International Journal of Pattern Recognition
and Artificial Intelligence
Vol. 23, No. 6 (2009) 1107–1127

World Scientific
www.worldscientific.com

SOLUTION OVER-FIT CONTROL IN EVOLUTIONARY MULTIOBJECTIVE OPTIMIZATION OF PATTERN CLASSIFICATION SYSTEMS

PAULO V. W. RADTKE[*,†,‡], TONY WONG[*,§]
and ROBERT SABOURIN[*,¶]

[*]*Département de génie de la production automatisée*
École de technologie supérieure
Montreal, Québec, Canada

[†]*Universidade Federal do Paraná, Setor Escola Técnica,*
Curitiba, Paraná, Brazil
[‡]*radtke@livia.etsmtl.ca*
[§]*Tony.Wong@etsmtl.ca*
[¶]*Robert.Sabourin@etsmtl.ca*

The optimization of many engineering systems is challenged by the solution over-fit to the data set used to evaluate potential solutions during the evolutionary process. The solution over-fit phenomenon is hard to detect and is especially prevalent in problems involving example-based training, such as pattern feature selection and pattern classifier design. For these applications, uncontrolled over-fit can lead to biased features being extracted and degraded classifier generalization abilities. This paper details the performance of a solution over-fit control strategy used in the multiobjective evolutionary optimization of a multileveled classification system. This control, embedded within a solution validation procedure, minimizes the over-fit effects without modifying the dominance relation used in the processing of candidate solutions. Extensive experimental analysis using multiobjective genetic and memetic algorithms demonstrates both the need and the efficiency of the proposed over-fit control for pattern classification systems optimization.

Keywords: Multiobjective optimization; over-fit control; feature selection; classifier design; pattern classification system.

1. Introduction

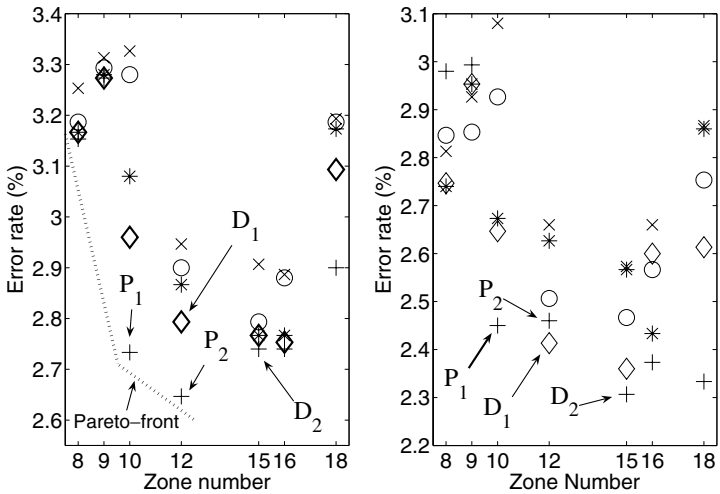
Classification system optimization using evolutionary algorithms is a trend observed in the literature.[19,28] A population-based approach is preferable to traditional single solution methods because of its robustness and its ability to avoid local optimal solutions. The Pareto-based approach used by multiobjective optimization algorithms allows the discovery of multiple tradeoffs between objective functions. Such algorithms apply a dominance relation, which states that a solution x is better than another solution y only if x is no worse than y in all objective functions, and x is

better than y in at least one objective function. If this relation is not verified, then both solutions are said to be nondominant to each other. The goal of multiobjective optimization is to find the best set of nondominated solutions, the Pareto front.[5]

From a pattern recognition perspective, modeling the pattern classification systems design as a multiobjective evolutionary optimization problem is most interesting because these systems are usually circumscribed by two competing factors — system complexity and classification accuracy while using a wrapper approach to evaluate classifier accuracy. In this context, the ultimate goal is to obtain the highest possible recognition rate while keeping the overall system complexity as low as possible. However, solution over-fit occurs when classification systems are optimized using a wrapper approach. The optimization process becomes a learning process, searching for solutions based on the wrapped classifier accuracy on the optimization data set. Again, solutions found at the end of the optimization process might be over-fitted to the optimization data. This effect occurs even when a validation procedure is used to train the wrapped classifier. Reunanen *et al.* demonstrated solution over-fit when comparing sequential and floating feature subset selection algorithms.[27] Loughrey and Cunningham discussed an early stopping approach to remove over-fit from solutions optimized by genetic algorithms in Ref. 22 and simulated annealing in Ref. 23. Llorà *et al.* used, in Ref. 21, an approach to analyze the optimized Pareto front generated by a multi-objective genetic algorithm (MOGA) to remove over-fit. A similar approach was also used by Mansilla *et al.* in Ref. 2.

Thus, evolutionary algorithms cannot, on its own, alleviate the solution over-fit problem. A similar problem is solved during example-based classifier training, and the same approach can be used to optimize a classification system using a wrapper approach. In a typical classifier training procedure, there are a number of training iterations where classifier parameters are adjusted based on the current accuracy obtained from the classification of the training data set. Normally, the classifier improves its accuracy at each training iteration. However, after a certain iteration, the classifier will start to memorize the training data instead of producing a more general model of the data. At this point, the classifier is said to be over-fitted to its training data set. Thus, the classifier training problem is to determine the termination iteration at which the training procedure stops. This can be achieved through a validation procedure by using a validation data set. At each training iteration, the classifier parameters are adjusted as usual, but its accuracy is evaluated on the validation data set. The termination iteration is determined as the last iteration during which the classifier improved its accuracy on the validation data set.

Example-based optimization of classification systems will transpose the same issue to the optimization process. Thus, a strategy to control solution over-fit to the optimization data set (data used during optimization) is necessary, in order to select solutions possessing good generalization power. Figure 1 shows a set of solutions explored by a Pareto-based MOGA. Each point represents a classifier with its accuracy (error rate) and complexity — feature vector cardinality, expressed in zones of interest. Figure 1(a) is the objective function space associated with the



(a) Optimization data set (b) Unknown observations

Fig. 1. Good solutions (classifiers) obtained during the optimization process perform differently on data not present in the optimization data set. Each point represents a classifier error rate and cardinality, ranked from lowest (+) to highest error rate (×) for each cardinality value on the optimization data. Solution P_2 has the smallest error rate on the Pareto front (a), but is dominated by D_1 when classifying new data (b). Solution D_2 generalizes best, but is discarded by traditional Pareto-based MOGAs in (a).

optimization data set, while Fig. 1(b) shows the performance of the same classifiers applied to data not present in the optimization set. As can be seen, classifier P_2 is the solution with the smallest error rate on the Pareto front, but it does not generalize as well as P_1 as shown in Fig. 1(b). A common technique used to overcome this type of over-fit following the optimization process is to validate the solutions on the Pareto front with yet another data set.[1,8,25] This strategy produces better results than selecting solutions based solely on the accuracy of the optimization data set, but it will miss D_1 and D_2. Both are dominated solutions discarded by the multiobjective evolutionary optimizer. Note that Oliveira *et al.* observed the same effect in Ref. 25 for the feature subset selection problem where classifier performance on unknown observations is different from the performance observed during the optimization process.

In this work, two optimization algorithms are used and compared during the experiments and their performance evaluated. The first is the *Fast Elitist Non-Dominated Sorting Genetic Algorithm* (NSGA-II), introduced by Deb *et al.* in Ref. 4 and known to be an efficient solver of multiobjective optimization problems in the literature.[26,32] NSGA-II has also been used as a baseline MOGA for problem-specific algorithm comparison[3] and in general MOGA studies,[16] a further justification of this choice. The second algorithm is the *Multi-objective Memetic Algorithm* (MOMA), introduced by the authors in Ref. 29. MOMA combines a traditional MOGA with a local search algorithm to create a more powerful search

mechanism, and belong to a class of algorithms known as memetic algorithms, which has been the subject of recent research.[15,17] Jaszkiewicz demonstrates in Ref. 11 that hybrid optimization methods outperform methods based solely on genetic operations, hence the interest in using MOMA.

MOMA differs from traditional memetic algorithms by replacing the Pareto dominance relation by the *decision frontier* concept. The goal is to develop candidate solutions dominated by the Pareto front that may provide better generalization power, as demonstrated in Fig. 1. For a classification system, the goal is to optimize the best error rate for each feature set cardinality, obtaining the complete frontier in the objective function space. Solutions are optimized regarding two objective functions, a discrete function o_1 and o_2, either discrete or continuous. Candidate solutions have their o_2 value optimized for each possible o_1 value, classifier error rate and representation dimensionality for classifiers. This decision allows MOMA to search for solutions as D_2 in Fig. 1. Also, MOMA uses an order preserving strategy to keep solutions as D_2 in Fig. 1. This strategy uses an auxiliary archive, keeping the best n solutions for each o_1 value.

This paper is organized as follows. Section 2 details the classification system optimization approach used to demonstrate the candidate solution over-fit and to verify the proposed over-fit control strategy. The over-fit control strategy is presented in Sec. 3, whereas the experimental protocol used to asses the over-fit control strategy is detailed in Sec. 4. Section 5 presents and compares the results attained. The final section presents the conclusions, followed by an appendix detailing the statistical tests performed to compare results.

2. Pattern Classification Systems Optimization

Classification systems are based on classifiers that are responsible for classifying observations for information post-processing. Optimization of these systems can be divided into three specific tasks. The first is *Intelligent Feature Extraction* (IFE),[30] which extracts feature sets for single classifiers. For higher accuracy, the feature sets obtained with IFE can be reduced through *feature subset selection* (FSS).[18] The feature sets can also be used to create an *ensemble of classifiers* (EoC).[6,14]

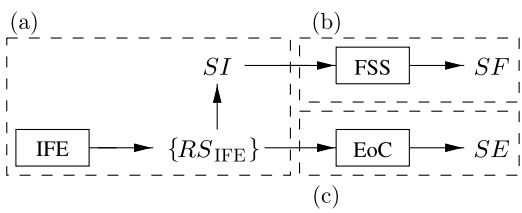


Fig. 2. Classification system optimization. Feature vectors obtained with IFE may be used to further improve accuracy with EoCs, or the complexity of a single classifier may be reduced through FSS.

The classification system is modeled by two different leveled processes as illustrated in Fig. 2. In both processes, the first level uses the IFE to obtain the feature vector set RS_{IFE} [Fig. 2(a)]. The feature vectors in RS_{IFE} are used to train classifiers that are further processed on the next level where the complexity of the best single classifier SI may be reduced through feature subset selection [Fig. 2(b)]. Finally, the trained classifiers may be considered for aggregation to form an EoC depending on the intended application [Fig. 2(c)].

2.1. *Intelligent feature extraction*

Human experts are traditionally responsible for choosing the feature vector used in classification systems. This vector is most often determined using domain knowledge and domain context on a trial-and-error basis. *Intelligent Feature Extraction* (IFE) uses the domain knowledge and domain context in an approach formulated as a multiobjective optimization problem to obtain a set of feature vectors.

IFE models patterns as features extracted from specific *foci* of attention on images (Fig. 3). This is a strategy known to provide better recognition results than the extraction of features from the whole image.[20] Two operators are used to generate feature vectors with IFE: a *zoning operator* to define foci of attention over images, and a *feature extraction* operator to apply transformations in zones. The choice of transformations for the feature extraction operator constitutes the domain knowledge introduced by the human expert. To obtain feature vectors for specific pattern recognition problems, the domain context is introduced in the form of actual observations in the optimization data set used to evaluate and compare solutions. Hence, IFE optimizes the zoning operator.

The IFE operators are combined to generate a feature vector such as illustrated in Fig. 3. The zoning operator defines the zoning strategy $Z = \{z^1, \ldots, z^n\}$, where $z^i, 1 \leq i \leq n$ is a zone in the image I and n the total number of zones. Pixels inside the zones in Z are transformed by the feature extraction operator in the representation $F = \{f^1, \ldots, f^n\}$, where $f^i, 1 \leq i \leq n$ is the partial feature vector extracted from z^i. At the end of the optimization process, the resulting feature

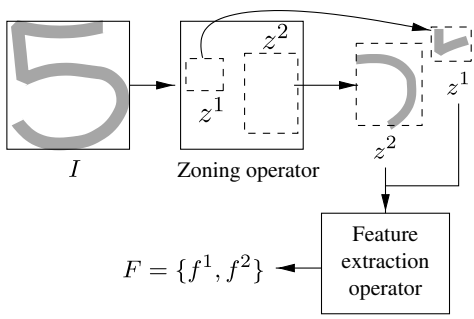


Fig. 3. IFE structure. Domain knowledge is introduced by the feature extraction operator, and the zoning operator is optimized based on the domain context.

vector set $RS_{\text{IFE}} = \{F^1, \ldots, F^p\}$ gives the user a choice among various trade-offs with respect to the optimization objectives.

In IFE, candidate solutions are evaluated with respect to two objective functions, classification accuracy and dimensionality. If the dimensionality is too high, the classifier tends to memorize the training set, an effect called the *curse of dimensionality*. Classification processing time is also related to feature dimensionality: the more features associated with the abstraction representation, the longer it takes to classify unknown observations. Hence, the wrapped classifier needs to be computationally efficient and reasonably accurate to prototype IFE solutions. Kimura *et al.* discussed in Ref. 13 the *projection distance* (PD) classifier. Based on hyper planes to model classes, the PD classifier fairly quickly trains and classifies observations. Therefore, the PD classifier was chosen for the wrapper approach used to evaluate IFE feature vectors.

2.2. *Feature subset selection*

Feature subset selection (FSS) is aimed at optimizing the classification process, thereby eliminating irrelevant features from the original feature set in order to create a smaller, yet accurate, abstract representation. Due to the curse of dimensionality, which favors smaller feature sets, FSS may also improve classification accuracy by eliminating irrelevant features. The FSS problem is defined as finding a good feature subset regarding some objective function. In this work, the objective functions will minimize classification error and feature vector cardinality.

Kudo and Sklansky compared FSS techniques in Ref. 18, concluding that a genetic algorithm performs better when the original feature vector length is long (more than 50 features). Oliveira *et al.* applied a genetic algorithm based FSS in Ref. 24, postulating that a MOGA could further enhance the results obtained, which was later confirmed in Ref. 25. The superiority of MOGA in FSS is also confirmed by Emmanouilidis *et al.* in Ref. 8 using sonar and ionosphere data.

2.3. *Ensemble of classifiers optimization*

A recent trend in machine learning has been to combine several learners to improve their overall performance.[6] Thus, classifiers can be combined to improve the classification stage in PR systems. An EoC is typically created by running a learning algorithm several times to create a set of classifiers, which are then combined by an aggregation function. One approach to create such classifier set is to manipulate the feature set used to train classifiers, which can be performed through feature subset selection, as in Ref. 19, or through transformations on the feature set, as in the random subspace approach.[10] The key issue in this process is to generate a set of diverse and fairly accurate classifiers for combination.[14]

The proposed EoC optimization selects which classifiers to aggregate from the classifier set $K = \{K^1, \ldots, K^p\}$, where K^i is the classifier trained with the feature vector F^i in the IFE resulting feature vector set $RS_{\text{IFE}} = \{F^1, \ldots, F^p\}$. This

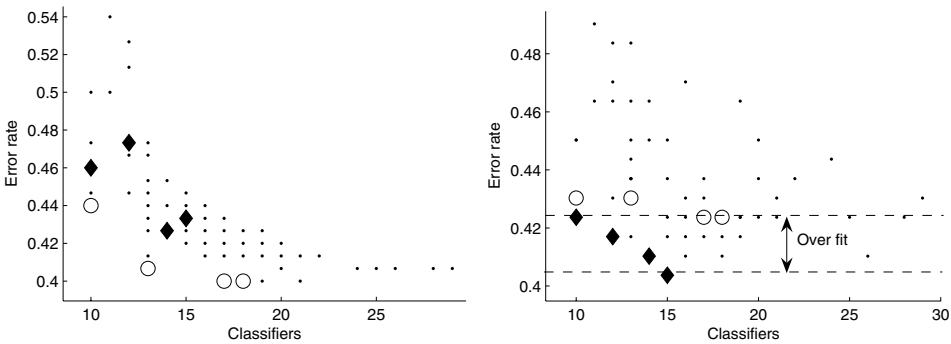
hypothesis assumes that RS_{IFE} generates a set K of p diverse and fairly accurate classifiers. To realize this task as a multiobjective optimization problem, the classifiers in K are associated with a binary string E of p bits, which is optimized to select the best combination of classifiers using a MOGA. The classifier K^i is associated with the ith binary value in E, which indicates whether or not the classifier is active in the EoC. Again, the optimization process is guided by two objectives — cardinality and quality. Ruta and Gabrys postulated in Ref. 28 that combined classifier performance is a reliable and meaningful criterion with which to compare EoCs. Thus, the goal is to minimize both EoC cardinality and the associated error rate on the optimization data set.

Evaluating the EoC error rate requires actual classifier aggregation, which depends on the classifier employed. The normalized continuous values of MLP classifier output are aggregated by their output average.[14] To speed up the experimental process, the MLP outputs are calculated once only, and their actual aggregation is calculated during runtime. PD classifiers are aggregated by majority voting. As with MLP classifiers, PD votes are calculated once only, and the votes are counted during runtime.

3. Solution Over-Fit Control Strategy

Similar to the validation process in classifier training, solution over-fit control needs to be performed at each generation. To illustrate this assertion, consider the case where NSGA-II is used to optimize the previously discussed classification system. Figure 4 shows all candidate solutions generated by the NSGA-II algorithm at some generation. Figure 4(a) is the objective function space used during the optimization process, and Fig. 4(b) is the objective function space used for validation. Points are candidate solutions in the current generation (MLP EoCs). Circles represent solutions in the current Pareto front, and diamonds the current Pareto front obtained in validation. The first observation is that, through the generations, nondominated solutions are not always the best after validation. The second observation is that solutions with good generalization power are eliminated by genetic selection, which emphasizes solutions with good performance on the optimization data set (memorization). Hence, the most appropriate approach is to validate candidate solutions in all generations during the optimization process with a selection data set. Performing over-fit control at each generation is hereafter referred to as the global control strategy.

Validating solutions in all generations requires an auxiliary archive with which to store good validated solutions. An algorithmic template for MOGAs using global control is detailed in Algorithm 1, which requires a disjoint selection data set and an auxiliary archive S to store the validated approximation set. A MOGA evolves the population P_t during mg generations. At each generation, the population P_{t+1} is validated and the auxiliary archive S is updated with solutions that have good generalization power. Like the validation strategy used when training classifiers,



(a) Optimization (b) Validation

Fig. 4. MLP EoC solutions as perceived by the optimization and validation processes at generation $t = 14$ with NSGA-II. Each point represents a candidate solution, circles represent nondominated solutions found during the optimization process and diamonds validated nondominated solutions.

the validation stage provides no feedback to the MOGA. At the end of the optimization process, the best candidate solutions are stored in S. To preserve a choice of trade-offs, solutions are inserted and removed from S according to the optimization algorithm used. For a Pareto-based MOGA such as NSGA-II, nondominated solutions in validation are inserted into S, and dominated solutions are removed if necessary. For MOMA, solutions are inserted according to the decision frontier concept and the maximum number of solutions allowed per cardinality value in the original archive.

4. Experimental Protocol

The test procedures indicated in Fig. 5 are performed to verify the impact of overfit on the previously discussed classification system optimization approach. They are performed with both NSGA-II and MOMA for comparison purposes. First, the IFE problem is solved to obtain the feature vector set RS_{IFE} (the auxiliary archive S). For NSGA-II, S is a Pareto front, while for MOMA, RS_{IFE} is a set of $\max_{S^l}$ fronts. These sets are used to train the classifier sets K_{PD} and K_{MLP} using the PD and MLP classifiers. For a single classifier system, the most accurate classifiers $\mathrm{SI_{PD}} \in \mathrm{K_{PD}}$ and $\mathrm{SI_{MLP}} \in \mathrm{K_{MLP}}$ are selected. EoCs are then created with $\mathrm{K_{PD}}$ and $\mathrm{K_{MLP}}$, producing the ensembles $\mathrm{SE_{PD}}$ and $\mathrm{SE_{MLP}}$. To further compare NSGA-II and MOMA, an EoC is created with NSGA-II using the feature vector set RS_{IFE} optimized by MOMA, producing the ensembles $\mathrm{SE'_{PD}}$ and $\mathrm{SE'_{MLP}}$. These tests are performed 30 times for meaningful statistical analysis. Then, the FSS approach further refines the classifier $\mathrm{SI_{MLP}}$ using both NSGA-II and MOMA. Because of the processing time required for the feature subset selection experiments (1 to 3 days), this specific test is limited to a single run.

Algorithm 1. Template for a MOGA with over-fit control. The population P_{t+1} is validated with the selection data set, and the solutions possessing good generalization power are kept in the auxiliary archive S. In order to avoid over-fitting solutions to the selection data set, no feedback is provided to the optimization process from the control strategy.

Result: Auxiliary archive S
Creates initial population P_1 with m individuals;
$S = \emptyset$;
$t = 1$;
while$(t < mg)$
{
    Evolves P_{t+1} from P_t;
    Validate P_{t+1} with the selection data set;
    Update the archive S with individuals from P_{t+1} based on validation;
    $t = t + 1$;
}

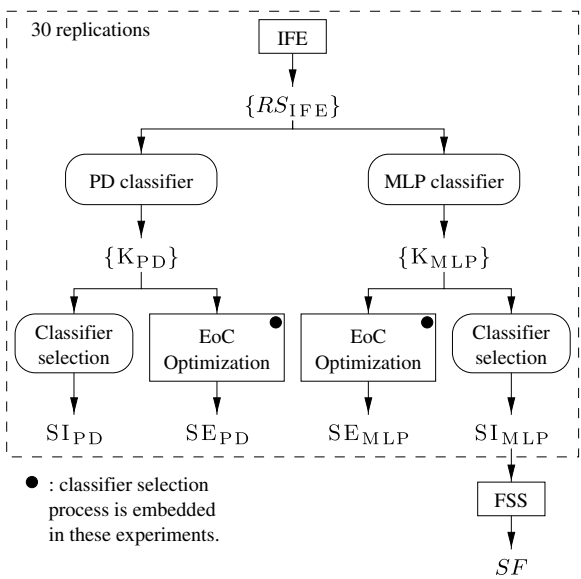


Fig. 5. Experimental overview. The classification system optimization approach is tested in two stages. IFE and the EoC problems are solved 30 times for statistical analysis. Experimentation on FSS is performed once, due to the processing time required. The PD classifier is tested only during the first stage, whereas the MLP classifier is tested in both.

To demonstrate solution over-fit, the experiments are analyzed in three situations. First, no control strategy is used and solutions are selected based only on the optimization data set accuracy. Next, candidate solutions are validated in the last generation with the selection data set. Finally, solutions are validated in all

Table 1. Handwritten digits data sets extracted from NIST-SD19.

Data set	Size	Origin	Sample Range
training'	50000	hsf_0123	1 to 50000
training	150000	hsf_0123	1 to 150000
validation	15000	hsf_0123	150001 to 165000
optimization	15000	hsf_0123	165001 to 180000
selection	15000	hsf_0123	180001 to 195000
test_a	60089	hsf_7	1 to 60089
test_b	58646	hsf_4	1 to 58646

generations with the global control strategy with the selection data set. The three approaches are compared to demonstrate which produces the best results.

The disjoint data sets in Table 1 are used in the experiments, which are isolated handwritten digits extracted from NIST-SD19. MLP hidden nodes are optimized as feature set cardinality fractions in the set $f = \{0.4, 0.45, 0.5, 0.55, 0.6\}$. MLP classifier training is performed with the *training* data set, while the PD classifier is trained with the smaller *training'* data set, to implement a computationally efficient wrapper approach for the IFE. The remaining data sets are used with both classifiers. The *validation* data set is used to adjust the classifier parameters (MLP hidden nodes and PD hyper planes). The wrapper approach is performed with the *optimization* data set, and the *selection* data set is used to validate candidate solutions (global control and control at the last generation). Solutions are compared with test_a and test_b, data sets unknown to the resulting solutions. It is known that test_b is more difficult to classify than test_a,[9] hence the robustness of the resulting solutions are tested on two different levels of classification complexity.

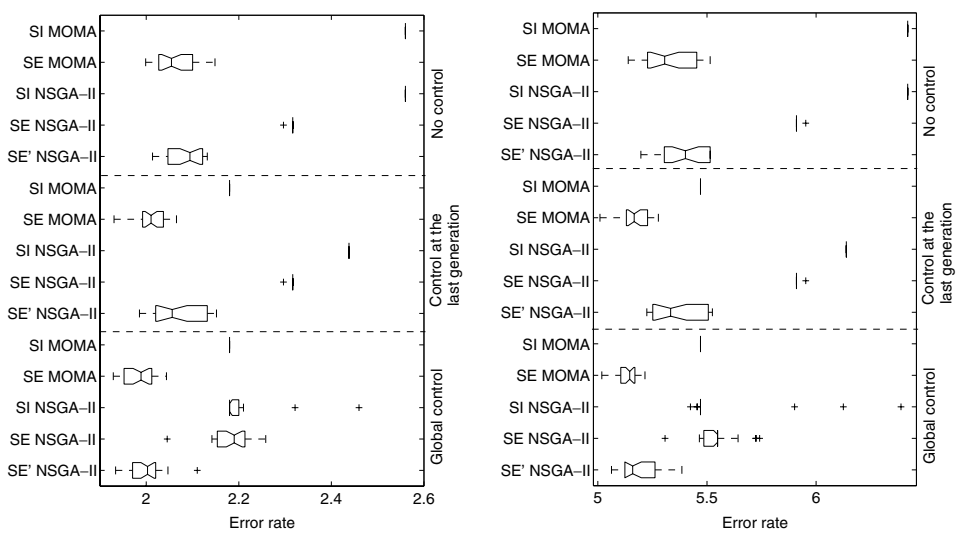
The parameters used with MOMA are the following: crossover probability is set to $p_c = 80\%$, and mutation is set to $p_m = 1/L$, where L is the length of the mutated binary string.[7] The maximum number of generations is set to $mg = 1000$ for all experiments, and the local search will look for $n = 1$ neighbors during $\text{NI} = 3$ iterations, with deviation $a = 0\%$. Each slot in the archive S is allowed to store $\max_{S^l} = 5$ solutions. These parameters were determined empirically. The same parameters ($p_c = 80\%$, $p_m = 1/L$ and $mg = 1000$) are used for NSGA-II. Population size depends on the optimization problem. To optimize the zoning operator, the population size is $m = 64$, while to optimize FSS, we use $m = 100$ to keep processing time reasonable. For EoC optimization, $m = 166$ is used when using the feature vector set RS_{IFE} optimized by MOMA, and $m = 32$ for RS_{IFE} optimized by NSGA-II. Individual initialization is performed in two steps for both optimization algorithms. The first step creates one individual for each possible cardinality value. For IFE and FSS optimization, one individual associated with each possible number of zones is added, while for EoC optimization, one individual is added for each possible EoC cardinality. The second step completes the population with individuals initialized with a Bernoulli distribution.

Experiments are conducted on a Beowulf cluster with 25 nodes using Athlon XP 2500+ processors with 1GB of RAM. The optimization algorithms were implemented using LAM MPI v6.5 in a simple master-slave distributed computing topology. PD votes and MLP outputs calculation were performed once in parallel and results were stored in files to be loaded into memory for the EoC optimization process.

5. Experimental Results

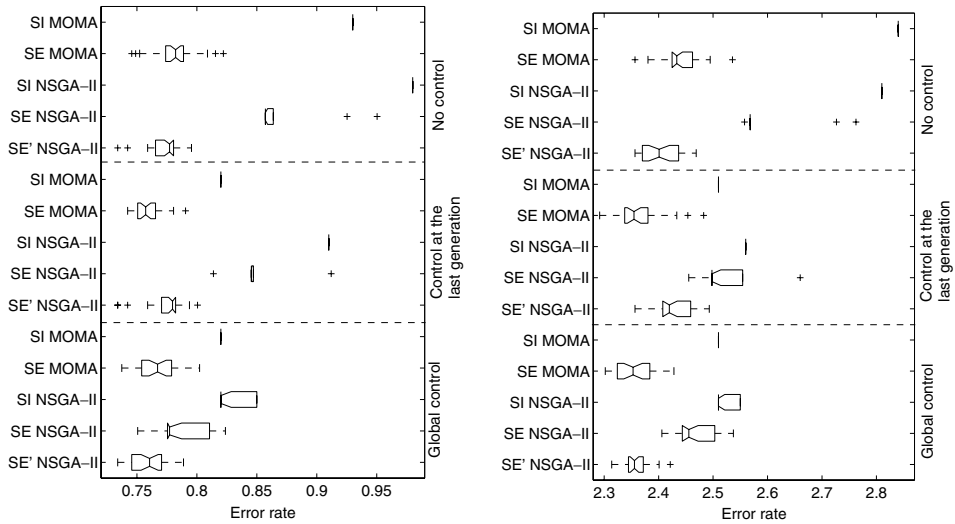
The first experiments optimize the IFE and EoC problems in 30 replications. For each run, the most accurate solution is selected according to the control strategy used. Figures 6 and 7 detail the error rate dispersion obtained in these experiments for the PD and MLP classifiers respectively. Mean values for these experiments are detailed in Tables 2 and 3. In both tables, C indicates the control strategy used (N for no control, LG for control at the last generation and G for global control). Z indicates the solution zone number, $|S|$ the solution cardinality in features or aggregated classifiers, and e_a and e_b the error rates in the test_a and test_b data sets. The baseline solution, taken from Ref. 25, is included in both tables for reference purposes.

In terms of control strategy, the results indicate an order relation between the approaches tested. Using no control is worse than using control at the last generation, which in turn is worse than using the global control strategy. Mean error rate values in Tables 2 and 3 are lower with the global control strategy, which is also



(a) test_a (b) test_b

Fig. 6. PD error rate dispersion on 30 replications. Each solution set relates to one control strategy tested: no control, control at the last generation and global control.



(a) test_a (b) test_b

Fig. 7. MLP error rate dispersion on 30 replications. Each solution set relates to one control strategy tested: no control, control at the last generation and global control.

Table 2. PD optimization results — mean values on 30 replications and standard deviation (shown in parenthesis).

C	Solution	MOMA				NSGA-II			
		Z	$\lvert S\rvert$	e_a	e_b	Z	$\lvert S\rvert$	e_a	e_b
—	*Baseline*	6	132	2.96%	6.83%	6	132	2.96%	6.83%
N	SI_{PD}	12	264	2.57% (0)	6.42% (0)	12	264 (0)	2.57% (0)	6.42% (0)
	SE_{PD}	—	16.67	2.07% (0.043)	5.32% (0.121)	—	7	2.31% (0.007)	5.92% (0.015)
	SE'_{PD}	—	—	—	—	—	12.47	2.08% (0.042)	5.40% (0.112)
LG	SI_{PD}	15	330	2.18% (0)	5.47% (0)	10	220	2.44% (0)	6.14% (0)
	SE_{PD}	—	25.5	2.01% (0.032)	5.17% (0.069)	—	7	2.31% (0.07)	5.91% (0.014)
	SE'_{PD}	—	—	—	—	—	11.47	2.07% (0.058)	5.37% (0.114)
G	SI_{PD}	15	330	2.18% (0)	5.47% (0)	15.77	346.85	2.22% (0.040)	5.55% (0.087)
	SE_{PD}	—	22.33	1.98% (0.033)	5.14% (0.056)	—	7.4%	2.18% (0.063)	5.53 (0.122)
	SE'_{PD}	—	—	—	—	—	24.67	2.00% (0.040)	5.19% (0.087)

Table 3. MLP optimization results — mean values on 30 replications and standard deviation (shown in parenthesis).

C	Solution	MOMA				NSGA-II			
		Z	$\|S\|$	e_a	e_b	Z	$\|S\|$	e_a	e_b
	Baseline	6	132	0.91%	2.89%	6	132	0.91%	2.89%
N	SI_{MLP}	8	176	0.93% (0)	2.84% (0)	6	132	0.98% (0)	2.81% (0)
	SE_{MLP}	—	10.07	0.78% (0.017)	2.44% (0.037)	—	4.73	0.88% (0.037)	2.61% (0.081)
	SE'_{MLP}	—	—	—	—	—	6.8	0.77% (0.015)	2.41% (0.037)
LG	SI_{MLP}	15	330	0.82% (0)	2.51% (0)	12	264	0.91% (0)	2.56% (0)
	SE_{MLP}	—	16.23	0.76% (0.012)	2.37% (0.042)	—	4.77	0.85% (0.015)	2.52% (0.040)
	SE'_{MLP}	—	—	—	—	—	4.9	0.77% (0.016)	2.42% (0.034)
G	SI_{MLP}	15	330	0.82% (0)	2.51% (0)	13.67	300.6	0.83% (0.013)	2.52% (0.018)
	SE_{MLP}	—	10.33	0.77% (0.017)	2.35% (0.030)	—	4.5	0.79% (0.023)	2.47% (0.040)
	SE'_{MLP}	—	—	—	—	—	14.13	0.76% (0.016)	2.36% (0.022)

observed in Figs. 6 and 7. While valid for the general case, in three specific situations the differences among control strategies are not significant. Using MOMA with IFE produces the same results with control at the last generation or the global control. The same is observed when creating MLP EoCs with MOMA. Finally, creating EoCs with NSGA-II will produce similar results with no control or control at the last generation. The effect observed with MOMA is related to its auxiliary archive strategy, which keeps a solution archive of $\max_{S^l} = 5$ solutions for each possible cardinality value, and partially removes the over-fit with the control at the last generation. The exception observed with NSGA-II is not relevant, as the global control strategy is better. The results indicate that the global control strategy is better for both the IFE and EoC problems, as the impact of over-fit on solutions is not known *a priori*.

Optimization algorithms are compared based on the results obtained with the global control strategy. MOMA outperforms NSGA-II with both classifiers to optimize IFE (solutions SI_{PD} and SI_{MLP}), selecting the zoning representation with 15 active zones. MOMA has no error rate dispersion with either classifier, whereas NSGA-II may produce suboptimal solutions. This result is no surprise, as MOMA was designed for the IFE problem. Another motivation for choosing MOMA at the IFE level is to create a more diverse feature vector set RS_{IFE}. In the 30 experimental replications, the RS_{IFE} cardinality is $|RS_{IFE}| = 82$ when $\max_{S^l} = 5$ with MOMA, while with NSGA-II the cardinality is only $|RS_{IFE}| = 10$. MOMA also

offers higher lateral diversity due to the decision frontier, and its RS_{IFE} set comprises individuals with up to 36 active zones, while the RS_{IFE} produced by NSGA-II contains individuals with at most 15 active zones. Classifier diversity is a key issue in optimizing EoCs, and this is reflected in the results in Tables 2 and 3, where the ensembles SE_{PD} and SE_{MLP} optimized by NSGA-II are worse than the ensembles SE'_{PD} and SE'_{MLP} optimized by NSGA-II with MOMA's feature vector set RS_{IFE}.

Comparing the EoCs optimized with the global control and using MOMA's feature vector set RS_{IFE}, there is no significant difference between MOMA and NSGA-II. Mean error rates in Tables 2 and 3 are comparable, as is the error rate dispersion in Figs. 3 and 6. NSGA-II is less computational-intensive than MOMA, however, and thus it is preferable to use NSGA-II to optimize EoCs with the feature vector set RS_{IFE} obtained by MOMA. The conclusions discussed for both IFE and the EoC were verified with a nonparametric multiple comparison procedure, which is discussed in the appendix.

Mean error rates in Tables 2 and 3 also demonstrate the accuracy improvement over the baseline reference defined by a human expert. For a single PD classifier, the IFE solutions reduced the number of misclassifications by 26.35% on test_a and 19.91% on test_b. For a PD EoC obtained by NSGA-II, misclassifications on test_a were reduced by 32.43% based on mean values and by 24.01% on test_b. While improvements are proportionally higher on test_a, numerically they are more significant on test_b, where the EoC can correctly recognize 1.64% (962) more observations. This is an important improvement, as test_b is more difficult to classify than test_a. For a single MLP classifier, the IFE solutions improved accuracy by 9.89% on test_a and by 13.15% on test_b. For an MLP EoC obtained with NSGA-II, improvements based on mean values are 16.48% on test_a and 18.33% on test_b. Again, improvements on test_b are numerically higher, as the EoC can correctly recognize 0.53% (311) more observations in test_b.

The next experiment reduces the single MLP classifier SI_{MLP} complexity through feature subset selection. The goal is to reduce feature complexity, while keeping the accuracy comparable to that of the original SI_{MLP}. Table 4 details the solutions obtained in the FSS optimization experiment with the use of all control strategies. In this table, C is the control strategy used (N for no control, LG for control at the last generation and G for global control), Z is the number of active zones, $|S|$ the feature cardinality (feature number) and e_a and e_b the error rates in the test_a and test_b data sets. The table also includes the baseline reference and the original SI_{MLP} representation optimized by IFE for comparison purposes.

Feature subset selection results also confirm the need for global control in classification systems optimization. Solutions selected with global control are more accurate than those selected with other strategies. Reducing complexity also improved accuracy on test_b, which is higher in comparison with that of SI_{MLP}. This improvement is associated with the higher generalization power of smaller feature sets. NSGA-II also reduced the SI cardinality from 330 features to 318 features in SF, a reduction of 3.64%. It can be said that IFE optimizes features with a low number

Table 4. FSS optimization results – best values from a single replication.

		MOMA				NSGA-II			
C	Solution	Z	$\|S\|$	e_a	e_b	Z	$\|S\|$	e_a	e_b
—	*Baseline*	6	132	0.91%	2.89%	6	132	0.91%	2.89%
—	SI_{MLP}	15	330	0.82%	2.51%	15	330	0.82%	2.51%
N	*SF*	15	301	0.86%	2.56%	15	296	0.85%	2.64%
LG	*SF*	15	307	0.82%	2.52%	15	296	0.85%	2.64%
G	*SF*	15	321	0.82%	2.46%	15	318	0.83%	2.51%

of correlated features, and thus FSS is not capable of removing a higher number of features, as these are required for classification.

On all optimization problems, it was observed that MOMA explored a higher number of unique solutions, as it optimizes the complete decision frontier and uses a local search approach to explore solutions. This approach allowed MOMA to find a better RS_{IFE} set for the IFE problem. The drawback is that MOMA uses more processing time, and NSGA-II offers a better compromise between solution quality and processing time for the EoC and FSS problems.

6. Conclusions

This research has demonstrated that, similar to learning algorithms, methodologies to optimize classification systems using a wrapped classifier are prone to solution over-fit. Control strategies to overcome this challenge have been discussed and tested. It was observed in some problems that the global control is not significantly better than control at the last generation. However, since the impact of over-fit on solutions obtained is unknown *a priori*, it is better to use global control as it guarantees solution quality once the optimization process has been completed in all situations.

Experimental tests were also used to evaluate the methodologies for optimizing classification systems with both the PD and MLP classifiers. IFE solutions outperformed the traditional human expert approach and produced a set of diverse classifiers, which can be aggregated into an EoC for higher accuracy than a single classifier. IFE also prototypes solutions using a computationally efficient wrapper classifier, which reduces processing time and turns IFE into a feasible approach for genetic optimization.

It was also demonstrated that MOMA outperforms NSGA-II for the IFE problem. This effect is associated with the exploratory mechanism in MOMA and its archiving strategy. The drawback is that MOMA requires more processing power to complete the optimization process. The EoC problem results indicate no significant advantage in accuracy for either algorithm, and thus we conclude that NSGA-II is the most appropriate optimization algorithm for EoC optimization, as it requires less processing time. As for the FSS approach, results for both algorithms are comparable in accuracy, and considering the limited experimental data we observed

that NSGA-II is most adequate in terms of required processing power. A more significant statistical analysis is necessary, however, the required processing time currently makes this analysis not feasible.

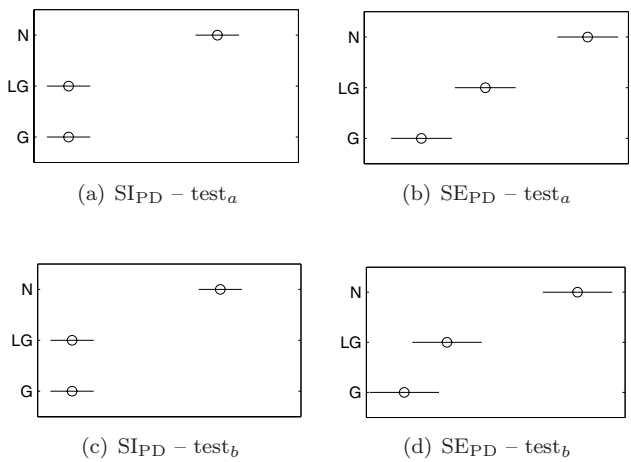
Acknowledgments

The first author would like to acknowledge the CAPES and the Brazilian government for supporting this research through scholarship grant BEX 2234/03-3. The other authors would like to acknowledge the NSERC (Canada) for supporting this research. The authors also acknowledge the anonymous reviewers for their invaluable feedback to improve this paper.

Appendix

To compare solutions obtained, a nonparametric multiple comparison procedure with Dunn–Sidak correction is performed. The null hypothesis, in this context, states that there is no significant statistical difference between the mean error rates of samples obtained from different experiments. The alternative hypothesis is that mean error rates are different. The null hypothesis is verified with a confidence level of 95% ($\alpha = 0.05$). The first goal is to determine which is the best over-fit control strategy. Next, results obtained are compared with both optimization algorithms. The multiple comparison is also used to verify improvements obtained with EoCs in comparison to a single classifier based approach.

All figures used to compare results (Figs. 8 to 13) indicate the statistical difference between mean values. Each experiment is represented by a line segment, where values in the x axis are the rank in the multiple comparison (not actual



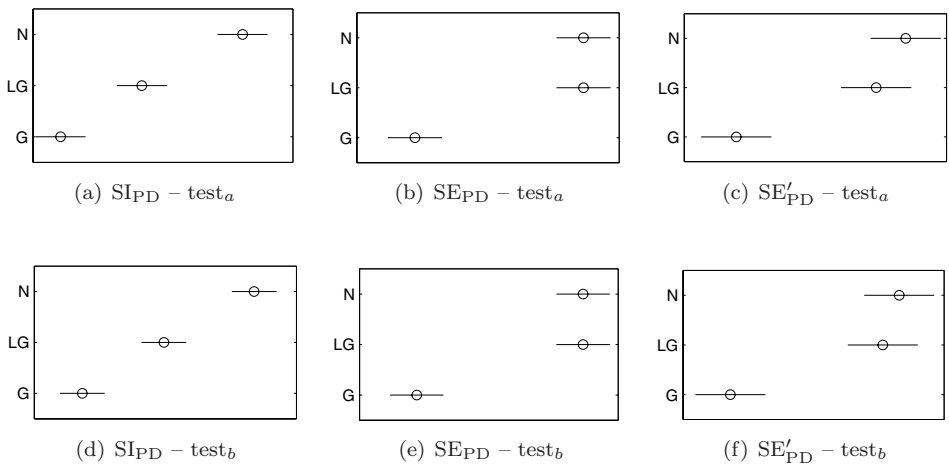
(a) $\mathrm{SI}_{\mathrm{PD}} - \mathrm{test}_a$ (b) $\mathrm{SE}_{\mathrm{PD}} - \mathrm{test}_a$

(c) $\mathrm{SI}_{\mathrm{PD}} - \mathrm{test}_b$ (d) $\mathrm{SE}_{\mathrm{PD}} - \mathrm{test}_b$

Fig. 8. PD multiple comparison results for the over-fit control strategies in each optimization problem with MOMA. Strategies tested are no over-fit control (N), control at the last generation (LG) and global control (G).

error rates). Overlaping line segments indicate comparable mean values, whereas nonoverlapping line segments indicate significantly different results.

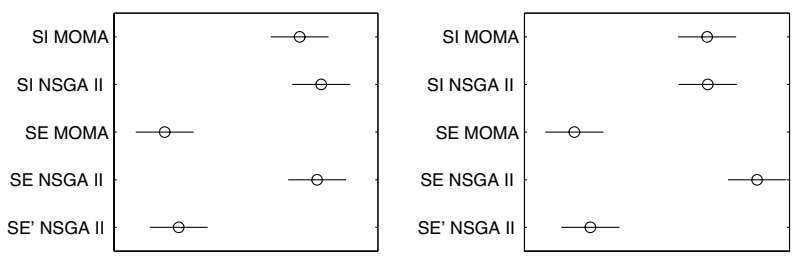
Figure 8 compares results obtained with MOMA using the PD classifier. Results obtained with global over-fit control and control at the last generation are comparable, owing to the order preserving archive used by MOMA. Results with the NSGA-II in Fig. 9 demonstrate that global control performs better. These results indicate that global control is the best choice when the over-fit impact on the optimization algorithm is unknown. Thus, the final comparison in Fig. 10 uses the global control to compare solutions optimized for the PD classifier. Both MOMA and NSGA-II are comparable to optimize the IFE for a single classifier system, however, if an EoC approach is targeted, MOMA produces a better RS_{IFE} set to create classifiers for EoC selection. The NSGA-II requires less processing power to optimize EoCs, then it is clear that it should be used in place of MOMA.



(a) $\mathrm{SI_{PD}}$ – test_a (b) $\mathrm{SE_{PD}}$ – test_a (c) $\mathrm{SE'_{PD}}$ – test_a

(d) $\mathrm{SI_{PD}}$ – test_b (e) $\mathrm{SE_{PD}}$ – test_b (f) $\mathrm{SE'_{PD}}$ – test_b

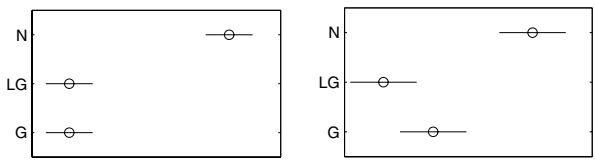
Fig. 9. PD multiple comparison results for the over-fit control strategies in each optimization problem with NSGA-II. Strategies tested are no over-fit control (N), control at the last generation (LG) and global control (G).



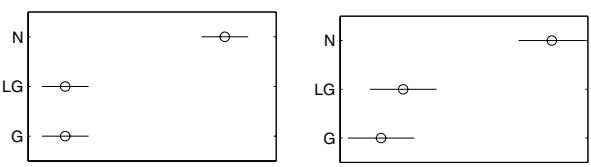
(a) test_a (b) test_b

Fig. 10. PD classification system optimization approaches multiple comparison using global control.

Figures 11 and 12 confirm with the MLP classifier that global control is better, since the over-fit impact is unknown *a priori*. Comparing the solutions obtained with this control strategy in Fig. 13 provides conclusions similar to those with the PD classifier. Again, both MOMA and NSGA-II are comparable to optimize the IFE for a single classifier system. When an EoC based system is targeted, MOMA provides a better solution set RS_{IFE}. Considering the required processing power, NSGA-II is better to optimize EoCs.

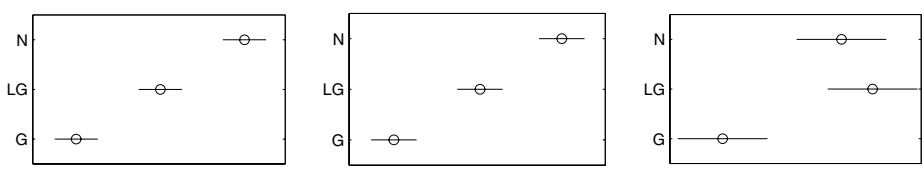


(a) $\text{SI}_{\text{MLP}} - \text{test}_a$ (b) $\text{SE}_{\text{MLP}} - \text{test}_a$

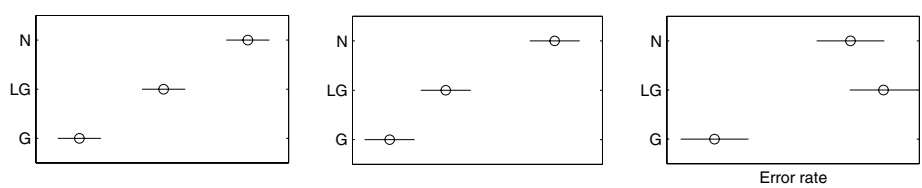


(c) $\text{SI}_{\text{MLP}} - \text{test}_b$ (d) $\text{SE}_{\text{MLP}} - \text{test}_b$

Fig. 11. MLP multiple comparison results for the validation strategies in each optimization problem with MOMA. Strategies tested are no over-fit control (N), control at the last generation (LG) and global control (G).

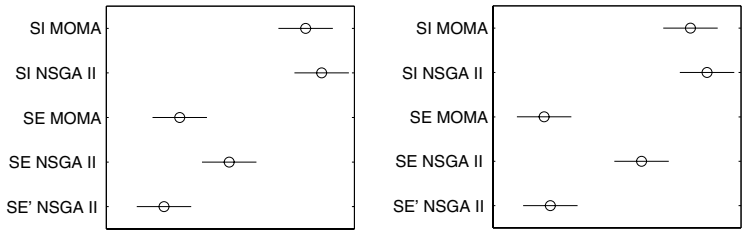


(a) $\text{SI}_{\text{MLP}} - \text{test}_a$ (b) $\text{SE}_{\text{MLP}} - \text{test}_a$ (c) $\text{SE}'_{\text{MLP}} - \text{test}_a$



(d) $\text{SI}_{\text{MLP}} - \text{test}_b$ (e) $\text{SE}_{\text{MLP}} - \text{test}_b$ (f) $\text{SE}'_{\text{MLP}} - \text{test}_b$

Fig. 12. MLP multiple comparison results for the validation strategies in each optimization problem with NSGA-II. Strategies tested are no over-fit control (N), control at the last generation (LG) and global control (G).



(a) test_a (b) test_a

Fig. 13. MLP classification system optimization approaches multiple comparison using global control.

These tests draws three conclusions: (1) the impact of over-fit is not known *a priori*, thus global control is the best approach; (2) MOMA is the best algorithm to optimize the IFE to provide a diverse RS_{IFE} result set; and (3) the NSGA-II should be used to optimize EoC from RS_{IFE} optimized by MOMA for its lower processing time.

References

1. S. Bandyopadhyay, S. K. Pal and B. Aruna, Multiobjective GAs, quantitative indices, and pattern classification, *IEEE Trans. Syst. Man, Cybern. — Part B: Cybern.* **34**(5) (2004) 2088–2099.
2. E. Bernadó i Mansilla, X. Llorà and I. Traus, Multiobjective learning classifier systems, *Studies in Comput. Intell.* **16** (2006) 1261–1288.
3. R. O. Day and G. B. Lamont, Extended multi-objective fast messy genetic algorithm solving deception problems, *Proc. Third Int. Conf. Evolutionary Multi-Criterion Optimization (EMO 2005)*, (2005), pp. 296–310.
4. K. Deb, S. Agrawal, A. Pratab and T. Meyarivan, A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II, *Proc. Parallel Problem Solving from Nature VI Conf.* (2000), pp. 849–858.
5. K. Deb, *Multi-Objective Optimization using Evolutionary Algorithms* (John Wiley, 2001).
6. T. G. Dietterich, Ensemble learning, *The Handbook of Brain Theory and Neural Networks*, 2nd Edition, ed. M. A. Arbib (MIT Press, 2002).
7. A. E. Eiben, R. Hinterdind and Z. Michalewicz, Parameter control in evolutionary algorithms, *IEEE Trans. Evolut. Comput.* **3**(2) (1999) 124–141.
8. C. Emmanouilidis, A. Hunter and J. MacIntyre, A multiobjective evolutionary setting for feature selection and a commonality-based crossover operator, *Proc. 2000 Congress on Evolutionary Computation* (2000), pp. 309–316.
9. P. J. Grother, NIST Special Database 19 — Handprinted forms and characters database, National Institute of Standards and Technoloy — NIST (Database CD documentation) (1995).
10. T. K. Ho, Nearest neighbors in random subspaces, *Proc. 2nd Int. Workshop on Statistical Techniques in Pattern Recognition* (1998), pp. 640–648.
11. A. Jaszkiewicz, Do multiple-objective metaheuristics deliver on their promise? A computational experiment on the set-covering problem," *IEEE Trans. Evolut. Comput.* **7**(2) (2003) 133–143.

12. F. Kimura, K. Takashina, S. Tsuruoka and Y. Miyake, Modified quadratic discriminant functions and the application to Chinese character recognition, *IEEE Trans. Patt. Anal. Mach. Intell.* **9**(1) (1987) 149–152.
13. F. Kimura, S. Inoue, T. Wakabayashi, S. Tsuruoka and Y. Miyake, Handwritten numeral recognition using autoassociative neural networks, *Proc. Int. Conf. Pattern Recognition* (1998), pp. 152–155.
14. J. Kittler, M. Hatef, R. P. W. Duin and J. Matas, On combining classifiers, *IEEE Trans. Patt. Anal. Mach. Intell.* **20**(3) (1998) 226–239.
15. J. D. Knowles and D. Corne, Memetic algorithms for multiobjective optimization: issues, methods and prospects, *Recent Advances in Memetic Algorithms*, eds. N. Krasnogor, J. E. Smith and W. E. Hart (Springer Verlag, 2004), pp. 313–352.
16. J. B. Kollat and P. M. Reed, The value of online adaptative search: a performance comparison of NSGA-II, ε-NSGA-II and ε-MOEA, *Proc. Third Int. Conf. Evolutionary Multi-Criterion Optimization (EMO 2005)* (2005), pp. 296–310.
17. N. Krasnogor and J. Smith, A tutorial for competent memetic algorithms: model, taxonomy, and design issues, *IEEE Trans. Evolut. Comput.* **9**(5) (2005) 474–488.
18. M. Kudo and J. Sklansky, Comparison of algorithms that select features for pattern classifiers, *Patt. Recogn.* **33**(1) (2000) 25–41.
19. L. I. Kuncheva and L. C. Jain, Design classifier fusion systems by genetic algorithms, *IEEE Trans. Evolut. Comput.* **4**(4) (2000) 327–336.
20. V. di Lecce, G. Dimauro, A. Guerriero, S. Impedovo, G. Pirlo and A. Salzo, Zoning design for hand-written numeral recognition, *Proc. Seventh Int. Workshop on Frontiers in Handwriting Recognition – IWFHR-7* (2000), pp. 583–588.
21. X. Llorà, D. E. Goldberg, I. Traus and E. Bernadó i Mansilla, Accuracy, parsimony, and generality in evolutionary learning systems via multiobjective selection, *Int. Workshop in Learning Classifier Systems* (2002), pp. 118–142.
22. J. Loughrey and P. Cunningham, Overfitting in wrapper-based feature subset selection: the harder you try the worse it gets, *Proc. Int. Conf. Innovative Techniques and Applications of Artificial Intelligence* (2004), pp. 33–43.
23. J. Loughrey and P. Cunningham, "Using early-stopping to avoid overfitting in wrapper-based feature subset selection employing stochastic search, *TCD-CS-2005-37*, Department of Computer Science, Trinity College (2005).
24. L. S. Oliveira, N. Benahmed, R. Sabourin, F. Bortolozzi and C. Y. Suen, Feature subset selection using genetic algorithms for handwritten digit recognition, *Proc. 14th Brazilian Symp. Computer Graphics and Image Processing* (2001), pp. 362–369.
25. L. S. Oliveira, R. Sabourin, F. Bortolozzi and C. Y. Suen, A methodology for feature selection using multi-objective genetic algorithms for handwritten digit string recognition," *Int. J. Patt. Recogn. Artif. Intell.* **17**(6) (2003).
26. G. L. Pappa, A. A. Freitas and C. A. A. Kaestner, Multi-objective algorithms for attribute selection in data mining, *Applications of Multi-Objective Evolutionary Algorithms*, eds. C. A. Coello Coello and G. B. Lamont (World Scientific, 2004), pp. 603–626.
27. J. Reunanen, Overfitting in making comparisons between variable selection methods, *J. Mach. Learn. Res.* **3** (2003) 1371–1382.
28. D. Ruta and B. Gabrys, Classifier selection for majority voting, *Inform. Fus.* **6** (2005) 63–81.
29. P. V. W. Radtke, T. Wong and R. Sabourin, A multi-objective memetic algorithm for intelligent feature extraction, *Proc. Third Int. Conf. Evolutionary Multi-Criterion Optimization* (2005), pp. 767–781.

30. P. V. W. Radtke, R. Sabourin and T. Wong, Intelligent feature extraction for ensemble of classifiers, *Proc. 8th Int. Conf. Document Analysis and Recognition (ICDAR 2005)* (2005), pp. 866–870.
31. P. V. W. Radtke, Classification systems optimization with multi-objective evolutionary algorithms, Ph.D. dissertation, École de technologie supérieure – ÉTS – Université du Québec, Montréal, Québec, Canada (2006).
32. F. Schlotmann, A. Mitschele and D. Seese, A multi-objective approach to integrated risk management, *Proc. Third Int. Conf. Evolutionary Multi-Criterion Optimization*, (2005), pp. 692–706.

Paulo V. W. Radtke holds a B.Sc. degree in computer science and a M.Sc. degree in applied informatics from the Pontifícia Universidade Católica do Paraná, Brazil, in 1996 and 2000 respectively, and a Ph.D. degree in engineering from École de technologie supérieure, Canada, in 2006. Currently, he is an assistant professor at the Universidade Federal do Paraná – Setor Escola Técnica, and coaches students participating at the ACM ICPC and at the SBGAMES Independent Games Festival.

His current research interests include multi-objective optimization, evolutionary algorithms, pattern recognition, game programming and artificial intelligence.

Tony K. N. Wong holds a B.Eng. and M.Eng. degrees from École de Technologie Supérieure in electrical engineering. He received his Ph.D. degree in computer engineering from École Polytechnique de Montréal. Dr. Wong is a professional engineer and chair of the Automated Manufacturing Engineering Department, École de Technologie Supérieure.

His current research interests are in multi-objective optimization using evolutionary algorithms and its parallel implementations.

Robert Sabourin joined the Physics Department of the Montreal University in 1977 where he was responsible for the design, experimentation and development of scientific instrumentation for the Mont Mégantic Astronomical Observatory. In 1983, he joined the staff of the École de Technologie Supérieure, Université du Québec, in Montréal where he co-founded the Department of Automated Manufacturing Engineering where he is currently Full Professor and teaches pattern recognition, evolutionary algorithms, neural networks and fuzzy systems. In 1992, he joined also the Computer Science Department of the Pontifícia Universidade Católica do Paraná (Curitiba, Brazil) where he was co-responsible for the implementation in 1995 of a master program and in 1998 a Ph.D. program in applied computer science. Since 1996, he is a senior member of the Centre for Pattern Recognition and Machine Intelligence (CENPARMI, Concordia University).

His research interests are in the areas of handwriting recognition, signature verification, intelligent watermarking systems and bio-cryptography.