

Multi-objective Particle Swarm Optimization Algorithm for Recommender System

^{1,2}Zhaoxing Li, ²Chen Li, ³Zhang Jue

¹School of Information and Science Technology, Northwest University

Xi'an, 710127, China (abde0309@126.com)

²School of Information Technology, Northwest University

Xi'an, 710127, China (chenli@nwu.edu.cn)

³School of Information and Science Technology, Northwest University

Xi'an, 710127, China (2105153266@qq.com)

Abstract

This paper models the process of a recommender system as a multiobjective optimization problem, a discrete particle swarm optimization framework is established and has been integrated into multiobjective optimization, consequently, a multiobjective discrete particle swarm optimization algorithm is proposed to solve the modeled optimization problem. Each run of the current mainstream recommender algorithms can only give one recommendation scheme, while the proposed algorithm can generate a set of schemes, and it provides the decision maker with more choices, a decision maker can make use of users' registration information to choose a personalized scheme to recommend to the user. In order to check the performance of the proposed algorithm, experiments on simulated and real-world data are tested and a comparison with the classical user based collaborative filtering recommendation method is made, experiments demonstrate the effectiveness of the proposed algorithm.

Key words: Recommender system, Collaborative filtering recommendation, Multi-objective optimization, Particle swarm optimization

1. Introduction

In the 21st century, an age of information, great changes have taken place in human's traditional way of communicating, living and producing via the striking advancements of computer technology and network technology. Before, the computer and the network had not developed enough, and people had narrow channel to gain information which was not rich, either. However, with the prevailing of mobile devices and the wild spread of network, information is more available to people, which aroused a problem, namely, information overload. Currently, a new term, "Big Data", has been put forward in academia. And how to deal with such "big data" has already become another need-to-be-solved scientific problem in the interdisciplinary circles.

Big data meets the demand of information to the people, but under its circumstance, it is difficult for people to gain information that they are interested. In order to provide users with better personalized services, recommendation system came into being [1-3]. It's a system that recommends users information or products' details they might be interested. The system has become commonplace in life, news portals such as Sina and Yahoo; online shopping sites such as Jingdong and Amazon; dating sites such as Baihe and Jiayuan etc. All of them are recommendation systems, which play an important role in modern production and life. The research on recommendation system has gradually formed a relatively independent subject.

The purpose of Recommendation System is to produce recommendation to the users, while the recommendation algorithm directly affects the recommendation quality. At present, the mainstream recommendation algorithm is Collaborative Filtering (CF) recommendation algorithm [4-6]. The core of CF algorithm is that the favorite products the users who have the similar interests with the target users like are that of target users like, and the similar items with that of target users like are the ones they like. As a recommendation system designer, the hope is that the recommended items are that of the users like or they are interested in. While as a good manufacturer, they not only want to be able to recommend hot items to the users, also hope to be able to recommend users of non-hot items as much as possible.

Based on the above two considerations, a multi-objective recommendation model is presented in this paper, with which one is to optimize the accuracy of recommendation, the other is to optimize the novelty of recommendation. In this paper, a discrete particle swarm optimization framework is proposed and combines with multi-objective recommendation model as well as the discrete multi-objective particle swarm optimization algorithm is proposed to recommending models in modelling. Classical algorithm CF provides only one solution at a time, while the algorithm proposed in this paper will provide a set of solutions, each solution represents

a recommended proposal, which means this algorithm is able to provide a series of alternatives for decision makers at a time. Decision makers can choose one suitable proposal based on the user's registration information for the user or select a recommendation randomly. The recommended algorithm in this paper can not only make up the shortcomings of traditional algorithm which only chase a single goal, but also more in line with the practical application. To test the performance of recommended algorithm, an algorithm test based on the simulated data and real data as well as a comparison with user-based collaborative filtering algorithm (UBCF algorithm) are carried out in this paper, which both prove the high efficiency of the algorithm presented in this paper.

2. Recommendation System And Evolutionary Optimization

Recommendation system plays a very important role in all aspects of our daily life. A general recommendation system has three important modules: user recommender module, recommended object model, recommendation algorithm. Fig. 1 shows a general model of the recommendation systems.

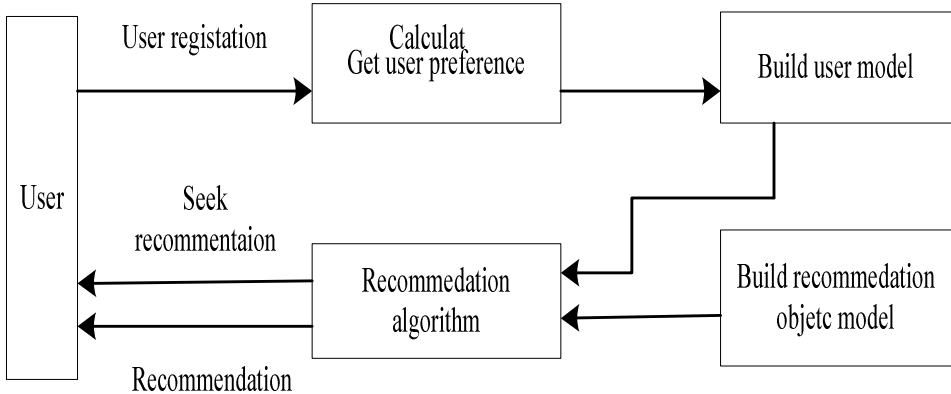


Fig.1. Recommendation System Model

The recommendation algorithm in figure 1 is the core of the recommendation system, the mainstream recommendation system algorithm is mainly divided into three categories: the recommendation algorithm based on content, recommendation algorithm based on the CF and hybrid recommendation algorithm. Among them, the CF algorithm is one of the more widely used one. The most common data expression of recommendation system is users-item rating matrix R, wherein, r_{ij} represents the likeness degree of user i to user j, and the likeness degree use 1-5 criteria, namely r_{ij} is a number between 1 to 5.

2.1 Sub-Section A -Subtitle

PSO is a stochastic searching algorithm. PSO is an artificial intelligent optimization method and it was originated from the social behaviors such as fish schooling and bird flocking. It was proposed by Eberhart and Kennedy in 1995. Ever since its advent, PSO has now become a popular optimization technique.

PSO optimizes[7,8] a problem by having a swarm of individuals. Each individual is called a particle. A particle can learn from its neighbors so as to adjust its velocity. In PSO, the position vector of a particle simulates a candidate solution to the optimization problem. After getting a new velocity, a particle will update its position vector, in other words, a new solution is generated. Given that the size of the particle swarm is pop and the dimension of the search space is n .

Let $V_i = \{v_i^1, v_i^2, \dots, v_i^n\}$ and $X_i = \{x_i^1, x_i^2, \dots, x_i^n\}$ be the i -th $i = (1, 2, \dots, pop)$ particle' velocity and position vectors, respectively, then the basic form of the rules for a particle i to adjust its status read:

$$V_i = V_i + c_1 r_1 (P_i - X_i) + c_2 r_2 (G - X_i) \quad (1)$$

$$X_i = X_i + V_i \quad (2)$$

where $P_i = \{p_i^1, p_i^2, \dots, p_i^n\}$ is the i -th particle's personal best position and $G = \{g_1, g_2, \dots, g_n\}$ is the best position of the swarm. Parameter c_1 and c_2 are the learning factors, and r_1 and r_2 are two random numbers between 0 and 1.

PSO has been proved to be very promising for solving many optimization problems. A lot of creative work has been done to improve the performance of the algorithm [13, 14, 15, 16]. PSO has less parameters compared with other evolutionary algorithms. The framework of PSO is concise and it is easy to realize. However, PSO is originally designed for solving continuous optimization problems. Driven by real applications, much effort has been devoted to extending continuous PSO to discrete ones.

2.2 Multi-objective Optimization

Objective optimization problem was originated in the design of many complex systems, modelling, planning issues. Since the 1960s, multi-objective optimization problem attracted the attention of a growing number of researchers from different backgrounds. Especially in recent years, multi-objective evolutionary algorithm is to obtain the optimization of the more widely used and studied, resulting in a series of novel algorithms and get a good application. Multi-objective optimization proposition is generally no unique global optimal solution, so this is actually a multi-objective optimization proposition is often how the process of seeking Pareto set.

Traditional multi-objective algorithm is often converted into a single objective proposition after the use of sophisticated single-objective optimization algorithm, the drawback is that the optimal solution can only be determined once a solution. And now the multi-objective evolutionary strategy tends to be more parallel computing can be solved once a sufficient number of solutions distributed on the Pareto front provides decision-makers to the next decision. Which PSO as a novel evolutionary computing strategy has been more and more widely used in multi-objective optimization problem. Multi-objective optimization is described as follows:

Definition 1:

$$\min f(x) = [f_1(x), f_2(x) \dots f_n(x)]$$

$$st. g_i(x) \leq 0, i = 1, 2 \dots m$$

$$h_j(x) = 0 \quad j = 1, 2, \dots, n$$

where, $x = [x_1, x_2, \dots, x_D] \in R^D$

$f_1(x)(i = 1, 2, \dots, n)$ is the objective function;

$f_1(x)(i = 1, 2, \dots, n)$ is the inequality constraint;

$f_1(x)(i = 1, 2, \dots, n)$ is the equality constraints.

Definition 2: global optimum

Given a multi-objective optimization of the overall proposition $f(x)$, $f(x)$ is called optimal if and only if $\forall x \in \Omega$, have $f(\bar{x}) \leq f(x)$.

Multi-objective optimization goal is to get a balance between each target, so will eventually get a good solution between a set of multi-objective optimization.

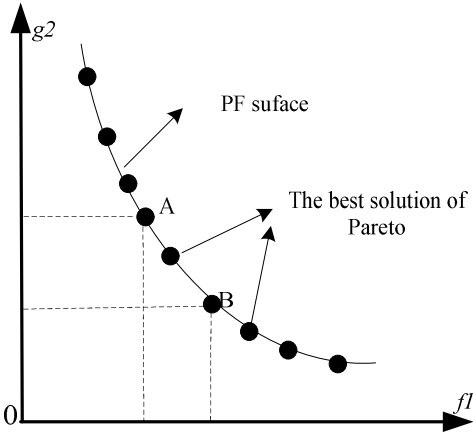


Fig.2. Multi-Objective Optimization Pareto Solution.

3. Particle Status Update Rule

3.1 The Representation and Status Updates of Particles

For the proposed algorithm, the particle status update rules are in the similar form with that in Gong et al. [12]. The rules read as follow

$$V_i = \phi(\omega V_i + c_1 r_1 (Pbest_i \oplus X_i) + c_2 r_2 (Gbest \oplus X_i)). \quad (3)$$

$$X_i = X_i \Theta V_i \quad (4)$$

In the above equations, ω is the inertia weight, c_1 and c_2 are the learning factors, and r_1 and r_2 are two random numbers within the range $[0,1]$. The symbol \oplus is the XOR operator. The function $\phi(t)$ is defined as follows:

$$\phi(t) = \begin{cases} 1 & \text{if } rand(0,1) < 1 / (1 + e^{-t}) \\ 0 & \text{if } rand(0,1) \geq 1 / (1 + e^{-t}) \end{cases} \quad (5)$$

It can be seen from Eq. 3 that the obtained particle velocity vector is a binary sequence. Therefore, the key component of the algorithm lies in the design of Eq. 4. The authors have defined the symbol Θ in Eq. 4 in the following way:

$$(6) \quad \begin{cases} X_i \Theta V_i = X_i' = \{x_{i1}', x_{i2}', \dots, x_{in}'\} \\ x_{ij}' = x_{ij} \text{ if } v_i = 0 \\ x_{ij}' = \arg \max_r \sum_{k \in N_j} \delta(x_{ik}, r) \text{ if } v_i = 1 \end{cases}$$

where N_j is the neighbor set of node j . $\delta(a, b) = 1$ if $a = b$, otherwise, 0.

The authors argue that the above defined particle position update rule makes sense, because the function $\arg \max_r f(x)$ returns the value r that maximizes $f(x)$. In other words, it assigns a node to the community that the majority of its neighbor nodes belong to. However, through our analysis we have found that this kind of update strategy may easily lead to a large connection of a community and it will eventually lead to small modularity values.

In this paper, we have redefined the position update rule in Eq. 6. The redefined equation reads:

$$x_{ij}' = \arg \max_{x_{ik}} \Delta Q(x_{ij} \leftarrow x_{ik} \mid k \in N_j) \text{ if } v_i = 1 \quad (7)$$

The above equation returns the value x_{ik} that can generate largest modularity increment when the current element x_{ij} of the position vector X_i is replaced by this vary value. To define the

position update rule in this form can gain large objective function value since each step of the updating always find the community identifier that can lead to the increment of the modularity. It should be pointed out that when there are multiple x_{ik} that can generate largest modularity increment, then we randomly choose one from them.

3.2 Particle Fitness Funtion

The fitness function is to evaluate the merits of the standard particle state. A good recommendation system not only try to give users recommend users favorite items , that is, to satisfied the high accuracy and recommended recall rate , but also to try to give users recommend users may prefer new items , that is, to meet the high degree of novelty . In view of this, the recommended model for multi-objective design follows this article:

$$\max \begin{cases} f_1 = |R \cap L|/|L| \\ f_2 = |R - R \cap L|/|R| \end{cases} \quad (8)$$

In the model, R represents the recommendation list, L represents the user preference list and $|\bullet|$ represents the number of elements. In the first function, f1 measures the recall rate of the recommendations; and if there are more elements in the intersection of R and L, there are more goods among recommendations that a user likes. In the second function, f2 measures the precisions of the recommendations; and if there are fewer elements in the intersection of R and L is, there are more goods among recommendations that are novel to a user. The increase of f1 will lead to the decrease of f2. Obviously, f1 and f2 satisfy the conditions of multi-objective optimization. The algorithm of this paper is aimed to design a recommendation system that can balance between the recall rate and the novelty.

3.3 Global Optimum Individual Selection

In order to make the optimization framework of discrete particle swarm apply to solve the recommendation model, a discrete recommend multi-objective particle swarm optimization algorithm was put forward by this paper. Evolution Multi-Objective Model in this paper was based on the non- dominated sorting genetic algorithm NSGAI [9]. The biggest difference between Multi-objective particle swarm optimization and single target particle swarm optimization lied in the selection of whole optimal solution, because the optimal solution in the single objective optimization had only one, while in the multi-objective optimization ,there were multiple optimal solution, how to select one from these solutions to guide the status updates of the particles was crucial, and several methods were put forward by many scholars according to the problems[10-11].

During this algorithm, random option principle was adopted by this paper, when particles updated every time, randomly choosing one from Pareto optional solution collection to be the whole optimal solution to conduct particles update process

4. Experiment Studies

4.1 Experiment Settings

In order to test the recommendation effectiveness of the algorithm in this paper, simulate data test and real data test of the algorithm will be used in this section, moreover, experimental comparison between the collaborative filtering algorithm based on users and Recommendation System algorithm. The programming language is Matlab, and programming environment is Matlab 2010, the experimental platform is a normal computer, the configuration is the Intel (R) Core (TM) i3 CPU, 3.19 GHz, 2GB memory, hard disk 500 G, the operating system is Windows 7 ultimate version. In this paper, the size of particle swarm algorithm is set to 100, external population size is set to 500, number iteration algorithm is set to 100.

Simulation data used in the experiment describes the judgment of 8 users to 8 movies, users-scoring matrix is shown in Table 1.

The real data used in the experiment is from the real film data aggregation called Minnesota collected by project team, GroupLens Research, Minnesota University. The online download link is <http://www.grouplens.org/>. The Movielens data adopted in this article are the grades made by 943 customers on 1682 movies. Including the lowest grade 20, the highest one 752, the total grade adds up to 100000.

Table 1. Recommended System Simulation Data

	M1	M2	M3	M4	M5	M6	M7	M8
U1	3.5	2	4.5	5	1.5	2.5	2	—
U2	2	3.5	—	2	3.5	—	3	4
U3	5	1	3	5	1	—	—	1
U4	3	4	—	3	4.5	4	2	4.5
U5	4	—	4	—	—	4	1	1
U6	—	4.5	5	5	4.5	4	4	4
U7	5	2	3	5	4	5	—	—
U8	3	—	5	4	2.5	3	—	—

4.1 Simulation Experiment Data

Figure 3 showed the PF side which was calculated from the multi-objective Particle Swarm Optimization Algorithm Recommendation System which has been proposed before in this text to do recommendation to 8 users. It can be found from the text through Figure 3 that Pareto distribution on PF side is smoother and more evenly. Every Pareto distribution represents a recommendation proposal. The arithmetic mentioned in this text could provide with a group of recommendation proposals for the decision-makers to make decision.

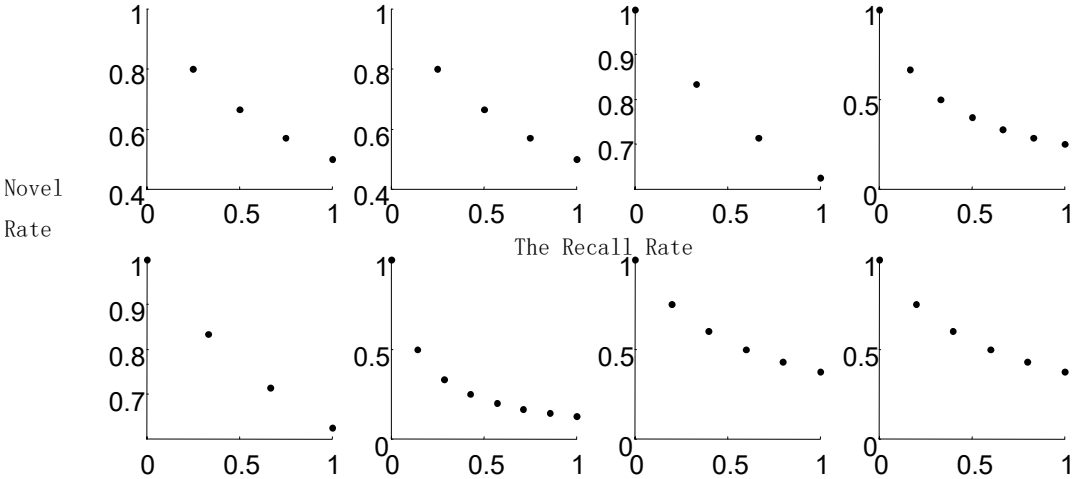


Fig.3. The Description of The State of PF Surface When Recommending to 8 Users

Due to the pursuit of accuracy of recommended system generally, in order to facilitate the comparison with the classic collaborative filtering UBCF algorithm, in this thesis, the most optimal solution based on accuracy on PF face is selected as the final results of the algorithm in this thesis.

Compared with UBCF, this algorithm involves several parameters, number of neighbors, the similarity measurement, in the simulation data experiments, these two parameters were set to 4 and cosine. Table 2 shows that the experimental comparison between algorithm in this thesis and UBCF algorithm under the recommended precision and recall rate.

Table 2 The Result Comparison between Recommendation System Algorithm Based on Multi-objective Particle Swarm Optimization and UBCF

		U1	U2	U3	U4	U5	U6	U7	U8
Precision	Algorithm	50%	50%	62.5%	75%	62.5%	12.5%	37.5%	37.5%
	UBCF	50%	50%	37.5%	75%	37.5%	87.5%	62.5%	62.5%
Recall rate	Algorithm	100%	100%	100%	100%	100%	100%	100%	100%
	UBCF	80%	66.67%	60%	85.71%	50%	87.5%	71.43%	62.5%

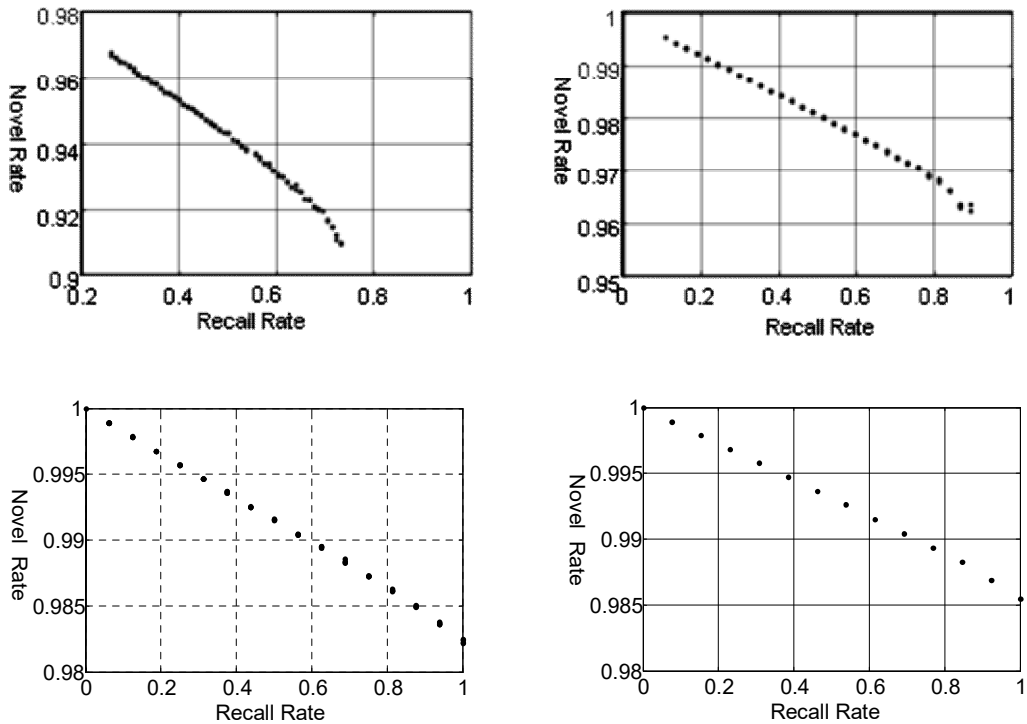


Fig.4. The Description of The State of PF Surface When Recommending to Users

Due to the pursuit of accuracy of recommended system generally, in order to facilitate the comparison with the classic collaborative filtering UBCF algorithm, in this thesis, the most optimal solution based on accuracy on PF face is selected as the final results of the algorithm in this thesis.

Compared with UBCF, this algorithm involves several parameters, number of neighbors, the similarity measurement, in the simulation data experiments, these two parameters were set to 4 and cosine. Table 2 shows that the experimental comparison between algorithm in this thesis and UBCF algorithm under the recommended precision and recall rate.

As can be seen from Table 2, when the first five users are tested, whether seeing from recommended precision or recall rate, the algorithm in this thesis is superior to the classical UBCF algorithms, when the algorithm is applied to the last 3 users, the recommended accuracy of the algorithm in this thesis declined slightly.

Generally speaking, algorithm in this article keep the higher recommendation accuracy and recall rate, meanwhile, also keep the higher novelty. And the higher novelty has more advantage for the promotion of recommend system.

4.3 The Real Experiment Data

In the above section, we had the algorithm experiments based on the simulation data. This section we will carry on the text experiment to the actual recommendation system data of algorithm. Due to large amounts of the actual data and without loss of generality, this article selected the four leading users for texting. Figure 5 gives a description of the state of PF surface when recommending to users. The experiment results demonstrate that the arithmetic used in this article can obtain a relatively high degree of recommendation accuracy while ensuring a relatively high degree of recommendation novelty. The algorithm of this article could generate more recommendation plans when recommending to users who own more historical information, which is reflected in the FIG.4, 3 and 4 are redundant to the solutions of Pareto in the PF 1 and 2.

5. Conclusion

Recommender system provide a good solution to information overload, the related study of recommender has received attention from all fields of society. Here the article raised a kind of multi-object recommended model, designed a recommended model which is discrete multi-objective particle swarm optimization algorithm. The proposed recommended validity of the algorithm has been verified on both analog data and real data. In the classic contrast experiment of collaborative filtering recommendation algorithm based on users, the recommendation algorithm the article raised can provide variety of recommended proposal to the decision maker. Future work will continue to further study recommendation system algorithm. Due to the recommendation system background database is so large, it is also needed to further study the recommendation algorithm which is faster to processing a large amount of data.

Acknowledgment

This work is supported by the the Science Project of Yulin University (Grant Nos. 16GK-25). The National Nature Science Foundation of China (No.11641002)

References

1. Bobadilla J, Ortega F, Hernando A, et al. Recommender systems survey[J]. *Knowledge-Based Systems*, 2013, 46: 109-132.
2. Adomavicius G, Tuzhilin A. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions[J]. *IEEE Transactions on Knowledge and Data Engineering*, 2005, 17(6): 734-749.

3. Lü L, Medo M, Yeung C H, et al. Recommender systems[J]. *Physics Reports*, 2012, 519(1): 1-49.
4. Schafer J B, Frankowski D, Herlocker J, et al. Collaborative filtering recommender systems[M]. *The adaptive web*. Springer Berlin Heidelberg, 2007: 291-324.
5. Su X, Khoshgoftaar T M. A survey of collaborative filtering techniques[J]. *Advances in artificial intelligence*, 2009, 2009: 4.
6. Hofmann T. Latent semantic models for collaborative filtering[J]. *ACM Transactions on Information Systems (TOIS)*, 2004, 22(1): 89-115.
7. Kennedy J, Eberhart R. Particle swarm optimization[C]//*Proceedings of IEEE international conference on neural networks*, 1995, 4(2): 1942-1948.
8. Clerc M. Particle swarm optimization[M]. *John Wiley & Sons*, 2010.
9. Deb K, Pratap A, Agarwal S, et al. A fast and elitist multiobjective genetic algorithm: NSGA-II[J]. *IEEE Transactions on Evolutionary Computation*, 2002, 6(2): 182-197.
10. Mostaghim S, Teich J. Strategies for finding good local guides in multi-objective particle swarm optimization (MOPSO)[C]//*Proceedings of the 2003 Swarm Intelligence Symposium*, 2003: 26-33.
11. Villalobos-Arias M A, Pulido G T, Coello C A C. A proposal to use stripes to maintain diversity in a multi-objective particle swarm optimizer[C]//*Proceedings of the 2005 Swarm Intelligence Symposium*, 2005: 22-29.
12. Gong M, Cai Q, Chen X, Ma L. Complex Network Clustering by Multiobjective Discrete Particle Swarm Optimization Based on Decomposition[J]. *IEEE Transactions on Evolutionary Computation*, 2014, 18(1): 82-97.
13. Shi, Y., Eberhart, R., 1998. A modified particle swarm optimizer. In: *Proceedings of 1998 IEEE Congress on Evolutionary Computation*. pp. 69-73.
14. Shi, Y., Eberhart, R., 1999. Empirical study of particle swarm optimization. In: *Proceedings of 1999 Congress on Evolutionary Computation*. Vol. 3. p. 3.
15. Ting, T. O., Shi, Y., Cheng, S., Lee, S., 2012. Exponential inertia weight for particle swarm optimization. In: *Advances in Swarm Intelligence*. pp. 83-90.
16. Zhou, Z., Shi, Y., 2011. Inertia weight adaption in particle swarm optimization algorithm. In: *Advances in Swarm Intelligence*. pp. 71-79.