COMPARISON BETWEEN MILTIOBJECTIVE OPTIMIZATION ALGORITHMS

H. E. Radhi[1], and S. Barrans[1]
[1] University of Huddersfield, Queensgate, Huddersfield HD1 3DH, UK

ABSTRACT

The first aim of this study was to perform a complete comprehensive comparison between multi-objective optimization methods by using the commercial modeFRONTIER software to solve the standard test problem SCH and FON and examining the efficiency of each method. Two numerical performance metrics and one visual criterion were chosen for qualitative and quantitative comparison : (1) Spacing metric (S) which indicates the distribution of the Pareto front in the objective space. (2) the ratio between the number of resulting Pareto front members and the total number of fitness function calculations, and lastly (3) graphical representation of the Pareto fronts for discussion. These metrics were chosen to represent the quality, as well as speed of the algorithms by ensuring well distributed solution.

Keywords Multi-objective optimization, Genetic algorithms, MOSA.

1 INTRODUCTION

Unlike single criterion optimization, in multiobjective problems it is not possible to have a single solution that optimizes all objectives, there usually exists a set of non-dominated solutions or Pareto optimal solutions. The mathematical multiobjective optimization statement can be defined as follows Deb (2001).

Minimize/ Maximize $f(x) = [f1(X), f2(x), \ldots\ldots\ldots, fn(x)]^T$ (1)

Subject to $g(x) \le 0 \qquad h(x) = 0$

$$x_{i,L} \le x_i \le x_{i,H} \qquad i = 1, 2, 3, \ldots.., n$$

Where *f(x)* is the set of objective function, *n* the total number of objective functions, *g* and *h* are vectors of inequality and equality constraint respectively and *x* is the set of design variables. Fig. 1 shows representative solutions of a multiobjective optimization problem. The dotted line represents the Pareto optimal solutions which are not dominated by any other solution, since no other solution in the set are equal or better for both objective functions. Note that solution 1 has a small value of *f1* but a large value of *f2*. Solution 5 has large value of *f1* but small value of *f2*; one cannot decide that solution 1 is better than solution 5, or vice-versa, if the goal is to minimize both objective functions. It is evident that solution 6 is not a good solution since it is dominated by 5. Deb (2001) used the concept of domination (and non-domination) to describe the Pareto front.
The solution x1 dominates a solution x2 if and only if:

- The solution *x1* is not worse than *x2* in any of the objectives.
- The solution *x2* is strictly better than solution *x2* in one objective at least.

Sedenko and raida (2010) performed a comparison between particle swarm optimization and genetic algorithms. Several multiobjective evolutionary algorithms (MOEA) were developed and compared by Deb (2001). Deb et al (2002) proposed a newer version of Non-dominated Sorting Genetic Algorithms NSGA-II and compared with a Pareto-Archived Evolution Strategy (PAES) and Strength-Pareto Evolutionary algorithm (SPEA).

2 BENCHMARK PROBLEMS

In the literature wide range of different benchmark problems with varying parameters are used to investigate the performance of multiobjective optimization algorithms. In this study two different benchmark problems were used. The Schaffer (SCH) and Fonesca (FON) problems are widely used in the field of multiobjective optimization.
The following parameters were used during this study initial population size, 100; crossover probability, 0.65; mutation probability, 0.1; and number of generation, 10. Each algorithm was allowed to run1000 function evaluation.

A. SCH problem (n=1)

This is a low dimensional convex problem suggested by Schaffer (1987).

$$SCH = Minimize\,(f1, f2) \tag{2}$$

$$f_1(x) = x^2 \tag{3}$$

$$f_2(x) = (x-2)^2 \tag{4}$$

$$x \in [-10^3, 10^3]$$

B. FON problem (n=3)

This is a problem used by Fonesca and Fleming (1998). It is characterizes by having a non-convex Pareto front and non-linear objective functions with their value concentrated around *f1 f2* = (-1, 1).

$$FON = Minimize(f1, f2) \tag{5}$$

$$f_1(x1, x2, x3) = 1 - \exp[-\Sigma_{i=1}^{3}(x_i - \frac{1}{\sqrt{3}})^2] \tag{6}$$

$$f_2(x1, x2, x3) = 1 - \exp[-\Sigma_{i=1}^{3}(x_i + \frac{1}{\sqrt{3}})^2] \tag{7}$$

3 PERFORMANCE METRICS

As mentioned before, in order to facilitate the quantitative assessment of the performance of a multiobjective optimization algorithm, the hit rate metric and spacing metric suggested by Schott (1995) were used , which measure the extent of diversity of an approximation set. The two metrics are summarized as follows.

A. Hit rate metric (HR %)

Different classifiers are used to describe the results, the number of resulting Pareto front point is given by PF, while the parameters FFC denotes the total number of fitness calculation. The final hit rate HR is computed according to the following equation

$$HR = \frac{PF}{FFC} 100[\%] \tag{8}$$

A higher hit rate indicates that fewer time consuming fitness computations were used to find the Pareto optimal solutions. The relationship between the size of the feasible design space and the ideal Pareto front should be considered in order to create a universal hit quantifier.

B. Spacing metric (S).

This metric as used by Schoot (1995), indicates how uniformly the points in the approximation set are distributed in the objective space as a variance S:

$$S = \sqrt{\frac{1}{|PA|} \sum_{i=1}^{|PA|} (di - \bar{d})^2} \tag{9}$$

$$di = \min[\sum_{m=1}^{k} |fm(ai) - fm(aj))|] \tag{10}$$

Where $\bar{d}$ is the average of all di, *k* is the number of objective functions and *PA* represents the Pareto optimal set. A zero value of this metric indicates all members of Pareto front are equidistantly spaced.

4 modeFRONTIER SOFTWARE

One of the aims of this paper was to design, optimize and compare different multiobjective optimization algorithms. For this purpose, a benchmark problem was designed in modeFRONTIER .This software package is a multiobjective optimization and multidisciplinary design optimization code written to allow easy coupling to different commercial computer aided engineering (CAE) tools.
ModeFRONTIER allows optimization analysis to be performed to modifying the values assigned to the input variables of various CAE tools, such as Finite Element Analysis (FEA), Computational Fluids Dynamics (CFD) and CAD software. The output from these CAE tools can then be described as the objectives and constraints of the design problem.
modeFRONTIER provides a GUI driven wrapper around the CAE tools. The user manual of this software illustrates how a variety of problems can be handled modeFRONTIER (2008). Handling the analysis tool within the modeFRONTIER framework is relatively straight forward with direct interfaces for Matlab and Simulink, Excel, CATIA, ANSYS Workbench and ABAQUS.

5 RESULTS AND DISCUSSIONS

Six multiobjective population-based optimization algorithms were introduced for comparison of the effectiveness of each. The six algorithms examined in this study were: Multiobjective Genetic Algorithm (MOGA-II) Silivia (2003), Adaptive range Multiobjective Genetic Algorithm (ARMOGA) Daisku (2005), Fast Multiobjective Genetic Algorithm (FMOGA-II), Non-dominated Sorting Genetic Algorithm (NSGA-II) Deb et al. (2000), Multiobjective Particle Swarm Optimization (MOPSO) Sanaz (2004) and Multiobjective Simulated Annealing (MOSA) Suppapinarm Sefan and Parks (2000).
To illustrate the performance of the six algorithms, the values obtained for the two comparison metrics are included in Table 1. For the SCH problem the solutions obtained by FMOGA-II and MOGA-II are the best regarding the spacing metric, while the solution sets obtained by NSGA-II and MOPSO are the second best with the same value of spacing metric.
In second problem (FON), it is clear that FMOGA-II is still the best with the highest hit rate and lowest value of spacing metric, MOGA-II is the second best, while MOSA is the worst case in terms of the two metrics. From the above results, it is concluded the FMOGA-II and MOGA-II algorithms are suitable for solving convex problems, whilst FMOGA-II outperforms all other algorithms regarding both spacing and hit rate for non-convex problem.
In the following section Figures 2 and 3 present the graphical results for all algorithms in the order MOGA-II, ARMOGA-II, NSGA-II, FMOGA-II, MOSA and MOPSO. This graphical representation of the Pareto optimal curve found by the six methods can be used to compare their performance. For the SCH problem Fig. 2, can be seen that the FMOGA-II and MOGA-II algorithms performed equally well. They both displayed an even distribution of points along the Pareto front. NSGA-II gave the second best results, whilst ARMOGA gave a poor distribution at one end of the curve.
For the FON problems shown in Figure 3, FMOGA-II shows a uniform distribution of points on the Pareto optimal curve. However other methods gave a poor distribution at one end of the curve such as ARMOGA, MOGA-II, NSGA-II and MOPSO or at both ends such as MOSA.

6 CONCLUSION

In this study we have performed an experimental comparison between six algorithms for multi-objective optimization. To evaluate the performance of the algorithms two well-known two objectives benchmark problems (SCH) and (FON), two quality indicators and one visual criterion are used.
According to experiments done with test functions and the parameter settings used, it can be conclude that

- In convex problem (SCH) the FMOGA-II significantly outperforms 5 other algorithms in terms of hit rate and spacing metric.
- In concave problem (FON) the performance of FMOGA-II is still very well in both metrics and has a well-distributed solution over the Pareto front.

On the whole, the FMOGA-II outperforms all other types of algorithms and suitable to use in convex concave multiobjective optimization problems.

Note: this study is a part of Ph D project (Optimization of welded joints under fatigue loading). So it is evident to classify the multiobjective optimization algorithms according to their performance and convergence for the future work of this project.

REFERENCES

DAISUKE S. (2005), *ARMOGA An efficient Multi-objective Genetic Algorithm*. ESTECO Technical Report

DEB K, (2001), *Multi-objective optimization using Evolutionary Algorithms*. John Wiley and Sons.

DEB K., AGARWAL S., PARTAP A. and MEYARIVAN T. (2000), *A Fast Non Dominated Sorting Genetic Algorithm for Multi-Objective Optimization NSGA-II*. Indian Institute of Technology.

DEB K., PARTAP A., AGARWAL S. and MEYARIVAN T. (2002), *A Fast and Elitist Non Dominated Sorting Genetic Algorithm NSGA-II*, IEEE Transactions on Evolutionary Computation, Vol. 6, pp. 182-197.

FONESCA C. M. and FLEMING P. J. (1998), *Multi-objective optimization and multiple constraints handling with evolutionary algorithms*. *Part II: Application examples*. IEEE Trans. Syst. Man, Vol. 228, pp. 38-47.

modeFRONTIER. (2008), *modeFRONTIER V4.0 User manual*. Trierste, Italy, ESTECO srl.

SANAZ M. (2004), *multi-objective Evolutionary Algorithm: Data structures, Convergence and Diversity*, Ph D thesis, University of Paderbornmr, Germany.
MODE

SCHAFER J. D. (1987), *Multi-objective optimization with vector evaluated genetic algorithms*. Paper presented in the proceeding of the 1st International Conference on Genetic Algorithms.

SCHOT J. R. (1995), *Fault tolerant design using single and multi-criteria genetic algorithms optimization.* Massachusetts Institute of Technology.

SEDENKA V. and RAIDA Z. (2010), *Critical Comparison of Multi-objective Optimization Methods: Genetic Algorithms versus Swarm Intelligence*. Radio Engineering, Vol. 19, pp. 369-377.

SILVIA P. (2003), *MOGA-II An improved Multi-objective Genetic Algorithm*: ESTECO Technical Report.

SUPPAPINARM A. STEFAN K. A. and PARKS G. T. (2000), *A simulated Annealing Algorithm for a Multi-objective Optimization. Engineering Optimization*, Vol 33, pp. 59-85.

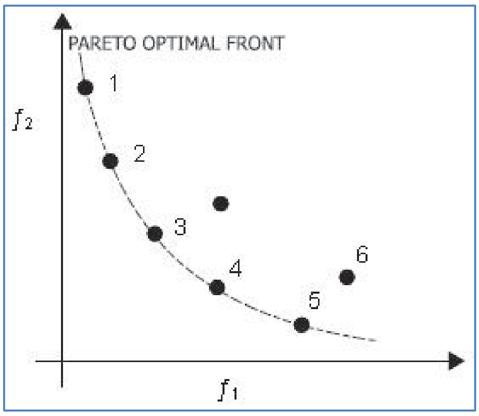

Figure 1: Main concept of Pareto dominance in a two objectives problem

Algorithm	Metrics	SCH Problem	FON Problem
MOGA-II	HR [%]	22.13E-03	92.8E-03
	Spacing(S)	4.098E-08	12.805E-03
ARMOGA	HR [%]	19.37E-03	61.7E-03
	Spacing(S)	1.64705	17.261E-03
NSGA-II	HR [%]	18.69E-03	110.9E-03
	Spacing(S)	0.25298	16.584E-03
FMOGA-II	HR [%]	29.89E-03	388E-03
	Spacing(S)	4.098E-08	2.101E-03
MOSA	HR [%]	5.33E-03	13E-03
	Spacing(S)	1.04307	64.871E-03
MOPSO	HR [%]	10.21E-03	42.3E-03
	Spacing(S)	0.26666	16.467E-03

Table 1: Performance measures of MOGA-II, ARMOGA, NSGA-II, FMOGA-II, MOSA and MOPSO for test problems considered in study showing the value of spacing (S) and hit rate metric (HR)

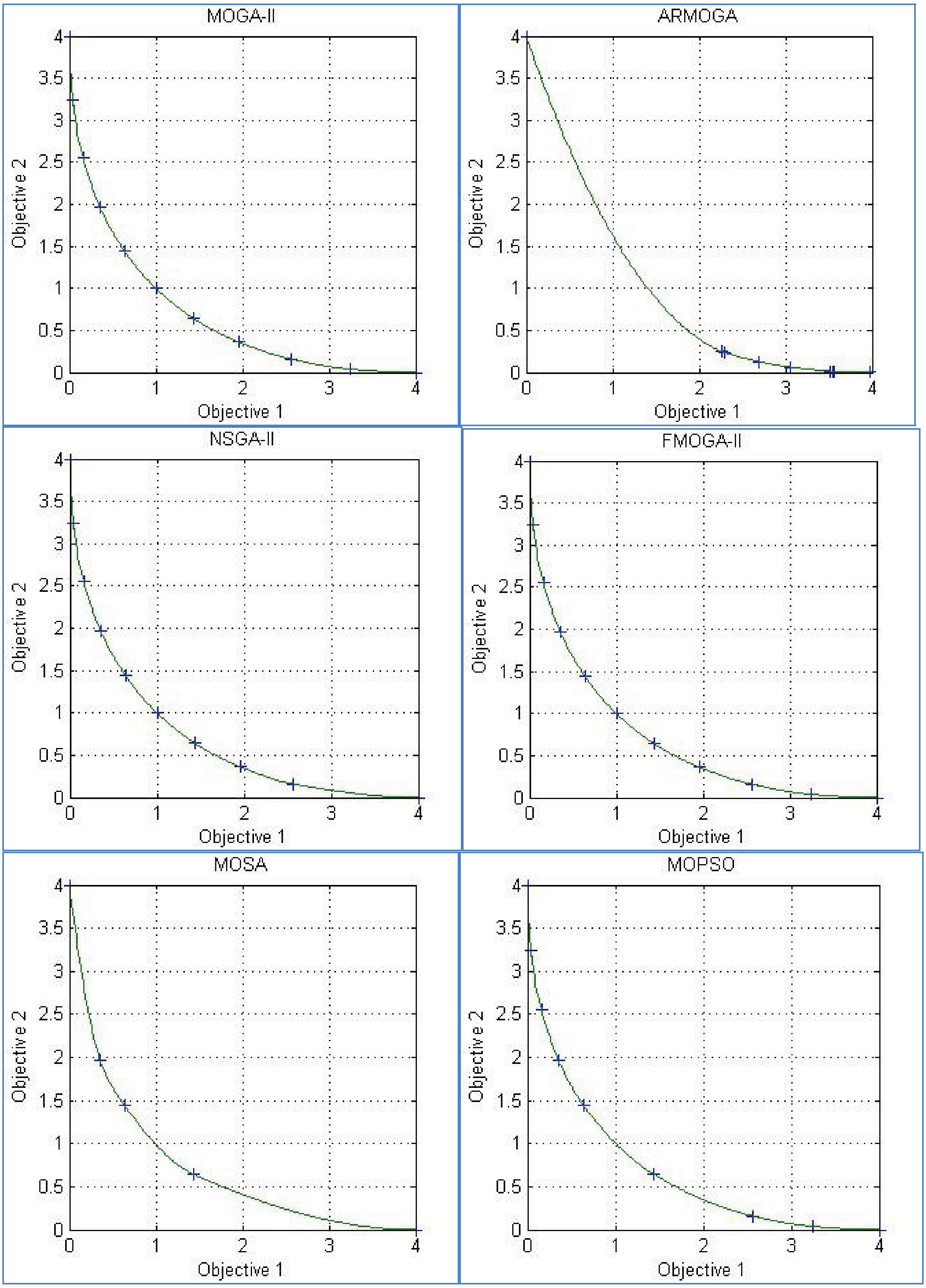

Figure 2: The evaluated front from MOGA-II, ARMOGA, NSGA-II, FMOHA-II, MOSA and MOPSO from SCH problem.

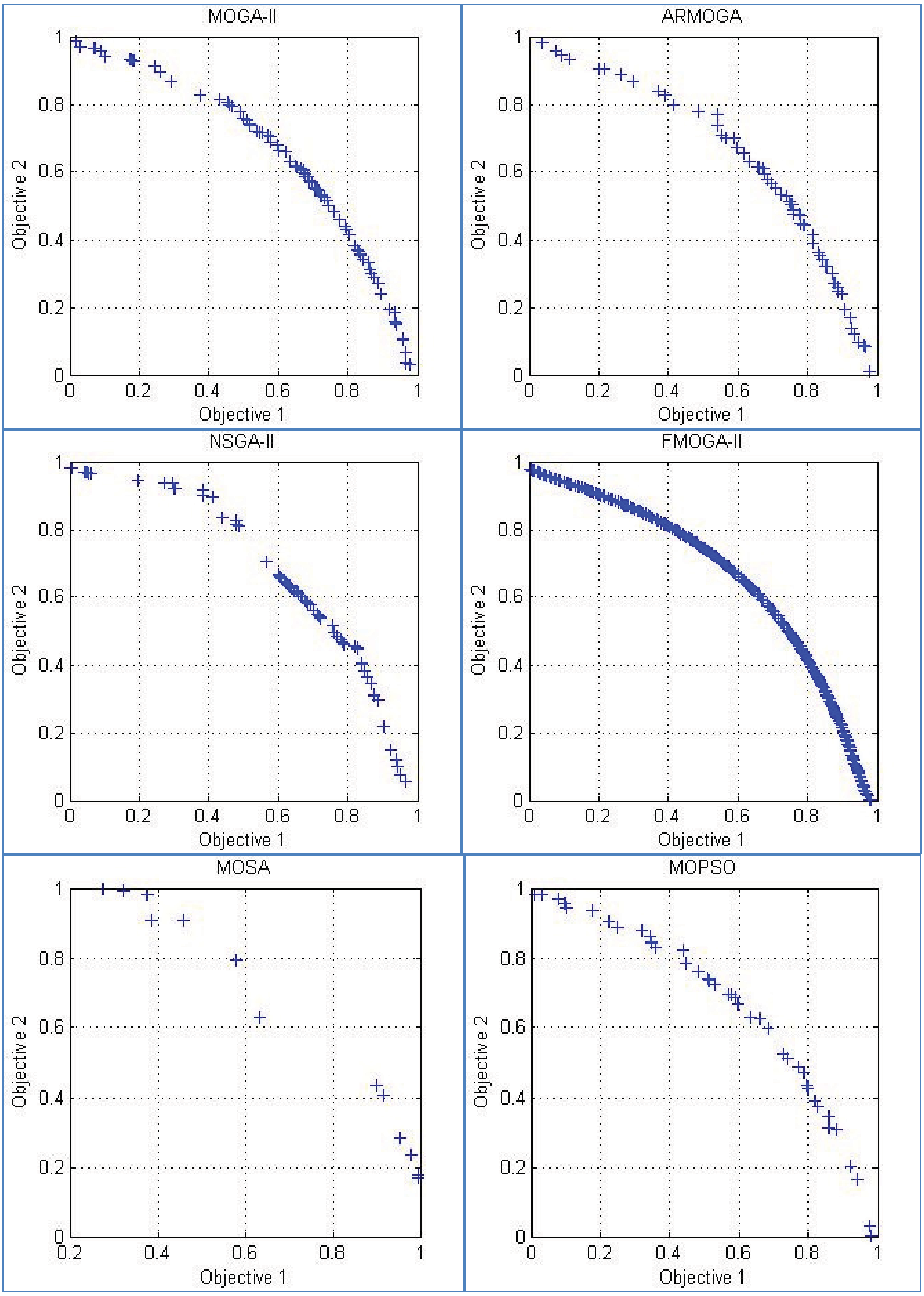

Figure 3: The evaluated front from MOGA-II, ARMOGA, NSGA-II, FMOHA-II, MOSA and MOPSO from FON problem.