SIFT-SS: An Advanced Steady-State Multi-Objective Genetic Fuzzy System

Michel González[1], Jorge Casillas[2], Carlos Morell[1]

[1] Universidad Central "Marta Abreu" de Las Villas, Cuba

[2] Dept. Computer Science and Artificial Intelligence
University of Granada, Spain
http://decsai.ugr.es/~casillas

Outline

1. Motivation
2. Framework
3. Description of the proposed modifications
4. Experimental results
5. Conclusion and future work

Motivation

- Most of the existent multi-objective genetic fuzz systems (MOGFSs) are based on a gross usage of standard multi-objective evolutionary algorithms (MOEAs) like NSGA-II, SPEA(2), PAES or MOEA/D

- There are specific problem requirements in fuzzy modeling that can be taken into account to enhance the search process [Gacto *et al.*, 2008]:
  - It is easier to decrease the number of rules than to reduce the system error. This provokes a faster generation of the simplest solutions before exploring more promising rule configurations (which are dominated by such premature solutions)
  - The obtained parameters (in general) tends to be optimal for these premature solutions making difficult the appearance of better alternative solutions

- This paper is a study case in which the specificities of fuzzy modeling are incorporated into an existent MOGFS

Framework

- Simplification of Fuzzy models by Tuning, **SIFT** [Casillas, 2009]
  - Designed to learn a large number of parameters of the fuzzy system (number of labels, type of labels, number of input variables, membership fuzzy parameters, and fuzzy rule set)
  - Based on the well known NSGA-II algorithm

Advantages

- Highly efficient for large-scale regressions problems
- Generate very interpretable fuzzy partitions
- The obtained models are compact and very accurate

Disadvantages

- Difficulty to work with large population sizes
- Two of the objectives (#Rules and #Labels) converge faster than the third objective (MSE) pulling to local minimums
- Leads to many solutions with very low number of rules and poor fitness, which is not very useful

Modifications

1) The generational scheme is changed to an iterational scheme
2) A new Objective Scale Crowding Distance is introduced
3) A new Crowding-Based Mating heuristic is introduced
4) The population size is dynamically adjusted
5) The phenotypic copies are removed

These modifications do not interfere with the algorithm’s specific components, thus they can be easily implemented in other existent algorithms

1. The steady-state algorithm SIFT-SS

- The steady-state scheme allows new born individual to instantly participate as parents despite the size of the population
- To apply this iterational scheme in multi-objective optimization, we have adopted the MOGA-based ranking approach to stratify the pool since it is much more efficient than the NSGA-based front one to recalculate the inclusion or removing of new individuals

Generate an initial population P and evaluate P

Build a dominance rank and calculate the crowding distance for every front

While not reached the maximum number of iterations do:

- Select two parents from P (mating heuristics)
- Cross and mutate the selected parents and produce two new individuals
- Evaluate the new individuals
- Insert the two new individuals in P, rebuild the rank and update crowding distance values
- Remove the two worst individuals and adjust the population size

Output P

2. Objective Scale Crowding Distance (OSCD)

- Is it always an equally spread solution set the most representative or desired?
  - In the case of a rule-based system learning algorithm like SIFT, the expert may be more interested in obtaining more accurate solutions but highest number of rules than having very inaccurate solutions with few number of rules
  - In order to promote accuracy, the standard crowding distance is modified
  - The OSCD will be equal to the product of the traditional crowding distance CD by an Expansion Factor

$$OSCD(s) = \left(\frac{p_{mse}(s)}{n} (e_{mse} - 1) + 1 \right) CD(s)$$

3. Crowding Based Mating (CBM)

- The Crowding-Based Mating (CBM) considers the crowding distance in order to exploit the most isolated solutions

- The use of CBM in combination with OSCD guaranties that one of the two parents is likely to be an accurate and isolated solution

  1. Select the first parent at random from the first front, using a discrete probability distribution proportional to each individual crowding distance

  2. Select the second parent by binary tournament selection

- To avoid over-fitting, CBM is applied with probability P_{CBM}. Otherwise, both parents are selected by binary tournament

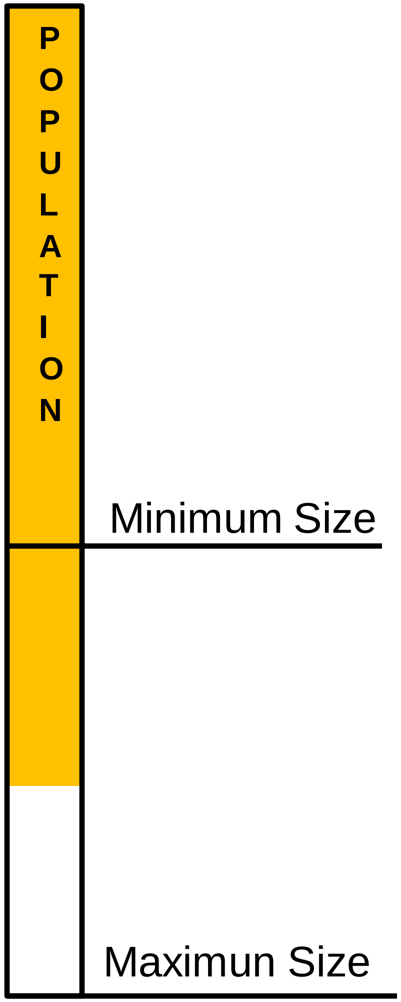
4. Variable Population Size (VPS)

- The stationary scheme allows to include a larger number of solutions without compromising too much the overall performance

- The population size can dynamically grow when all the individuals are non-dominated and dynamically shrink when dominated individuals appear

- The VPS gives some degree of flexibility to keep optimal solutions that, otherwise, would disappear

- In problems that tends to many non-dominated solutions (either because of the large search space or the use of many objectives to be optimized) this type of population size management can be profitable



5. Copies Check (CC)

- The SIFT-SS also implements a copy check routine that prevents the insertion of phenotypic copies in the population

- The decision of removing copies is founded on two aspects:

  1. A copy consumes space in the population with identical objective vectors that does not help to the exploration process

  2. In the iterational scheme, copies are not as important for the reproduction of elite individuals as they are in generational scheme

Experiments – Setup

Dataset				Semantic	
Name	N_{in}	N_{out}	N_{inst}	N_{terms}	Structure
diabetes	2	1	43	21	7*2+7
ele1	2	1	495	21	7*2+7
laser	4	1	993	27	5*4+7
ele2	4	1	1066	27	5*4+7
dee	6	1	365	37	5*6+7
concrete	8	1	1030	47	5*8+7
wankara	9	1	1609	52	5*9+7
mortgage	15	1	1049	52	3*15+7
treasury	15	1	1049	52	3*15+7
elevators	18	1	16599	61	3*18+7
compactiv	21	1	8192	70	3*21+7
ailerons	40	1	13750	127	3*40+7

- 5 fold cross-validation
- 6 different random seeds
- 30 runs per problem
- 12 data sets with up to 40 input variables and 16,599 instances
- 50,000 evaluations

Experiments – Configurations

SIFT vs 9 different configurations of **SIFT-SS**

Algorithm	P_{cbm}	E_{mse}	$OSCD$	VPS	CC	Description
sift-ss.1	-	-	no	-	no	sift-ss
sift-ss.2	-	-	no	-	yes	sift-ss + cc
sift-ss.3	-	-	no	30-60	no	sift-ss + vps 30-60
sift-ss.4	-	2x	yes	-	no	sift-ss + oscd 2x
sift-ss.5	0.5	2x	yes	-	no	sift-ss + cbm 50% + oscd 2x
sift-ss.6	1.0	2x	yes	-	no	sift-ss + cbm 100% + oscd 2x
sift-ss.7	-	2x	yes	-	yes	sift-ss + oscd 2x + cc
sift-ss.8	0.5	2x	yes	-	yes	sift-ss + cbm 50% + oscd 2x + cc
sift-ss.9	1.0	2x	yes	-	yes	sift-ss + cbm 100% + oscd 2x + cc

Performance Metric

- Generational Distance (GD) [Van Veldhuizen, 1999]

$$GD(S) = \frac{1}{|S|} \sum_{x \in S} \min \{\|f(x) - f(y)\| : y \in S^*\}$$

with *S** being the set of non-dominated solutions among all solutions examined in our computational experiments [Ishibuchi *et al.*, 2008] at each data set partition

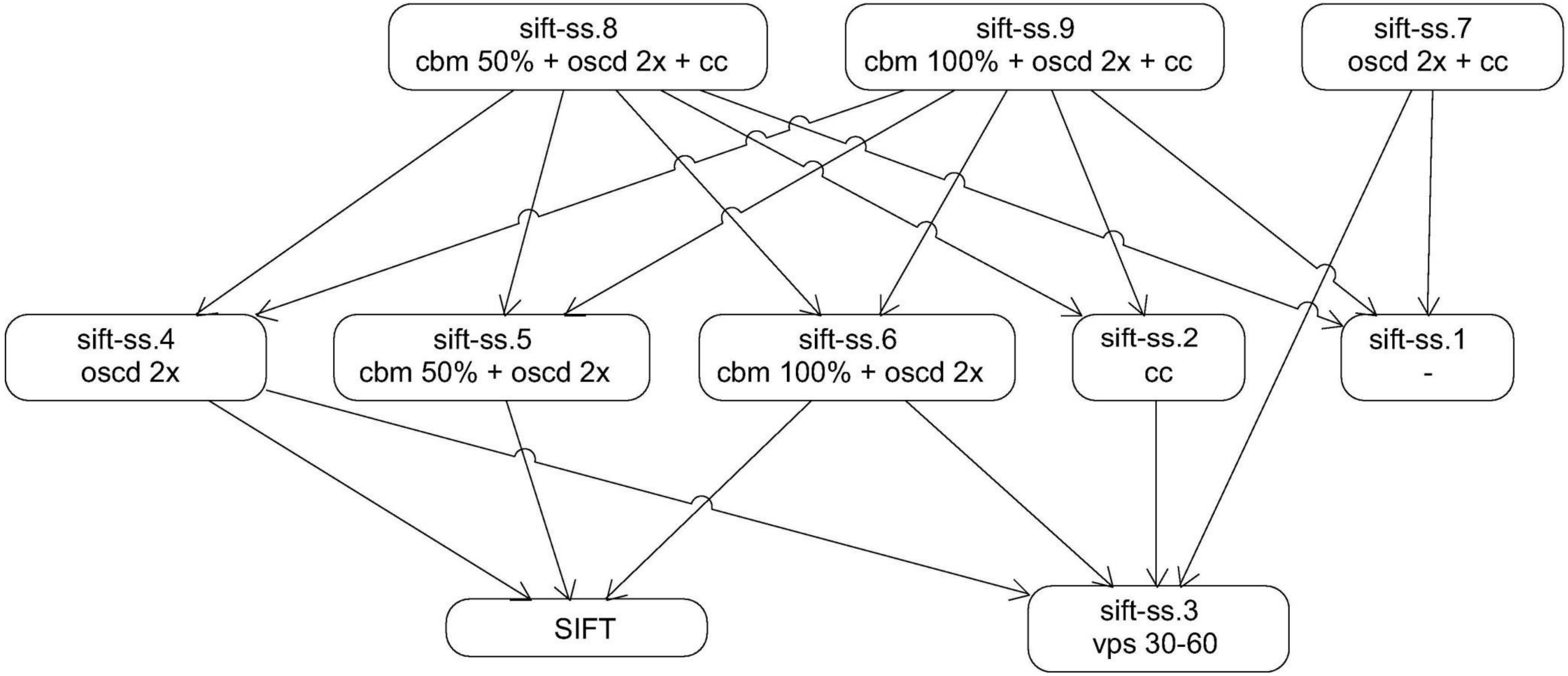
Obtained Results

Dataset	SIFT	SIFT-SS 1	2	3	4	5	6	7	8	9
diabetes	<u>0.0769</u>	0.0789	0.0746	0.0895	0.0921	0.0840	0.0862	0.0783	0.0774	0.0888
ele1	0.0619	0.0557	0.0596	0.0595	0.0576	0.0609	0.0623	0.0619	<u>0.0530</u>	0.0589
laser	0.0906	0.0911	0.0865	0.0758	0.0791	0.0816	0.0801	<u>0.0712</u>	0.0767	0.0790
ele2	0.0867	0.0682	0.0770	0.0863	0.0706	0.0671	0.0691	0.0645	0.0599	<u>0.0540</u>
dee	0.0440	0.0393	0.0370	0.0354	0.0361	0.0364	0.0373	0.0358	0.0361	<u>0.0343</u>
concrete	0.0531	0.0560	0.0587	0.0587	0.0510	0.0528	0.0574	0.0535	0.0514	<u>0.0509</u>
ankara	0.1176	0.1314	0.1119	0.1462	0.0717	0.0912	0.0771	0.0857	<u>0.0589</u>	0.0613
mortgage	0.1546	0.1921	0.1379	0.1824	0.1533	0.1171	0.1387	0.1601	0.1333	<u>0.1178</u>
treasury	0.0922	0.1091	0.1146	0.1660	0.0914	0.0862	0.0817	0.0751	0.0896	<u>0.0625</u>
elevator	0.0863	0.0808	0.0522	0.0782	<u>0.0506</u>	0.0718	0.0747	0.0668	0.0585	0.0587
compactiv	0.0471	0.0498	0.0486	0.0505	0.0507	0.0515	0.0470	0.0473	0.0424	<u>0.0405</u>
ailerons	0.0653	0.0453	0.0526	0.0562	0.0481	0.0569	0.0489	<u>0.0429</u>	0.0436	0.0473
Mean Ranks	7.50	7.00	6.17	7.50	5.17	5.83	6.08	4.33	2.67	2.75

Best performing configurations (8 and 9) are the ones who combines CBM + OSCD + CC

Friedman test: H_0 rejected with $p < 0.05$

Wilcoxon Signed-Rank Test



Each arrow indicates that there is significant difference ($p < 0.05$) between a pair.

The algorithms with the best results are ordered from the top down

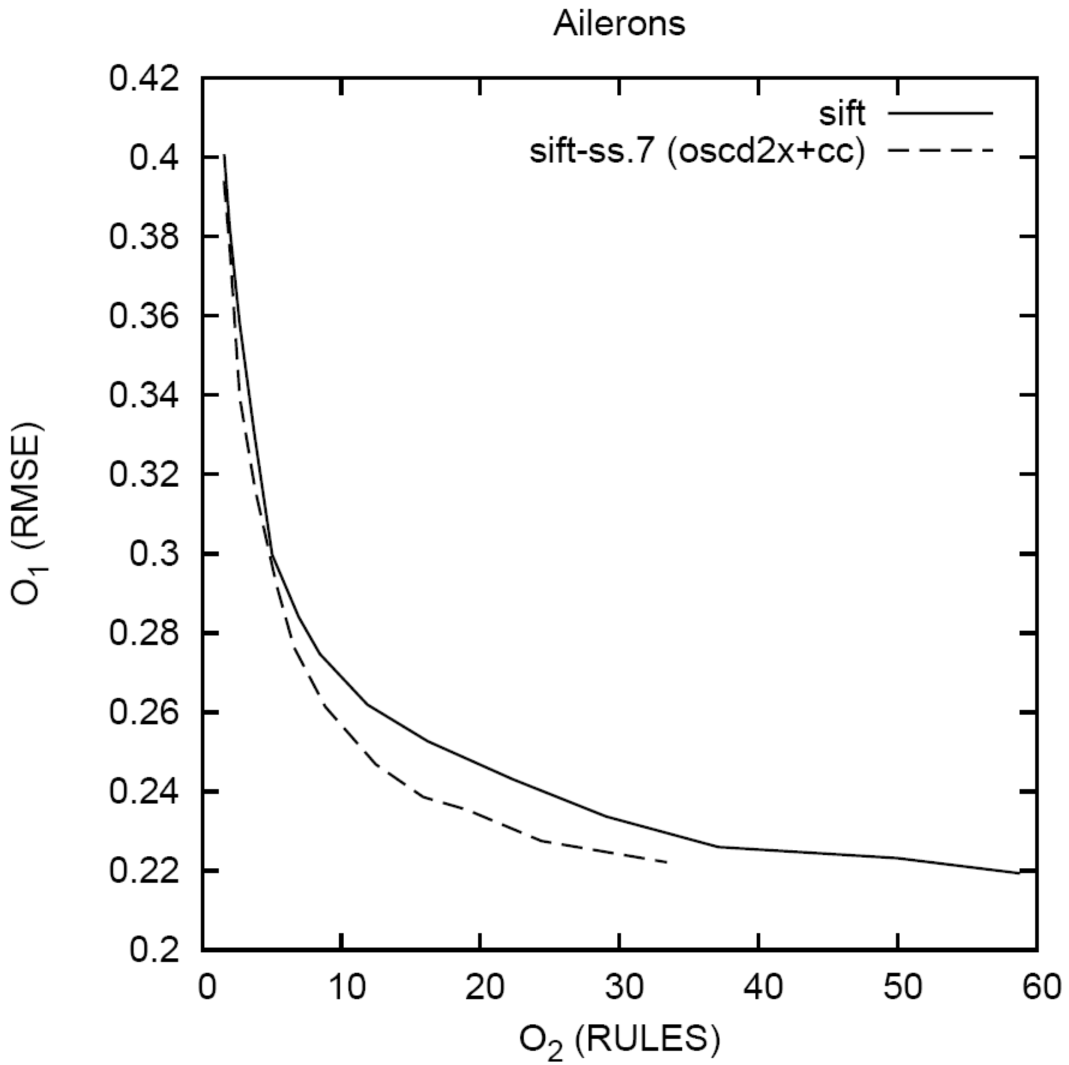
Examples of Average Pareto Fronts

SIFT

vs

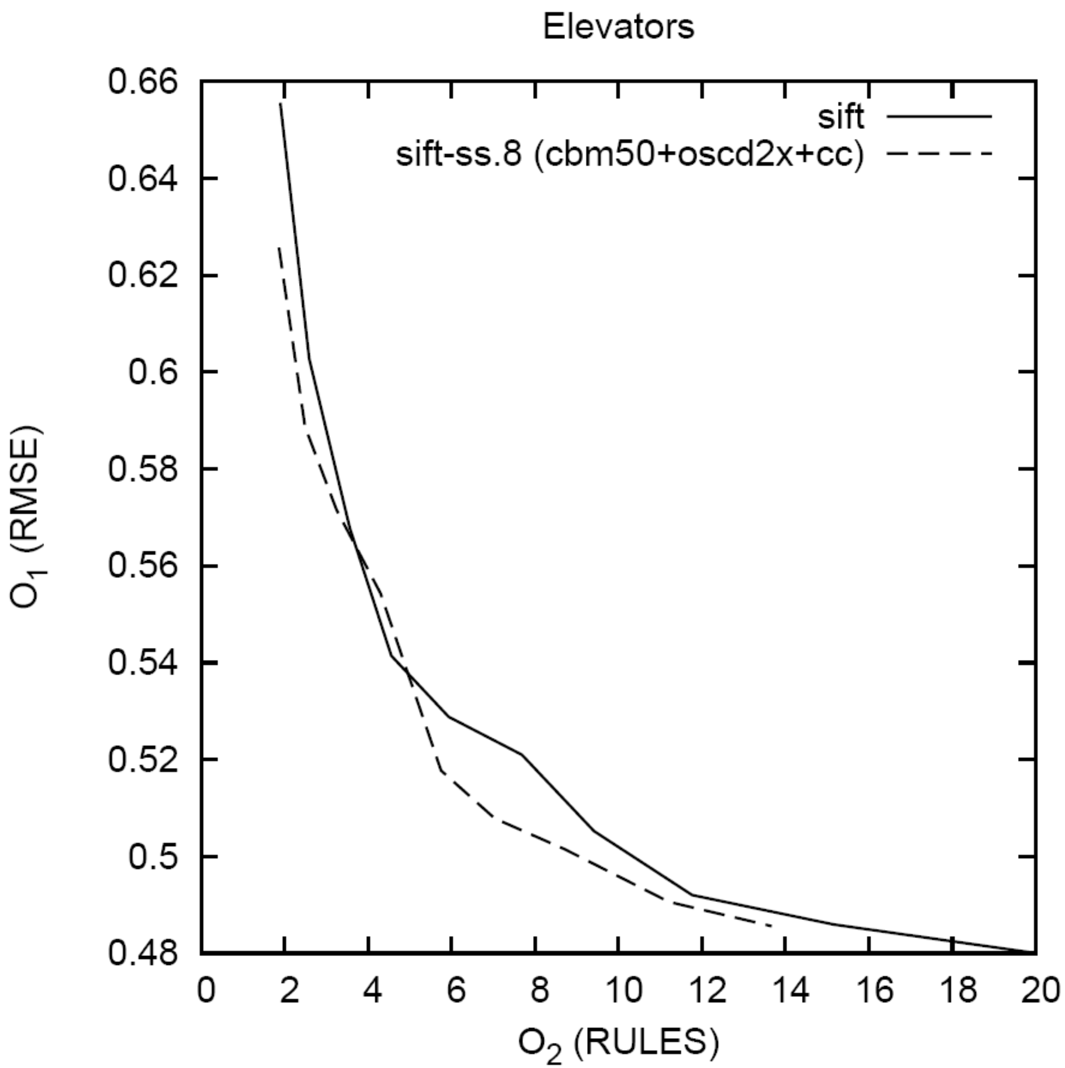
SIFT-SS.7

(OSCD2x + CC)



Examples of Average Pareto Fronts

SIFT
vs
SIFT-SS.8
(OSCD2x+CBM50%+CC)



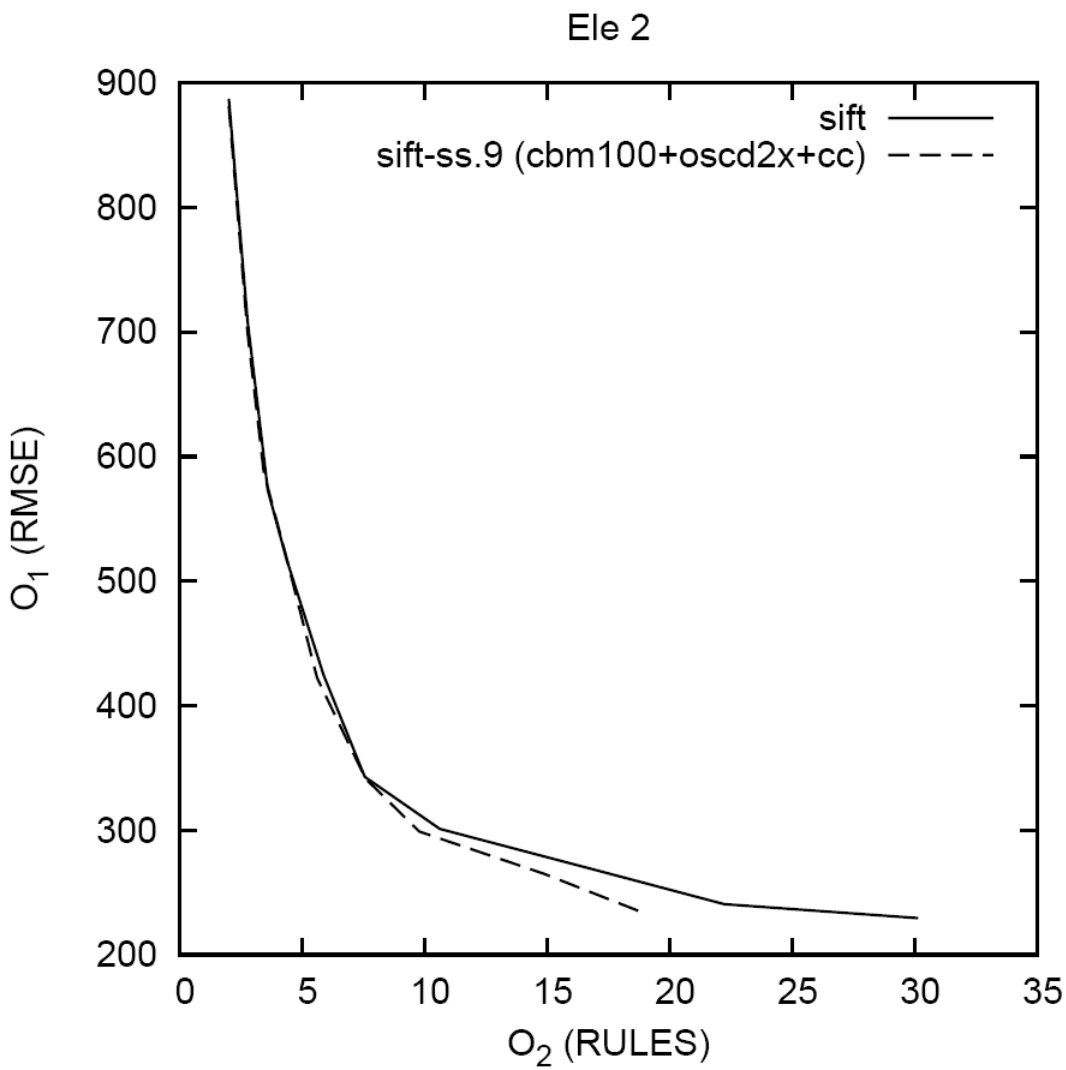
Examples of Average Pareto Fronts

SIFT

vs

SIFT-SS.9

(OSCD2x+CBM100%+CC)



Conclusion

OSCD: Objective Scale Crowding Distance
CBM: Crowding Based Mating
VPS: Variable Population Size
CC: Copies Check

- The OSCD performs really good, whether in combination with CC or alone (configurations 4 and 7)
  - The increased density in the zone of accurate solutions makes possible a substantial reduction of the number of rules
- The CBM reinforced the effect of the OSCD, but this only led to a better performance if the copies were avoided by the CC approach
  - In general, the removal of copies (CC) was found important when using OSCD or CBM as can be observed in configurations 7, 8 and 9 compared to 4, 5 and 6
- The VPS approach did not make any difference. Due to the interpretability constrains and the objectives used in SIFT, the number of non-dominated solutions was not high and the modification was not effective

Current and Future Work

- We are currently working with a richer set of optimization objectives (not always correlated among them) that allow deeper evaluation of the quality of the solutions
- This leads to a higher number of non-dominated solutions that saturates the population size. In such cases, the VPS is being found to be effective to preserve optimal solutions
- As a consequence of including new optimization objectives, we are working in many-objective optimization techniques applied to genetic fuzzy systems in order to appropriately manage the explosion of the Pareto size

References

Gacto, M.J., Alcalá, R., Herrera, F.: Adaptation and application of multi-objective evolutionary algorithms for rule reduction and parameter tuning of fuzzy rule-based systems. Soft Computing 13(5), 419–436 (2008).

Casillas, J.: Efficient multi-objective genetic tuning of fuzzy models for large-scale regression problems. In: Proceedings of 2009 IEEE International Conference on Fuzzy Systems, Jeju, Republic of Korea, pp. 1712–1717 (2009)

Van Veldhuizen, D.A.: Multiobjective Evolutionary Algorithms: Classifications, Analyses, and New Innovations. Ph.D. dissertation. Air Force Institute of Technology, Dayton (1999)

Ishibuchi, H., Narukawa, K., Tsukamoto, N., Nojima, Y.: An empirical study on similarity-based mating for evolutionary multiobjective combinatorial optimization. European Journal of Operational Research 188(1), 57–75 (2008)