Available online www.jocpr.com

Journal of Chemical and Pharmaceutical Research, 2014, 6(7):2547-2553

Research Article

ISSN : 0975-7384
CODEN(USA) : JCPRC5

A study of evolutionary algorithms based on multi-objective pareto optimality

Xue Ding[1]*, Honghong Yang[2], Ren Zhang[3] and Wen Qing Ming[1]

[1]*School of Enrolment and Job-guidance Center, Yunnan Normal University, Kunming, China*
[2]*School of the Arts, Yunnan Normal University, Kunming, China*
[3]*Research Department, Yunnan Normal University, Kunming, China*

ABSTRACT

In real life, there are a lot of Multi-objective Optimization Problem, which is shorted for MOP in the process of people working in production and economic and engineering activities. These problems are often very complex and nonlinear, and even conflicted with each other. When solving these problems, Multi-objective Optimization, shorted for MOO should be done on these issues. For example, for a project, people always want to spend minimum and get maximum efficiency. And here the cost and efficiency are two objectives of this project. In 1896, the French economist Vilfredo Pareto explained the MOP from the perspective of economics, which is now commonly referred as Pareto Optimization. In order to optimize the overall goal, it is necessary to consider the subgoals comprehensively which are conflicted with each other, that is to say that compromise on multiple objectives is needed, so it has multiple solutions. Multi-objective optimization algorithm based on Pareto just uses the algorithm to find the optimal solution to the multiple objectives.

Keywords: Multi-objective, Pareto Optimality, Evolutionary Algorithms

---

INTRODUCTION

The basic concepts of multi-objective optimization

Multi objective optimization is usually called Pareto optimization. Vilfredo Pareto first generalized this concept from perspective of multi-objective optimization. As an economist, he changed the uncomparable multi-objective problem into a single-objective problem in economy. Now it has become a complete theory system [1].

Definition 1 Search Space

Search space is also called decision space, which is a space with all decisive variables. If all decisive variables in search space are real numbers, it can be referred as $x \in Rn$ (n is the number of the decisive variables). The feasible region of search space S is the zone where all constraint conditions are satisfied in search space.

Definition 2 Multi-objective Optimization Problem，MOP:

Multi-objective problem is composed of, n variables, k objective functions, and n constraint conditions. Below is function relationship among objective function, constraint condition and decisive variable.

$$\text{Maximize } F(x) = (f1(x), f2(x), f3(x) \ldots fk(x)) \quad (1)$$

$$\text{Subject to } C(x) = (c1(x), c2(x), c3(x) \ldots ck(x)) \leq 0 \quad (2)$$

In $X = (x1,x2,x3\ldots xn) \in S, Y = (y1, y2, y3\ldots yn) \in O$，x means search vector, and y means objective vector, which is in the data range of constraint condition of $C(x) \leq O$.Therefore, the multi-objective problem can be changed into a mapping process from search vector to objective vector.

Definition 3 Feasible Solution, FS
Feasible solution Xf meets the constraint condition C（x）which is Xf={x∈S | C（x）≤0}. For all Xf in search space, the graph of Xf can be referred as feasible solution region. Through objective function mapping to objective space, the search vectors of this subspace all belong to set of feasible solution.

Definition 4 Domination Criteria, DC
For two vectors in search space x' , x, there are three situations:
x' ＞ x (x' controls strictly x) s.t. F（x'）>F（x）
x' ⩾ x (x' controls x) s.t. F（x'）≥F（x）
x' ↔x (x' and x are incomparable) s.t. F（x'）~ F（x）

Definition 5 Non-dominated Set，NdS
Supposing A∈Xf p（A）={a∈ A| a is non-domination vector in A}, p（A）is set of non-domination search vectors in A. In this set, all vectors are not dominated by other vectors.

Definition 6 Pareto Optimal Front, POF
The corresponding target vector function f (P (A)) is called a non-dominated front-end. For Xf, Xp=p (Xf) is called Pareto optimal set, while Yp=f (Xp) is called Pareto optimal front.

Definition 7 Globle Optimal Solution, GOS and Local Optimal Solution, LOS
For search vector $A \subseteq X_f$,
(1)The set A is called optimal solution for local Pareto, ∀a∈A:∄ x∈Xf: x≻ a⋀||x-a||<ε⋀||f（x）-f（a）||<δ, and the ||.|| is distance, ε >0, δ >0.
(2)The set A is called the optimal global solution, $\forall a \in A: \nexists x \in X \quad f: x \succ a$.

Pareto Optimization
Definition 1 Control:
A decisive vector can control another decisive vector, which is referred as, if and only if: x_1 is no worse than x_2 on all targets. That is to say, $f_k(x_1) \leq f_k(x_2)$, $\forall k = 1,...,n_k$, and at least x_1 is better than x_2 on one target, that is to say, $\exists k = 1,...,n_k$: $f_k(x_1) \prec f_k(x_2)$ .Apparently, one objective f_1 vector controls another one f_2 . If f_1 is no worse than f_2 on all targets, and at least better on one target, then the objective vector control is referred as $f_1 \prec f_2$.

Definition 2 Weak control
A decisive vector weakly dominates another decision vector, which can be referred as $x_1 \leq x_2$: if and only if x_1 is no worse than x_2 on all targets, that is to say, $f_k(x_1) \leq f_k(x_2)\ \forall k = 1,...,n_k$。.

Definition 3: Pareto optimization
One decisive vector $x^* \in F$ is Pareto optimal. If there is no decision variable $x \neq x^* \in F$ control it, that is to say, $\overline{\exists} k : f_k(x) \leq f_k(x^*)$. If x is Pareto optimal, objective vector $f^*(x)$ is Pareto optima.

Definition 4 Pareto optimal set
All Pareto optimal decisive vectors consist Pareto optimal set P^*, that is: $P^* = \{x^* \in F \mid \exists x \in F : x \prec x^*\}$

Definition 5 Pareto optimal front
give an optimal set P^* to target vector $f(\mathrm{x})$ and the Pareto , and the Pareto optimal front $pF^* \subseteq O$ is defined as: $pF^* = \{\mathrm{f} = (\mathrm{f}_1(\mathrm{x}^*), \mathrm{f}_2(\mathrm{x}^*),...,\mathrm{f}_k(\mathrm{x}^*)) \mid \mathrm{x}^* \in \mathrm{P}\}$.The aim to solve the multi-objective problem is to estimate the real Pareto optimal front, then choose the best trade-off result. But is is usually infeasible to find an exact Pareto optimal front computationally. Therefore, the actual application is to find an estimate about Pareto front-end which must meet the condition that, the distance to Pareto front-end must be minimum distance, and the more

dispersed the solutions in Pareto optimal set are, the better; and at last keep the non-dominated solution. Objectives: the first goal is to make sure the estimation is as accurate as possible, the second goal is to ensure that the entire Pareto is covered.

EXPERIMENTAL SECTION

Division of multi-objective evolutionary algorithm

Since Schaffer first proposed multi-objective evolutionary algorithm [1] in 1985, there has come a large number of research results about multi-objective evolutionary algorithm (MOEAs :Multi-Objective Evolutionary Algorithms). VanVeldhuizen[2], divided MOEAs into 3 categories:

A priori decision method That decision makers will combine multi-objective into one objective function first, and then to search the optimal solution by algorithm. The advantage of this method is convenient and quick, and can be solved by the traditional genetic algorithm, with a complete theoretical system. The disadvantage of this method is that a lot of optimal solutions may be lost.

Adaptative method. Decision makers and the algorithm are interactive. Decision makers provide the primary and secondary relation of the objectives. The algorithm provides better target priority for decision makers. This method is a better method, and the disadvantage is that the design process is relatively complex.

A posteriori decision method. Through calculating the algorithm can get a better solution set; and provide a set of candidate solutions for decision makers to choose according to the preference information of decision makers. The advantage of this method is that the algorithm is relatively simple, and the information provided is more flexible.

According to the characteristics of the above methods above to divide multi-objective evolutionary algorithm, most multi-objective evolutionary algorithms are used a posteriori decision method to solve the multi-objective optimization problem.

Literature review of multi-objective evolutionary algorithm

At present, study on multi-objective evolutionary algorithm mainly focus on experiment, and researchers created a variety of multi-objective evolutionary algorithm based on different genetic strategies[3].

Schaffer was first invented VEGA (Vector Evaluated Genetic Algorithm, VEGA) [4], after this, the researchers used many simple methods to solve the multi-objective problem. The most commonly used method is the method of linear polymerization equation[5]. In this method, multiple objective functions are changed into one target and take it as the fitness. Then the evolutionary algorithm is used to get the optimal solution. Nonlinear polymerization method at this time is also very popular[6]. Dictionary order method is often used more. That is to say, first select a target (considered one of the most important) as the optimization object, and does not consider other objectives. Then, the second objectives are optimized, and the optimized result will not reduce the quality of the first results. This process will continue until all goals are calculated over [7].

David E Gokdberg[8] first used the Pareto genetic algorithm theory to summarize these algorithms. When he quoted Schaffer's VEGA, Goldberg recommended to use non-dominated ranking and selection to keep the direction of evolutionary population toward the Pareto front-end. The basic idea is: find a set of solutions in the population, and the solution set for the other individual population is non-dominated. Notably, these solutions can be assigned as a higher level or to be eliminated in the competition. Then, the Pareto non-dominated set is assigned to a collection of second high. Before finding the right individual, the process will continued. Goldberg also recommended the GA(genetic algorithm) take a certain mechanism to prevent the whole algorithm set finally gathered to a certain point in front of the Pareto solution. Although there is no specific feasible way to give from Goldberg, a lot of multi-objective evolutionary algorithms use this thought in his book.

From 1989 to 1998, there are a large number of multi-objective evolutionary algorithm with the efficiency as characteristics during this period. The representatives are: Non-dominated Sorting Genetic Algorithm, NSGA [9], Niched-Pareto Genetic Algorithm, NPGA[10], Multi-Objective Algorithm,MOGA[11].

Few people compare the advantages and disadvantages between the multi-objective evolutionary algorithms. The algorithm mainly emphasizes simplicity, and lack the correct method to test them. Because there is no test function, people usually validate them just by observation. Masahiro Tanaka[12] has done important work, and he first put the user's preference information in the multi-objective evolutionary algorithm. Because the people in the real world often don't need the whole Pareto set but only need a very small component (probably only one solution). Therefore, the users can reduce the search range according to some preference information and can enlarge the certain part of the Pareto front-end. Fonseca and Fleming also presented a method to measure the performance of the algorithmin evolutionary on computation magazine[13]. This method does not depend on the Pareto front actual problem. In short, the early algorithm can select the non-dominated individuals (possible, but is not required) and keep the diversity of population in a certain extent.

Since then, scholars have done a lot of research on the multi-objective evolutionary algorithm performance. Multi-objective evolutionary algorithms which emphasize the performance began to appear. These algorithms appeared with the elitist strategy as a symbol. Although many early researchers have considered the concept of the elite strategy, the first person who introduced it to multi-objective evolutionary algorithms is Eckart Zitzler. He invented Strength Pareto Evolutionary Algrotithm, SPEA[14]. This is a milestone of multi-objective evolutionary algorithms. Since SPEA was proposed, most researchers began to study the external population (save the elite population). In the multi-objective optimization, elitist strategy usually involves the external population (or second population). In this population, non-dominated individuals can be preserved with the process of evolution. The basis of this idea mainly lies: the current population of non-dominated individuals with evolution, may no longer be the non-dominated individuals. Therefore, the simplest method is to preserve the non-dominated solutions in the external preservation. If this individual dominates other individuals, and isn't controlled by the external population, the individual will be a part of the external population as elitist.

The the typical algorithms with elitist strategy are: Strength Pareto Evolutionary Algorithm, SPEA[15], Pareto Archived, Evolution Strategy, PAES[16], Non-dominated Sorting Genetic Algorithm II, [17] NSGAII. Zheng Jinhua and Shi Zhongzhi also proposed a DMOEA[17] .
Among these objective evolutionary algorithms, the fitness is no longer the only method to change population diversity. Researchers not only consider the algorithm level, but also the data structure level.

Initialization of chaotic sequence based on Pareto Optimality
In order to overcome the non-uniformrandom generated from the traditional method of initialization of population, real coded chaotic initialization based on Pareto Optimality is used in this paper. Zhang Xi and so on analized real number coding method [17]and summarized the characteristics. In general, first, by using real number as the initial population, the encoding process is simple. Secondly, real coding can eliminate the cliff problem in the process of "Hamming" of common binary code. Finally, real coding is easy to control, especially for chaos initialization and chaotic mutation.Population initialization is based on the scope of the problem and constraints. The algorithm proposed in this paper is based on the triangular tent map:

$$p(k+1)=\begin{cases} 2p(k), 0 \le p(k) \le 1/2 \\ 2-2p(k), 1/2 \le p(k) \le 1 \end{cases} \quad (1)$$

mapping directly to [0,1], and consists Pareto Optimality chaos initialization.

Pareto Optimality chaotic mutation
The variation based on Pareto Optimality multi-objective evolutionary algorithm is achieved by Tent chaotic mapping.

Set the mutation step size as δ , δ also has Tent chaotic mapping.

$$\delta(k+1)=\begin{cases} 2\delta(k), 0 \le \delta(k) \le 1/2 \\ 2-2\delta(k), 1/2 \le \delta(k) \le 1 \end{cases} \quad (2)$$

Chaotic mutation:

$$c_k = p_k + (p_k^u - p_k^l)\delta_k \quad (3)$$

Here, c_k stands for the offspring and p_k stands for parent, and p_k^u is the upper bound of the parent component. p_k^l is the lower bound. δ_k is a small step on chaotic sequences.

RESULTS AND DISCUSSION

Improved computation of crowding distance

In the multi-objective evolutionary algorithm, the objective function is optimized with a plurality of. It is an urgent problem to be solved how to use the objective function score of the individual, . In this paper, crowing distance is used to choose the individual. Suppose a multi-objective optimization problem with two objectives and, crowding distance of one individual is quadrilateral long distance and wide range, the crowding distance with representatives of individuals here, $P[i]_{distance}$ means the crowding distance, $P[i].m$ means the target value of individual:

$$P[i]_{distance} = (P[i+1].f_1 - P[i-1].f_1) + (P[i+1].f_2 - P[i-1].f_2) \quad (4)$$

Like the original thoughts, first calculate the crowding distance of the front of each one individual and then, calculate the crowding distance in front of the individual.

$$\bar{d} = \sum_{i=1}^{popsize} P[i]_{distance} / popsize \quad (5)$$

The distance can be defined as:

$$d_i^{'} = P[i]_{distance} / \bar{d} \quad (6)$$

Dynamic mutation probability

The mutation operator is the most important part of Pareto Optimality evolutionary algorithm. The good mutation operator can directly affect the performance of the algorithm. Therefore, the introduction of dynamic variation can better regulate the mutation (fine turning) role. This mutation was proposed by Srinvivas. In his method, *P* can change with the fitness of the change. The algorithm follows [15]:

$$p = \begin{cases} k(f_{\max} - f)/(f_{\max} - f_{avg}), f \geq f_{avg} \\ k', f < f_{avg} \end{cases} \quad (7)$$

Here, $f_{\max}$ is the maximum fitness value of the population, f is the individual fitness value, f_{avg} is the average fitness value. The mutation probability is adjusted by k, k'.

This method is a good method of mutation, but due to in the multi-objective evolutionary algorithm of Pareto Optimality, comparison between individuals is not in accordance with the fitness function size to compare but according to dominant fashion, so this method can't be used well in the multi-objective evolutionary algorithm.

This is a dynamic mutation probability based on crowding distance, described as follows:

$$p = \begin{cases} k(d - d_{\min})/(d_{avg} - d_{\min}), d \leq d_{avg} \\ k', d > d_{avg} \end{cases} \quad (8)$$

Among them, d is the crowding distance needs of individual variation, d_{min} is the minimum crowding distance of population, and d_{avg} is the average crowding distance of population. This expression means when the crowding distance is less than the average crowding distance, variate according to the dynamic mutation probability. When the crowding distance is greater than the mean crowding distance, in order to prevent the discrete individual excessive, mutation probability k' should be set appropriately

Mutation probability based on iterative times

Hashem also presented an algorithm which can vary dynamic mutation with the running time [16]. Methods of mutation probability based on iterative times are also proposed in this pape.

Fitness scaling should solve two problems:

Probability of using chaotic mutation in the early process is very high. Using the stochastic property and ergodicity of chaos mutation and searching in the decision space in a wide range of search can avoid local optimal solution. This can better maintain the diversity of population.

When the evolutionary algorithm process reaches a certain stage and the algorithm converge to the global optimal solution, with the reason that the individual fitness values are very close, in order to prevent the individual swings in larger range , probability of mutation should be reduced and rely on binary crossover slowly close to optimal Pareto surface.

The mutation algorithm is proposed in this paper:

$$p = (n - i)/n \times k \tag{9}$$

Here,*n* means the iterative times of the multi-objective evolutionary algorithm, *i* means the current iteration, *k* means the mutation probability parameters.

Evolutionary algorithm of optimal problem based on multi-objective Pareto

The method mentioned in this paper mainly used to solve the multi-objective optimization problems. With chaos initialization, dynamic chaotic mutation, as well as new crowding distance computation, the specific algorithm is described as follows:

Input: Num (size of the initial population)
F (objective function)
T (the largest genetic algebra)
pF^* (Pareto optimal front-end)
Output: Nd_Set (non-dominated set)
P^* (Pareto optimal set)

The first step, create the initial population: generate P_0 by chaos mutation, t = 0.

The second step, cross selection: use tournament algorithm into the mating pool, its size is equivalent to Num.

The third step, gene operation: cross and dynamic chaos variation in the mating pool, and the new individual enters the Que_t.

The fourth step, calculating the degree of adaptation by crowding distance: the calculation individual crowding distance of Por_t and Que_ttwo.

The fifth step, select individual based on the objective function: copy Por_t and Que_t to the non-dominant set Nd_Set. If the number of individuals in the Nd_Set is greater than Num, use crowding distance to sort. And the crowding distance low individual access to Por_t first until the scale is equal to Num; if its size is greater than or equal to Num, copy Nds crowding distance larger individuals into the Por_t+1, and so on. What needs to pay attention to is, the part whose size is less than Num, select the dominance high part into Por_t.

The sixth step, end: if t <Times, t = t+1, for the second step, and output the individuals in the non-dominated set P_{t+1}, get P^*

Summary

This paper first analyzes the significance of optimal multi-objective evolutionary algorithms based on Pareto. The Pareto optimization method can select local optimum quickly, and have a good ability of global optimization, and introduce chaotic mutation and Pareto optimization into multi-objective evolutionary algorithm.

Then the thesis introduces Pareto optimal and chaos theory and optimization method, and explains some characteristics of Tent sequences. Through some analysis and argumentation, compared with other chaotic sequences, the Tent sequence has better uniformity and ergodic which is better to search as the mutation operator of multi-objective evolutionary algorithm and chaotic search.

At the end of this paper, multi objective evolutionary algorithms based on Pareto optimal is designed and carried out.

And then analysis is carried out to the practicality of the algorithm.

REFERENCES

[1] Vilfredo Pareto.Cours D Economie Politique,volume I and II[M]. F.Rouge ,Lausanne,**1896**.
[2]Schaffer J.D.,Multi objective optimization with vector evaluated genetic algorithms[C].1st International Conference on Genetic Algorithms,Lawrence Erlbaum,**1985**.
[3]Xie Tao, Chen Huowang, Knag Lishan. The evolution algorithm of the multi-objective optimization. *Computer Journal*.**2003**,26（8）: 997-1003.
[4]Back T.Evolutionary Algorithms in theory and practice.New york:Oxford University Press,**1996**.
[5]Neil H.Eklund and Mark J.Embrechts.GA-Based Multi-Objective Optimization of Visi-ble Spectra for Lamp Design[J].Neural Networks,Fuzzy Logic,Evolutionary Programming,Data Mining and Complex Systems,**1999**.
[6]Carlos A.Coello Coello and Alan D.Christiansen.Two New GA-based methods for multi-objective optimization.*Civil Engineering Systems*, 15（3）:207–243,**1998**.
[7]Louis Gacogne.Research of Pareto Set by Genetic Algorithm , Application to Multicriteria Optimization of Fuzzy Controller[C].The 5th European Congress on Intelligent Techniques and Soft Computing EUFIT'97,pages 837–845,**1997**.
[8]E.Goldberg.Genetic Algorithms in Search,Optimization and Machine Learning[M].Addison-Wesley Publishing Company,Reading, Massachusetts, **1989**.
[9]N.Srinivas and Kalyanmoy Deb.Multiobjective Optimization Using Nondominated Sort-ing in Genetic Algorithms[J].*Evolutionary Computation*,2（3）:221–248,Fall **1994**.
[10]Jeffrey Horn,Nicholas Nafpliotis,and David E.Goldberg.A Niched Pareto Genetic Algorithm for Multiobjective Optimization[C].The First IEEE Conference on Evolutionary Computation,*IEEE World Congress on Computational Intelligence*,volume 1,pages 82–87,**1994**.
[11]Carlos M.Fonseca and Peter J.Fleming.Genetic Algorithms for Multiobjective Optimization:Formulation,Discussion and Generalization[C].The Fifth International Conference on Genetic Algorithms,pages 416–423,**1993**.
[12]Masahiro Tanaka and Tetsuzo Tanino.Global optimization by the genetic algorithm in a multiobjective decision support system[C].The 10th International Conference on Multiple Criteria Decision Making,**1992**.
[13]Carlos M.Fonseca and Peter J.Fleming.An Overview of Evolutionary Algorithms in Multiobjective Optimization[J].*Evolutionary Computation*,3（1）:1–16,Spring **1995**.
[14]Eckart Zitzler and Lothar Thiele.Multiobjective Evolutionary Algorithms:A Comparative Case Study and the Strength Pareto Approach[J].*IEEE Transactions on Evolutionary Computation*,3（4）:257–271,**1999**.
[15]Joshua D.Knowles and David W.Corne.Approximating the Nondominated Front Using the Pareto Archived Evolution Strategy[J].*Evolutionary Computation*,8（2）:149–172,**2000**.
[16]K.Deb,A.Pratab,S.Agarwal,and T.Meyarivan,A fast and elitist multiobjective genetic algorithm:NSGAII[J],*IEEE Trans.Evol.Compute.* ,vol.6,Apr.**2002**.,pages 182-197.
[17]Zheng Jinhua, Shi Zhongzhi.The rapid multi-objective genetic algorithm based on clustering[J]. *Computer Study and Development,***2004**,41（47）： 1081～1087.2.