Порівняння алгоритмів багатокритеріальної оптимізації

Г.Е. Раді[1], С.М. Барранс[2]

Факультет обробки даних та інженерії, Університет Гаддерсфілду, Гаддерсфілд HD1 3DH, ВЕЛИКОБРИТАНІЯ
[1]E-mail: h.radhi@hud.ac.uk, [2]E-mail: s.barrans@hud.ac.uk

При розв'язанні задач багатокритеріальної оптимізації неможливо одержати єдиний розв'язок, який б оптимізував всі критерії, тут зазвичай існує набір недомінованих розв'язків або оптимальних розв'язків Парето.

Першочерговою метою цього дослідження є повне та вичерпне порівняння методів багатокритеріальної оптимізації за допомогою стандартних тестових задач SCH та FON з використанням програмного забезпечення modeFRONTIER для визначення ефективності кожного методу. Щоб оцінити ефективність і дієвість кожного алгоритму, було обрано тестові задачі з літератури.

Два цифрові показники та один візуальний критерій були взяті для якісних і кількісних порівнянь:

(1) показник інтервалу (S), що відображає розподіл Парето-фронту в цільовому просторі;

(2) співвідношення між кількістю отриманих членів Парето-фронту та загальною кількістю розрахунків функції пристосовуваності, яка визначається коефіцієнтом ефективності пошуку [1], і

(3) графічне вираження Парето-фронтів для обговорення. Ці показники були вибрані для відображення якості, а також швидкості алгоритмів в забезпеченні поширених розв'язків.

Деб [2] використовує термін домінування (та недомінування) для опису Парето-фронту. Розв'язок *x1* домінує над розв'язком *x2*, якщо і тільки якщо:

- Розв'язок *x1* не є менш важливий ніж *x2* за будь-яким з критеріїв.
- Розв'язок *x2* є значно важливіший ніж розв'язок *x2* принаймні за одним критерієм.

Седенко та Райда провели порівняння між методом рою часток та генетичними алгоритмами [1]. Було розроблено та проведено порівняння між кількома багатокритеріальними еволюційними алгоритмами (БКЕА) [2], Деб та ін. запропонували нову версію назви – Недоміновані Сортувальні Генетичні Алгоритми НСГА-II і порівняли PAES із SPEA [3].

Одним із завдань цієї роботи є створення, оптимізація та порівняння різних багатокритеріальних алгоритмів. Для цього за допомогою програмного забезпечення modeFRONTIER створено типову задачу. Багатокритеріальна оптимізація, а також оптимізація міждисциплінарних розробок забезпечують легке підключення до різноманітного комерційного комп'ютеризованого інженерного обладнання.

Comparison between Multiobjective Optimization Algorithms

H.E. Radhi[1], S.M. Barrans[2]

School of Computing and Engineering, University of Huddersfield, Huddersfield HD1 3DH, UK
[1]E-mail: h.radhi@hud.ac.uk, [2]E-mail: s.barrans@hud.ac.uk

The first aim of this study was to perform a complete comprehensive comparison between multi-objective optimization methods by using standard test problems SCH and FON by using modeFRONTIER software, to determine the efficiency of each method. In order to measure the effectiveness and competence of each algorithm, the test problems were chosen from the literature.

One of the aims of this paper is to design, optimize and compare different multi-objective optimization algorithms. For this purpose, a benchmark problem is designed in modeFRONTIER. It is a multi-objective optimization and multidisciplinary design optimization written to allow easy coupling to different commercial computer aided engineering (CAE) tool.

Keywords – Multi-objective optimization ,Genetic algorithms, MOSA.

I. Introduction

The mathematical multi-objective optimization statement can be defined as follows [2].
Minimize / Maximize

$$f(x) = [f1(x), f2(x), \ldots\ldots, fn(x)]^T \qquad (1)$$

$$\textit{Subject to} \quad g(x) \le 0 \quad h(x) = 0$$

$$x_{i,L} \le x_i \le x_{i,H}$$

$$i = 1,2,3,\ldots\ldots,n$$

Where $f(x)$ is the objective function, n the total number of objective functions, while g and h are vectors of inequality and equality constraint respectively, x the set of design variables. Fig. 1 shows the solutions of multi-objective optimization problem. The dotted line represents the Pareto optimal solutions which are not dominated by any other solutions, since no other solutions 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 $f1$; one cannot decide that solution 1 better than solution 5, or vice-versa, if the goal is to minimizing both objective functions. It is evident solution 6 not good solution, since it is dominated by 5.

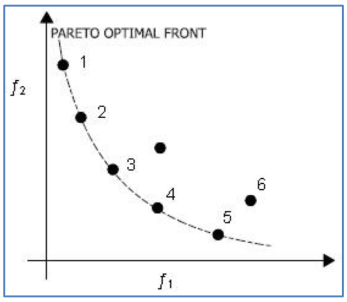

Fig. 1. Main concept of Pareto dominance in a two objective problem

II. Benchmark Problems

The measurement of performance of multi-objective optimization algorithms is done by applying them to two types of benchmark problems, in literature, there is wide range of different problems with varying parameters. In this study two different benchmark problems are used. SCH and FON are widely used problems in the field of multi-objective optimization.

The following parameters are used during this study initial population size, 100; crossover probability, 0.65; mutation probability, 0.1; and number of generations, 10.

A- SCH problem (n=1)

It is a low dimensional problem suggested by [4]. Each algorithm is allowed to run 1000 function evaluations

$$SCH = Minimize(f1, f2) \quad (2)$$

$$f_1(x) = x^2 \quad (3)$$

$$f_2(x) = (x-2)^2 \quad (4)$$

$$x \in [-10^3, 10^3] \quad (5)$$

B- FON problem (n=3)

It is a problem used by [5]. It is characterizes by non-convex Pareto front and non-linear objective functions with their value concentrated around *f1 f2*= (1, 1).

$$FON = Minimize(f1, f2) \quad (6)$$

$$f_1(x1, x2, x3) = 1 - \exp[-\Sigma_{i=1}^{3}(x_i - \frac{1}{\sqrt{3}})^2] \quad (7)$$

$$f_1(x1, x2, x3) = 1 - \exp[-\Sigma_{i=1}^{3}(x_i + \frac{1}{\sqrt{3}})^2] \quad (8)$$

III. Performance Metrics

As we mention before, in order to compare the performance of multi-objective optimization, we adopt the hit rate metric and spacing metric suggested by [6], which measure the extent of diversity of an pproximation 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 is given by PF, while the parameter FFC denotes the total number of fitness calculation.

The final hit rate HR is computed according to the following equation

$$HR = \frac{PF}{FFC} 100[\%] \quad (9)$$

The higher hit rate indicates that the less of time consuming fitness computation were used to find Pareto optimal solutions, with each solution only the hit rate have to be compared. The relation between the size of feasible design and ideal Pareto should be considered in order to create a universal hit quantifier

B - Spacing Metrics (S)

The same metric that were used by Schoot are used and they are presented in [6], to measure how the points in the approximation set are distributed in the objective space

$$S = \sqrt{\frac{1}{|PA|} \sum_{i=1}^{|PA|} (di - \bar{d})^2} \quad (10)$$

$$di = \min[\sum_{m=1}^{k} |fm(ai) - fm(aj)|] \quad (11)$$

Where $\bar{d}$ is the average of all *di*, *k* is the number of objective functions and PA represents the Pareto optimal set.

IV. ModeFRONTIER Software

modeFRONTIER allows an environment, which provides product engineers and designers to integrate their various CAE tools, such as Finite Element Analysis, Computational Fluid Dynamics (CFD) and CAD software. To perform optimization by modifying the value assigned to the input variables, and computing the outputs as they can be described as objectives and constraints of the design problem.

modeFRONTIER provides a GUI driven wrapper around the CAE tool. The user manual of modeFRONTIER illustrates how given problem can be handled [7]. Handling the analysis tool within the modeFRONTIER framework is slightly straight forward with direct interfaces for Mat lab and Simulink, Excel, CATIA, ANSYS Workbench and ABAQUS

V. Results and Discussions

Six multiobjective population-based optimization algorithms are introduced for comparison and the effectiveness of each. The six algorithms examined in this study are namely: Multiobjective Genetic Algorithm (MOGA-II)[8], Adaptive range Multiobjective Genetic Algorithm (ARMOGA)[9], Fast Multiobjective Genetic Algorithm (FMOGA-II), Non-dominated Sorting Genetic Algorithm (NSGA-II)[10], Multiobjective Particle Swarm Optimization (MOPSO)[11] and Multiobjective Simulated Annealing (MOSA)[12].

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

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

To illustrate the working of six algorithms, we have included in Table 1 the obtained value of two comparison metrics, for SCH problem as we can see those obtained by FMOGA-II and MOGA-II are the best choice when regarding the spacing metric, while the solution sets

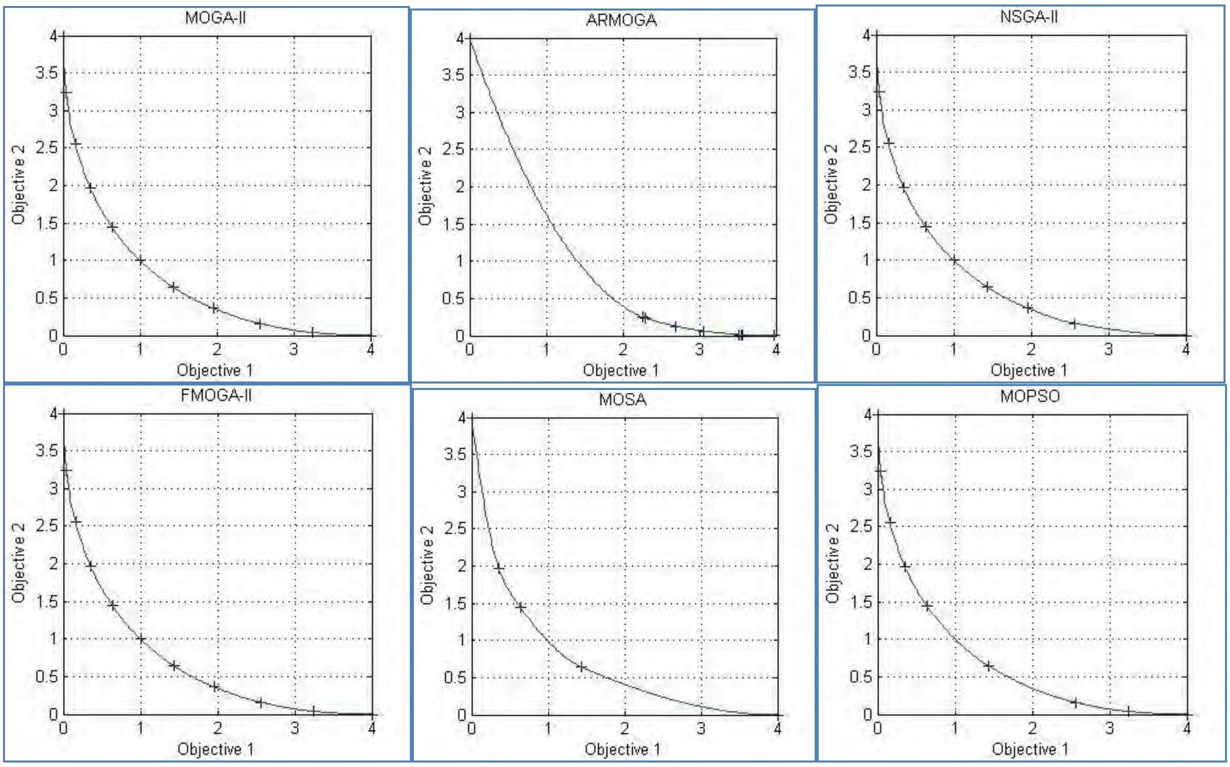

Fig. 2: The evaluated front from MOGA-II, ARMOGA, NSGA-II, FMOGA-II, MOSA and MOPSO for SCH problem.

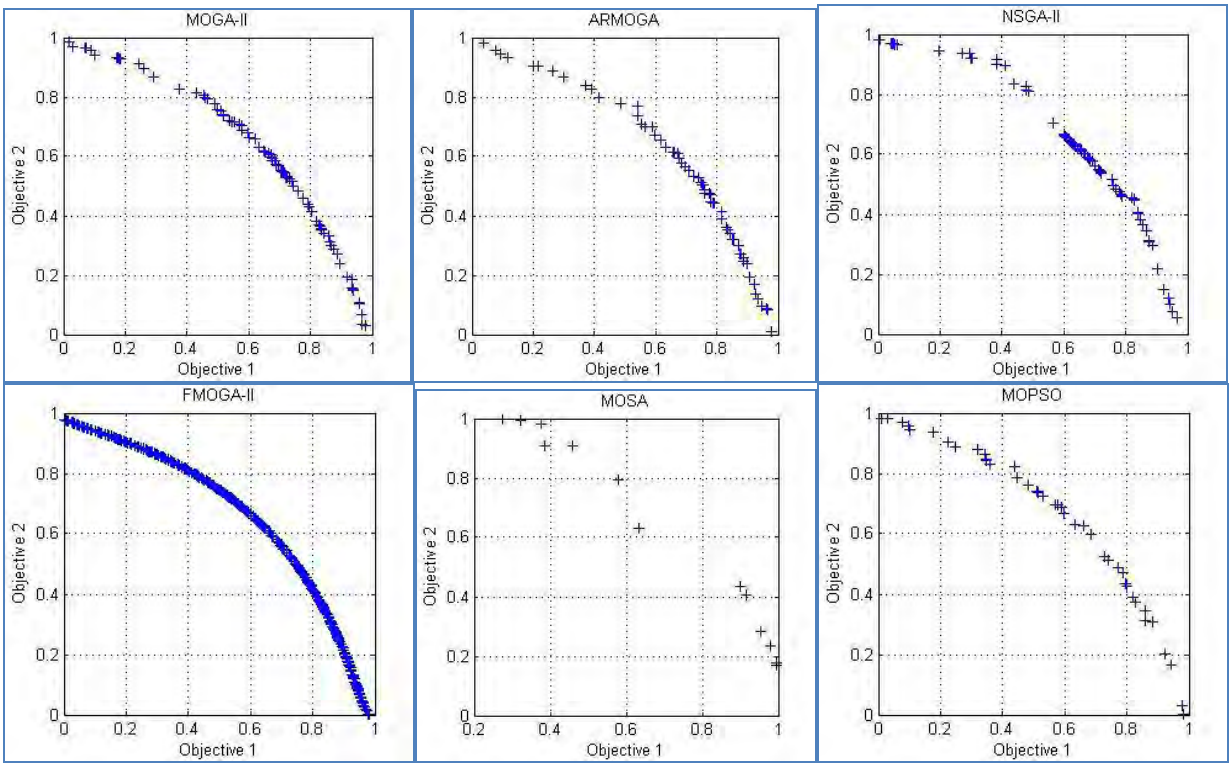

Fig. 3: The evaluated front from MOGA-II, ARMOGA, NSGA-II, FMOGA-II, MOSA and MOPSO for FON problem.

obtained by NSGA-II and MOPSO are the second best with the same value of spacing metric.

In second problem (FON) , it is clear to notice that FMOGA-II still the best one with highest hit rate and lowest value of spacing metric, MOGA-II is the second , while the MOSA is the worst case in terms of two metrics.

From the above results, it is concluded the FMOGA-II and MOGA-II are suitable to solve convex problem; on the other side the FMOGA-II outperforms all algorithm regarding spacing and hit rate for no convex problem.

In the following section Fig. 2-3 present the graphical results for all algorithms in the order of MOGA-II, ARMOGA, NSGA-II, FMOGA-II, MOSA, and MOPSO. By using graphical representation of Pareto optimal curve found by six methods to compare their performance, in SCH problem Fig. 2, it is evident to notice that the FMOGA-II and MOGA-II model performed equally well they both displayed a better distribution of the Pareto front, NSGA-II the second best, on the other side ARMOGA gave a poor distribution at one end of the curve.

For FON problems Fig. 3, it is clear to mention FMOGA-II shows a uniform distribution of a 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 end such as MOSA.

Conclusion

The first conclusion can be drawn from this study is that the FMOGA-II algorithm is always the best regarding the performance metrics and graphical distribution of Pareto front in both types of problems, another significant point is that the behavior of some algorithms is changed depending on nature of problem.

References

[1] V. Sedenka and Z. Raida, "Critical comparison of multi-objective optimization methods: genetic algorithms versus swarm intelligence," Radio-engineering, vol. 19. no. 3, pp. 369-377, 2010.

[2] K. Deb, Multi-objective optimization using Evolutionary Algorithms. John Wiley and Sons, 2001

[3] K. Deb, A. Partap, S. Agarwal, and T. Meyarivan, "A Fast and Elitist Multi-objective Genetic Algorithm: NSGA-II., IEEE Transactions on evolutionary computationvol. 6. pp. 182-197.

[4] J. D. Schaffer, "Multiple objective optimization with vector evaluated genetic algorithms, " In proceedings of the 1st International Conference on Genetic Algorithms, J. J. Crefensttete, Ed. Hillsdale, NJ: Lawrence Erlbaum, 1987, pp. 93-100.

[5] C. M. Fonseca and P. J. Fleming, "Miltiobjective optimization and multiple constraint handling with evolutionary algorithms. Part II: Application examples, " IEEE Trans. Syst, Man Cybern. A, vol. 28, pp. 38-47, Jan 1998.

[6] J. R. Schott, "Fault tolerant design using single and multicriteria genetic algorithms optimization, " Master thesis , Massachusetts Institute of Technology, 1995.

[7] ModeFRONTIER v4.0 " User manual, " ESTECO srl, Trieste, Italy, 2008.

[8] P. Silvia, "MOGA-II An improved Multiobjective Genetic Algorithmnn, " ESTECO Technical Report, 2003.

[9] S. Daisuke, "ARMOGA An efficient Multiobjective Genetic Algorithm, " ESTECO Technical Rerporet, 2005.

[10] K. Deb, S. Agarwal, A. Partap and T. Meyarivan, "A Fast Elitist Non Dominated Sorting Genetic Algorithm for Multiobjective Optimization NSGA-II, " Indian Institute of Technology, 2000.

[11] M. Sanaz, "Multiobjective Evolutionary Algorithm: Data structures, convergence and diversitym, " Sharker Verlag, Ph D Thesis, 2004.

[12] Suppapinarm, K. A. Seffen and G. T. Parks, "A Simulated Annealing Algorithm for a Multiobjective Optimization, "Engineering Optimization, vol. 33, pp. 59-85.