2011 International Conference on Information and Electronics Engineering
IPCSIT vol.6 (2011) © (2011) IACSIT Press, Singapore

A Chaos Search for Multi-Objective Memetic Algorithm

Paranya Ammaruekarat [1+] and Phayung Meesad [2]

[1] Department of Information Technology, King Mongkut's University of Technology North Bangkok, Bangkok, Thailand.
[2] Department of Teacher Training in Electrical Engineering, King Mongkut's University of Technology North Bangkok, Bangkok, Thailand.

Abstract. In this research, the efficacy improvement of Multi-Objective Memetic Algorithms (MOMAs) was studied. For this matter, a Local Search referred to as Chaos Search is incorporated with the MOMA to achieve a better search result. The resulting method is called Multi-Objective Chaos Memetic Algorithm (MOCMA). In this research, such technique is applied to solve Multi-Objective equations i.e., DTLZ 1-4 type 2 objectives. The resulted outcomes will then be measured for their capabilities to find out the best outcome group in 2 aspects, namely the capability to converge to the true outcome and the capability to spread the outcome groups found. The capabilities of the technique are then compared with existing Multi-Objective genetic local search (MOGLS), a highly efficient Multi-Objective Memetic Algorithms. This research shows that the Convergence Measurement of MOCMA process has a better rate of convergence than that of MOGLS. Also the Spread Measurement of MOCMA is scattered more evenly than that of MOGLS. It can be seen that the value of MOCMA's capability measurement in both aspects are closer to 0 than those of MOGLS.

Keywords: Multi-Objective Memetic algorithm, Local Search, Chaos Search

1. Introduction

Many problems in the real world, such as industry or academia, are multi-objective problems. As a general example, two common goals in product design are certainly to maximize the quality of the product and to minimize its cost. Compared with single objective, multi-objective optimizing problems become more complex. Because these optimized goals are generally conflicting and competing, they give rise to a set of optimal non-dominated solutions (or Pareto-optimal solutions) for the decision maker, instead of a single optimal solution.

Multi-objective optimization attracts more and more researchers' attention. With the introduction and improvements of evolutionary algorithms, it has become an important technology of solving multi-objective optimization problems. Multi-objective evolutionary algorithms commonly known include Vector Evolution Genetic Algorithm (VEGA), Multi-Objective Genetic Algorithm (MOGA), and non-dominated Sorting Genetic Algorithm (NSGA). Subsequently, there many algorithms were presented such as Niched Pareto Genetic Algorithm (NPGA), Pareto Achieved Evolution Strategy (PAES), Strength Pareto Evolutionary Algorithm (SPEA), and Particle Swarm Optimizer (PSO) etc. Some other new approaches have been introduced which provide fast and effective approximations to the Pareto front for a variety of benchmark problems.

Memetic Algorithm (MA) is one of evolutionary algorithm in combinatorial optimization. Recently memetic algorithms have been applied to multi-objective optimization problems for efficiently finding their Pareto-optimal or near Pareto-optimal solutions. Memetic Algorithms has been very popular from the research be applied to solve problems as shown in such research [1], [2], [3]. However, there are not many studies on Memetic Algorithms for multi-objective optimization [4]. The first MOMA called a Multi-Objective Genetic Local Search (MOGLS) algorithm was proposed by Ishibuchi and Murata in 1996 [5]. A variant of MOGLS with higher search ability was proposed by Jaszkiewicz [6]. It is shown that the MOGLS outperformed other MOMAs (i.e., M-PAES [7] and the original MOGLS).

In the education and development to leverage efficiency, the MOMAs has been studied. As shown in research such as [8, 9], the results from past research can enhance the efficiency of estimation solutions. The best answer of the Memetic Algorithms is better than previous algorithms but they still have disadvantages in terms of use longer searching time and some algorithms is not suitable for problems with many variables.

+ Corresponding author. Tel.: + 66 86768-4979; fax: + 662-912-2019.
E-mail address: sutthaya@hotmail.com, pym@kmutnb.ac.th

Therefore, this research will focus on the Multi-Objective Memetic Algorithms's optimization with Chaos Search, which is capable of finding the best answer of the function. The Chaos Search allows getting answer near global optimum easier because it can help avoid local optimum thus it helps improve the search better and more suitable to the problem with many variables.

2. Background

Here we briefly describe the background required for this paper.

2.1. Optimization of Multiple Objectives

Multi-Objective Optimization consisting of *m* Objective and Decision Variables, generally written as per below; [10]

Minimize (or maximize) **f(x)** = $\{f_1(\mathbf{x}), f_2(\mathbf{x}), ..., f_m(\mathbf{x})\}$.

Where x represents Decision Variables' Vector

$f_i(\mathbf{x})$ represents Objective's function i, i = {1,2, ..., m}.

g (x) represents Limit Vector .

The solution of the equation is referred to as Pareto front or set of uncovered results as shown in fig. 1.

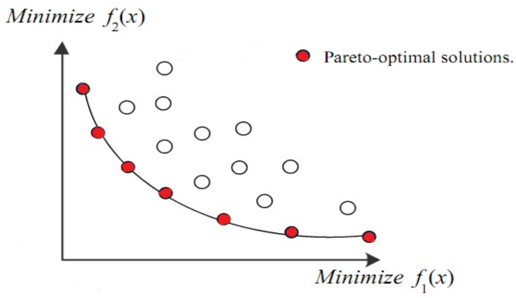

Fig. 1: Pareto Front on the domain of objective functions 1 and 2.

2.2. Memetic Algorithms : MA

MAs are inspired by Dawkins' notion of a meme [11]. MAs are similar to GAs but the elements that form a chromosome are called memes, not genes. The unique aspect of the MAs is that all chromosomes and offsprings are allowed to gain some experience, through a local search, before being involved in the evolutionary process [12]. As such, the term MAs is used to describe GAs that heavily use local search [13].

Similar to the GAs, an initial population is created at random. Afterwards, a local search is performed on each population member to improve its experience and thus obtain a population of local optimum solutions. Then, crossover and mutation operators are applied, similar to GAs, to produce offsprings. These offsprings are then subjected to the local search so that local optimality is always maintained.

2.3. Chaos theory

Chaos is a kind of universal nonlinear phenomena in all areas of science, which occurs in many systems [14]. It is widely known as its important dynamic properties: ergodicity, intrinsic stochastic property and the sensitive dependence on initial conditions. The randomness of chaos is a result of the sensitivity of chaotic systems to the initial conditions, so the chaos movement could go through every state in certain scale according to its own regularity and ergodicity [15]. It could be introduced into the optimization strategy to accelerate the optimum seeking operation and find the global optimal solution.

The chaos search algorithm was proposed in literature [16]. The main idea of chaos search is to set up an optimal point set in the search process and make the probability that the global optimal solution is in the boundary (envelop) of optimal point set. The chaos search is described as follows:

Considering the well-known logistic equation :

$$z^{k+1} = \mu z^k (1 - z^k) \quad , k = 0,1,\ldots \tag{1}$$

Where $z^k \in [0,1]$ is chaos variable on the k^{th} iteration . When $\mu = 4$, $z^0 \in \{0, 0.25, 0.5, 0.75, 1\}$

Decision vectors of the final parent population at the first stage : $x_i^* = (x_{i1}^*, x_{i2}^*, ..., x_{in}^*)$, i = 1,…,N define the domain for each individual of population at the Second stage : ε is specified for the radius of neighborhood. When $x_{ij}^{*} - \varepsilon < a_j , x_{ij}^{*} + \varepsilon > b_j$ Let $x_{ij}^{*} - \varepsilon = a_j , x_{ij}^{*} + \varepsilon = b_j$ In the following , Describe the process of chaos search.

Step1 : Perform chaos map by applying the iteration operator define by (1)

$$cx_{ij}^{k+1} = 4cx_{ij}^{k}(1 - cx_{ij}^{k}) \qquad I = 1,,N \ , \ j = 1,..,n \tag{2}$$

Step2 : Chaos variable cx_{ij}^{k+1} is mapped into the variance range of optimization valuable $[a ,b]$ x_{ij}^{k+1} by

$$x_{ij}^{k+1} = a_j + (b_j - a_j)cx_{ij}^{k+1} \tag{3}$$

and to get equation by $x_{ij}^{k+1} = x_{ij}^{*} - \varepsilon + 2\varepsilon cx_{ij}^{k+1}$ (4)

Step3 : Set the number of point N in the optimal points set. Initialize the optimal points set by choosing first N chaos iterated points.

Step4 : Define x_{ij}^{k+1} as the k^{th} iterated result. Define $X_{max}*$ as the maximum value of function in the optimal points set. Let $X*$ as the minimum point in the optimal points set. $f* = f(X*)$

Step5: Set iteration number as $k = 1$. Do $x_{ij}^{k+1} = x_{ij}^{*} - \varepsilon + 2\varepsilon cx_{ij}^{k+1}$

If $f(x_{ij}^{k+1}) < f_{max}^{*}$ then set $X_{max}* = x_{ij}^{k+1}$

Else if $f(x_{ij}^{k+1}) \geq f_{max}^{*}$ then give up the k^{th} iterated

Result x_{ij}^{k+1}

$k = k + 1$ Loop runs until f_{max}^{*} is not improved after N searches.

Step6: Choose the boundary (envelop) value of optimal point set as the new search range. Perform the range transformation. Chaos variable is mapped into the new search range by (4). If the precision could not be satisfied, then go to step 5.

Step7: If the precision of search result satisfies the stop condition, stop search process and put out the minimum point in the optimal point set $X*$ as the best solution.

3. Multi-Objective Chaos Memetic Algorithm : MOCMA

In the following, we propose a novel iterative local search procedure, called the Chaos Search, which is designed to be used within a memetic strategy. The integration of the chaos search into MOMAs and the resulting modifications will be addressed in the next section.

The method used in the experiment as follows.

- Local search : Chaos search
- Selection : Tournament Selection
- Crossover : Modified Order Crossover
- Mutation : Insertion Mutation
- Fitness Assignment : Pareto Ranking Approach

Parameters in the experiments in Table 1.

Table 1: parameters used in the experiment.

Parameter	Value
Population size	100
Number of Generation	500
Crossover probability	0.9
Mutation probability	0.5
Local search probability	0.8

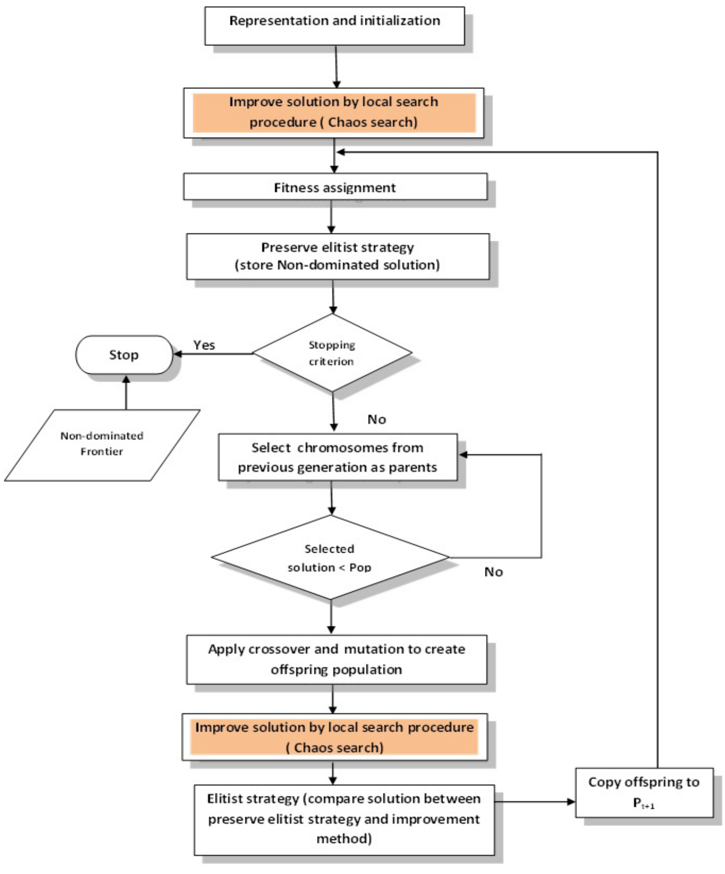

Fig. 2: Multi-Objective Chaos Memetic Algorithm

4. Simulation Result

Experiments were conducted to compare the performance of this algorithm with MOGLS.

4.1 Test Problems

In this paper, the so called DTLZ (Suite of continuous test problems by Deb, Thiele, Laumanns, Zitzler) test problems. Problems are chosen as the benchmark problems to measure the algorithms' performance. These problems are described in [17].

4.2 Comparison Metrics

- Convergence measurement [2] The convergence of the obtained Pareto-optimal solution towards a true Pareto-set ($A*$) is the difference between the obtained solution set and the true-Pareto set. Mathematically, it is defined as (5) and (6).

$$\text{Convergence(A)} = \frac{\sum_{i=1}^{|A^*|} dt_i}{|A^*|} \tag{5}$$

$$dt_i = \min_{j=1}^{|A^*|} \sqrt{\sum_{k=1}^{k} \left[\frac{f_k(x) - f_k(y)}{f_k^{\max} - f_k^{\min}} \right]^2} \tag{6}$$

Where $|A^*|$ is the number of elements in set A. dt_i is the Euclidean distance between non-dominated solution i^{th} in the true-Pareto frontier (y) and the obtained solution (x). $f_k^{\min}$ and $f_k^{\max}$ are minimum and maximum values of k^{th} objective functions in the true Pareto set respectively. k is the number of objective functions. For metric A, lower value indicates superiority of the solution set. When all solutions converge to Pareto-optimal frontier, this metric is zero indicating that the obtained solution set has all solutions in the true Pareto set.

- Spread measurement [2] The second measure is a spread metric. This measure computes the distribution of the obtained Pareto-solutions by calculating a relative distance between consecutive solutions as shown in (7) and (8).

$$\text{Spread(A)} = \frac{sd_f + sd_l + \sum_{i=1}^{|A|-1} \left\| sd_i - \overline{sd} \right\|}{sd_f + sd_i + (|A| - 1)\overline{sd}} \tag{7}$$

$$sd_i = \sqrt{\sum_{k=1}^{k} \left[\frac{f_k(x_i) - f_k(x_{i-1})}{f_k^{\max} - f_k^{\min}} \right]^2} \tag{8}$$

Where sd_f and sd_l are the Euclidean distances between the extreme solutions and boundary solutions of the obtained Pareto-optimal. $|A|$ is the number of elements in the obtained-Pareto solutions. sd_i is the Euclidian distance of between consecutive solutions in the obtained-Pareto solutions for $i = 1,2,\ldots, |A|-1$. $\overline{sd}$ is the average Euclidean distance of sd_i. Operator "‖ ‖" means an absolute value. The value of this measure is zero for a uniform distribution , but it can be more than 1 when bad distribution is found.

4.3 The Results

The comparisons of MOCMA with MOGLS in Table 2.

Table 2 : Comparison Result performance of the solution.

DTLZ	Var.	Performance Measure	MOGLS	MOCMA
1	6	Convergence	0.0151	**0.0045**
		Spread	0.7829	**0.7193**
2	11	Convergence	0.2572	**0.0826**
		Spread	**0.4633**	0.5925

From Table 2.

we can see that MOCMA method was the best performance measured by using Convergence measurement and Spread measurement in experiments. The results are as follows: DTLZ1 = 0.0045 :0.7193 , DTLZ2 = 0.0826:0.5925, DTLZ3 = 0.0175:0.4632 and DTLZ4 = 0.786:0.7125, respectively.

Table 2 (con.): Comparison Result performance of the solution.

DTLZ	Var.	Performance Measure	MOGLS	MOCMA
3	11	Convergence	0.0194	**0.0175**
		Spread	0.5421	**0.4632**
4	11	Convergence	0.2562	**0.0786**
		Spread	0.8354	**0.7125**

5. Conclusions and Future Works

This research investigates the efficiency optimization for Multi-Objective Memetic Algorithms by implementing Chaos Search which is a type of Local Search resulting in a better outcome solution. The technique is tested by solving the DTLZ1-4 multi-Objective test problem and the resulted outcomes are analyzed for outcome capability. The research found that the capability to converge the True Pareto Set of MOCMA is better than that of MOGLS. As per the Spread Measurement, the outcome group resulted from MOCMA is spread more evenly than that of MOGLS. It can be seen that the values of the measurements for both aspects is closer to 0 than those of MOGLS. Further functions are subject to be tested in the next research.

6. References

[1] A. Alkan, E. Özcan, "Memetic algorithms for timetabling" Yeditepe University, 34755 Kayisdagi - Istanbul/Turkey, 2003.

[2] Kumar R, Singh PK , "Pareto evolutionary algorithm hybridized with local search for biobjective TSP" In: Grosan C, Abraham A, Ishibuchi H (eds) Hybrid Evolutionary Algorithms, Springer, Berlin, 2007, pp 361–98.

[3] Nguyen Quoc Viet Hung, Ta Quang Binh and Duong Tuan Anh, "A Memetic Algorithm for Timetabling" Proceedings of 3nd Int,Conf. RIVF'05 Research Informatics Vietnam-Francophony , 2005 , pp. 289 – 294.

[4] H Ishibuchi, Y Hitotsuyanagi, N Tsukamoto, Y., " Implementation of Multiobjective Memetic Algorithms for Combinatorial Optimization Problems: A Knapsack Problem Case Study " Multi-objective Memetic Algorithms, 2009 , pp. 27–49.

[5] Ishibuchi, H., Murata, T., "Multi-Objective Genetic Local Search Algorithm" In: Proc. of 1996 IEEE International Conference on Evolutionary Computation, 1996, pp. 119–124

[6] Jaszkiewicz, A., "Genetic Local Search for Multi-Objective Combinatorial Optimization" European Journal of Operational Research 137, 2002, pp. 50–71

[7] Knowles, J.D., Corne, D.W., "M-PAES: A Memetic Algorithm for Multiobjective Optimization" In: Proc. of 2000 Congress on Evolutionary Computation, 2002, pp. 325–332

[8] Duan, H. and Yu, X., "Hybrid Ant Colony Optimization Using Memetic Algorithm for Traveling Salesman Problem" Proceedings of the IEEE Symposium on Approximate Dynamic Programming and Reinforcement Learning, 2007 , pp. 92-95.

[9] Adriana Lara, Gustavo Sanchez, Carlos A. Coello Coello and Oliver Schütze, "HCS: A New Local Search Strategy for Memetic Multiobjective Evolutionary Algorithms" IEEE Transactions on Evolutionary Computation, Vol.14, No. 1, February 2010

[10] Konak, A., Coit, D.W., Alice E.S., "Multi-objective optimization using genetic algorithms: A tutorial. Reliability Engineering and System Safety" ,2006, pp.992-1007.

[11] Dawkins R. The selfish gene, Oxford : Oxford University Press, 1976.

[12] Merz P, Freisleben B, "A genetic local search approach to the quadratic assignment problem" In : Back CT, editor. Proceedings of the 7th international conference on genetic algorithms, San Diego, CA: Morgan Kaufmann, 1997, pp. 465–72.

[13] Moscato P, Norman MG, "A memetic approach for the traveling salesman problem implementation of a computational ecology for combinatorial optimization on message-passing systems" In: Valero M, Onate E, Jane M, Larriba JL, Suarez B, editors. International conference on parallel computing and transputer application, Amsterdam, Holland: IOS Press, 1992, pp. 177–86.

[14] D. J. Jefferies, J. H. B. Deane and G. G. Johnstone, "An introduction to chaos" Electronics and Communication Engineering Journal, 1989, pp. 115-123.

[15] F. Chen, "Chaos theory and its application" Peking: Chinese Electrical Power Press, 1998.

[16] Shuang Cong, Guodong Li and Xianyong Feng, "An Improved Algorithm of Chaos Optimization", Proceedings of the 8th IEEE International Conference on Control and Automation Xiamen, China, June 9-11, 2010, pp.1196 - 1200

[17] Deb, K., Thiele, L. , Laumanns, M. and Zitzler, E., "Scalable Multi-Objective Optimization Test Problems" CEC 2002 , IEEE Press, 2002, pp. 825 – 830.