-GP, and the Smith and Smith (1997) Boolean IQBE-GP proposal. In Figure 14, one can see that IQBE-NSGA-II-GP gets the best set of solutions, with a good precision–recall trade-off. IQBE-NSGA-II-GP is the IQBE MOEA-GP that covers more objective space. It also improves the performance of the Smith and Smith Boolean IQBE-GP proposal, whose unique learned query is overcome (in both precision and recall performance criteria) by all queries learned by the IQBE-NSGA-II-GP.

From all these results, we can conclude that IQBE-NSGA-II-GP is the best IQBE-GP approach for learning Boolean queries.

[4] The results correspond to Test Query 1 of the Cranfield collection.

DOI: 10.1002/asi

TABLE 8. Average time consumption (in seconds) for each IQBE-MOEA-GP on each test query.

Query no. (collection)	IQBE-NSGA-II-GP	IQBE-SPEA2-GP	IQBE-SPEA-GP	IQBE-MOGA-GP
1 (Cranfield)	**14, 93**	288, 57	20, 80	16, 30
2 (Cranfield)	**16, 10**	330, 43	28, 83	16, 90
23 (Cranfield)	**14, 77**	287, 93	26, 37	16, 17
73 (Cranfield)	**14, 77**	288, 07	20, 23	15, 43
157 (Cranfield)	**14, 80**	287, 33	25, 67	16, 00
220 (Cranfield)	**14, 17**	287, 93	24, 93	15, 73
225 (Cranfield)	**14, 23**	289, 53	20, 33	15, 57
10 (CACM)	134, 33	416, 00	**101, 53**	132, 57
14 (CACM)	144, 63	415, 63	**108, 30**	141, 30
25 (CACM)	134, 33	415, 43	**101, 43**	131, 77
26 (CACM)	134, 97	414, 90	**102, 27**	131, 87
43 (CACM)	134, 23	416, 10	**101, 43**	131, 47
58 (CACM)	132, 27	417, 73	**103, 73**	130, 90
59 (CACM)	134, 03	419, 20	**102, 80**	131, 63
61 (CACM)	133, 13	419, 70	**103, 97**	131, 03
1 (MEDLINE)	13, 47	283, 00	19, 60	**12, 27**
20 (MEDLINE)	13, 70	283, 00	19, 33	**13, 23**
23 (MEDLINE)	13, 27	282, 50	19, 67	**13, 13**
28 (MEDLINE)	13, 87	283, 17	19, 07	**13, 07**
29 (MEDLINE)	**11, 60**	282, 70	19, 43	13, 23
2 (TREC-WSJ)	**305, 30**	558, 77	306 ,70	306, 70
3 (TREC-WSJ)	**304, 17**	562, 03	305 ,87	305, 87
8 (TREC-WSJ)	**304, 77**	606, 87	307 ,13	307, 13
11 (TREC-WSJ)	306, 30	560, 07	**305, 93**	**305, 93**
21 (TREC-WSJ)	**305, 13**	560, 83	305, 77	305, 77
22 (TREC-WSJ)	**304, 67**	560, 87	305, 73	305, 73

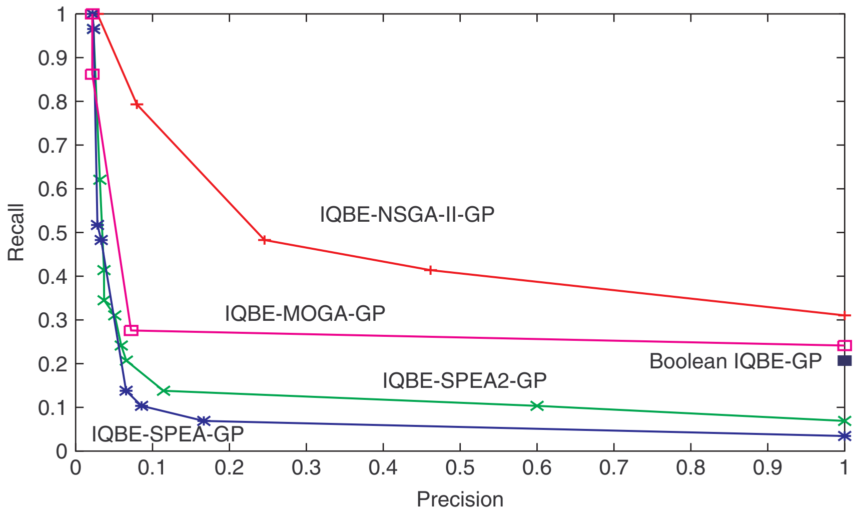


FIG. 14. Pareto sets achieved by IQBE-NSGA-II-GP, IQBE-MOGA-GP, IQBE-SPEA-GP, IQBE-SPEA2-GP, and Smith and Smith's (1997) Boolean IQBE-GP for the Test Query 1 on the Cranfield collection.

In addition, we can say that the use of MOEAs improves the IQBE process by providing more queries in a unique run. Several queries with different precision–recall trade-offs can be given to user. This fact is graphically shown in Figure 14, where:

- The two best IQBE MOEAs-GP, IQBE-NSGA-II-GP and IQBE-MOGA-GP, always get queries with better performance, in both criteria, than the one learned by the classic non-multi-objective Boolean IQBE-GP approach; and
- the two "worst" IQBE MOEAs-GP, IQBE-SPEA-GP and IQBE-SPEA2-GP, learn several queries. Some of these queries have a worse performance (in both criteria) than did the unique query learned by the classic Boolean IQBE-GP approach, but other queries get good results in recall. Therefore, we cannot say that IQBE-SPEA-GP and

IQBE-SPEA2-GP are worse than the classic Boolean IQBE-GP approach, which only gets a unique query with "good" precision and poor recall.

Conclusions

In this contribution, an analysis of performance in the Boolean IRSs context of four of the most currently used MOEAs in the specialized literature was performed. The studied MOEAs were applied on the automatic learning of Boolean queries adapted to use GP components. All of them extend the Smith and Smith (1997) Boolean IQBE-GP proposal to work in the multi-objective context.

The experimental results and the statistical analysis showed that NSGA-II, with GP (IQBE-NSGA-II-GP), is the best IQBE MOEA-GP approach, considering the *C* measure, the hypervolume indicator, and time consumption. That is, IQBE-NSGA-II-GP obtained the best set of solutions with a good precision–recall trade-off for each test query.

IQBE-NSGA-II-GP is also the IQBE MOEA-GP that achieved a larger set of different queries in both the decision and the objective space. It also improved the performance of the Smith and Smith (1997) Boolean IQBE-GP proposal learning more than one query in a unique run.

Finally, with this study, the benefits of using MOEAs in the IQBE process also were proven.

Acknowledgment

This work was supported by the projects FUZZY-LING, Cod. TIN2007-61079 and Proyecto de Excelencia de la Junta de Andalucia SAINFOWEB, Cod. 00602.

References

Abdelmgeid, A.A. (2007). Applying genetic algorithm in query improvement problem. International Journal of Information Technologies and Knowledge, 1, 309–316.

Baeza-Yates, R., & Ribeiro-Neto, B. (1999). Modern information retrieval. Reading, MA: Addison-Wesley.

Chen, H., Shankaranarayanan, G., She, L., & Iyer, A. (1998). A machine learning approach to inductive query by example: An experiment using relevance feedback, ID3, genetic algorithms, and simulated annealing. Journal of the American Society for Information Science, 49(8), 693–705.

Coello, C.A., Van Veldhuizen, D.A., & Lamant, G.B. (2002). Evolutionary algorithms for solving multi-objective problems. Dordrecht, The Netherlands: Kluwer.

Cordón, O., Herrera-Viedma, E., López-Pujalte, C., Luque, M., & Zarco, C. (2003). A review on the application of evolutionary computation to information retrieval. International Journal of Approximate Reasoning, 34, 241–264.

Cordón, O., Herrera-Viedma, E., & Luque, M. (2002). Evolutionary learning of boolean queries by multiobjective genetic programming. Proceedings of PPSN-VII (pp. 710–719), Granada, Spain, LNCS 2439. Berlin/Heidelberg, Germany: Springer.

Cordón, O., Herrera-Viedma, E., & Luque, M. (2006a). Improving the learning of boolean queries by means of a multiobjective IQBE evolutionary algorithm. Information Processing & Management, 42(3), 615–632.

Cordón, O., Herrera-Viedma, E., & Luque, M. (2006b). Multi-objective machine learning. Studies in computational intelligence series. In Y. Jin (Ed.), Multiobjective genetic algorithm for linguistic persistent query learning in text retrieval (pp. 601–627). Berlin/Heidelberg, Germany: Springer.

Cordón, O., Herrera-Viedma, E., Luque, M., Moya, F., & Zarco, C. (2003). Analyzing the performance of a multiobjective GA-P algorithm for learning fuzzy queries in a machine learning environment. Proceedings of the International Fuzzy Systems Association World Congress 3 (pp. 611–619), Istanbul, Turkey, LNAI 2715.

Cordón, O., Moya, F., & Zarco, C. (2000). A GA-P algorithm to automatically formulate extended boolean queries for a fuzzy information retrieval system. Mathware & Soft Computing, 7(2–3), 309–322.

Cordón, O., Moya, F., & Zarco, C. (2004). Fuzzy logic and multiobjective evolutionary algorithms as soft computing tools for persistent query learning in text retrieval environments. Proceedings of the IEEE International Conference on Fuzzy Systems (pp. 571–576), Budapest, Hungary.

Deb, K. (2001). Multi-objective optimization using evolutionary algorithms. New York: Wiley.

Deb, K., Pratap, A., Agrawal, S., & Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Transactions on Evolutionary Computation, 6, 182–197.

Demsar, J. (2006). Statistical comparisons of classifiers over multiple data sets. Journal of Machine Learning Research, 7, 1–30.

Fonseca, C.M., & Fleming, P.J. (1993). Genetic algorithms for multiobjective optimization: Formulation, discussion and generalition. Proceedings of the 5th International Conference on Genetic Algorithms (pp. 416–423), San Mateo, CA.

García, S., & Herrera, F. (2008). An extension on "Statistical comparisons of classifiers over multiple data sets" for all pairwise comparisons. Journal of Machine Learning Research, 9, 2677–2694.

García, S., Molina, D., Lozano, M., & Herrera, F. (in press). A study on the use of non-parametric tests for analyzing the evolutionary algorithms' behaviour: A case study on the CEC'2005 special session on real parameter optimization. Journal of Heuristics. DOI: 10.1007/s10732–008–9080–4.

Goldberg, D.E. (1989). Genetic algorithms in search, optimization and machine learning. Reading, MA: Addison-Wesley.

Herrera-Viedma, E., Cordón, O., Luque, M., López, A.G., & Muñoz, A.M. (2003). A model of fuzzy linguistic IRS based on multi-granular linguistic information. International Journal of Approximate Reasoning, 34, 221–239.

Howard, L., & D'Angelo, D. (1995). The GA-P: A genetic algorithm and genetic programming hybrid. IEEE Expert, 10(3), 11–15.

Knowles, J., Thiele, L., & Zitzler, E. (2006). A tutorial on the performance assessment of stochastic multiobjective optimizers (Rev. ed.), Tech. Rep. No. 214). Zurich, Switzerland: ETH, Computer Engineering and Networks Laboratory (TIK).

Koza, J. (1992). Genetic programming: On the programming of computers by means of natural selection. Cambridge, MA: MIT Press.

Kraft, D.H., Petry, F.E., Buckles, B.P., & Sadasivan, T. (1997). Genetic algorithms and fuzzy logic systems. In E. Sanchez, T. Shibata, & L.A. Zadeh (Eds.), Genetic algorithms for query optimization in information retrieval: Relevance feedback (pp. 155–173). Hackensack, NJ: World Scientific.

Michalewicz, Z. (1996). Genetic algorityms + data structures = evolution programs. Springer.

Morse, J.N. (1980). Reducing the size of the nondominated set: Pruning by clustering. Computers and Operations Research, 7(1–2), 55–66.

Salton, G. (1989). Automatic text processing: The transformation, analysis and retrieval of information by computer. Reading, MA: Addison-Wesley.

Salton, G. (1997). The SMART retrieval system. Experiments in automatic document processing. Englewood Cliffs, NJ: Prentice-Hall.

Salton, G., & Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. Information Processing & Management, 24, 513–523.

Salton, G., & McGill, M.J. (1983). An introduction to modern information retrieval. New York: McGraw-Hill.

Sheskin, D.J. (2003). Handbook of parametric and nonparametric statistical procedures. Boca Raton, FL: CRC Press.

Smith, M.P., & Smith, M. (1997). The use of genetic programming to build boolean queries for text retrieval through relevance feedback. Journal of Information Science, 23(6), 423–431.

van Rijsbergen, C.J. (1979). Information retrieval. London: Butterworth.

Zitzler, E., Deb, K., & Thiele, L. (2000). Comparison of multiobjective evolutionary algorithms: Empirical results. Evolutionary Computation, 8(2), 173–271.

Zitzler, E., & Thiele, L. (1998). Multiobjective optimization using evolutionary algorithms: A comparative case study. In A.E. Eiben, T. Bäck, M. Schoenauer, & H.P. Schwefel (Eds.), Proceedings of the 5th International Conference on Parallel Problem Solving from Nature (pp. 292–301), Berlin, Germany: Springer.

Zitzler, E., & Thiele, L. (1999). Multiobjective evolutionary algorithms: A comparative case study and the strength pareto approach. IEEE Transactions on Evolutionary Computation, 3(4), 257–271.

Zitzler, E., Laumanns, M., & Thiele, L. (2002). SPEA2: Improving the strength pareto evolutionary algorithm for multiobjective optimization. In K. Giannakoglou, D. Tsahalis, J. Periaux, K. Papailiou, & T. Fogarty (Eds.), Evolutionary methods for design, optimization and control with application to industrial problems (pp. 95–100). Barcelona, Spain: International Center for Numerical Methods in Engineering.

Appendix

Wilcoxon's Test Tables 9–14.

TABLE 9. IQBE-NSGA-II-GP versus IQBE-MOGA-GP using Wilcoxon's tests (ps = .05 and .1).

R^+ IQBE-MOGA-GP	R^- IQBE-NSGA-II-GP	Critical value ($p = .05/p = .1$)	Significant differences? ($p = .05/p = .1$)
351	**0**	98/110	Yes/Yes

TABLE 10. IQBE-NSGA-II-GP versus IQBE-SPEA2-GP using Wilcoxon's tests (ps = .05 and .1).

R^+ IQBE-SPEA2-GP	R^- IQBE-NSGA-II-GP	Critical value ($p = .05/p = .1$)	Significant differences? ($p = .05/p = .1$)
351	**0**	98/110	Yes/Yes

TABLE 11. IQBE-NSGA-II-GP versus IQBE-SPEA-GP using Wilcoxon's tests (ps = .05 and .1).

R^+ IQBE-SPEA-GP	R^- IQBE-NSGA-II-GP	Critical value ($p = .05/p = .1$)	Significant differences? ($p = .05/p = .1$)
351	**0**	98/110	Yes/Yes

TABLE 12. IQBE-SPEA2-GP versus IQBE-MOGA-GP using Wilcoxon's tests (ps = .05 and .1).

R^+ IQBE-SPEA2-GP	R^- IQBE-MOGA-GP	Critical value ($p = .05/p = .1$)	Significant differences? ($p = .05/p = .1$)
317	**34**	98/110	Yes/Yes

TABLE 13. IQBE-SPEA2-GP versus IQBE-SPEA-GP using Wilcoxon's tests (ps = .05 and .1).

R^+ IQBE-SPEA-GP	R^- IQBE-SPEA2-GP	Critical value ($p = .05/p = .1$)	Significant differences? ($p = .05/p = .1$)
351	**0**	98/110	Yes/Yes

TABLE 14. IQBE-SPEA-GP versus IQBE-MOGA-GP using Wilcoxon's tests (ps = .05 and .1).

R^+ IQBE-SPEA-GP	R^- IQBE-MOGA-GP	Critical value ($p = .05/p = .1$)	Significant differences? ($p = .05/p = .1$)
154, 5	196, 5	98/110	No/No

DOI: 10.1002/asi