Introduction to Hybrid algorithm

Bhupender Sharma

Department of Computer Science and Information Technology Dronacharya College Of Engineering, Farrukh Nagar, Gurgaon, IndiaEmail- bhupander0001@gmail.com

Ashwani Kumar

Faculty of Computer Science and Information technology Dronacharya College Of Engineering Farrukh Nagar, Gurgaon, India

Abstract

An algorithm is a set of step-by-step procedures, or a set of rules to follow,for completing a specific task or solving a particular problem. The word algorithm was first coined in the 9th century. Algorithms are all around us. Common examples include: the recipe for baking a cake, the method we use to solve a long division problem, the process of doing laundry, and the functionality of a search engine are allexamples of an algorithm.

Hybrid algorithms play a prominent role in improving the search capability of algorithms. Hybridization aims to combine the advantages of each algorithm to form a hybrid algorithm, while simultaneously trying to minimize any substantial disadvantage. In general, the outcome of hybridization can usually make some improvements in terms of either computational speed or accuracy. This chapter surveys recent advances in hybridizing different algorithms. Based on this survey, some crucial recommendations are suggested for further development of hybrid algorithms.

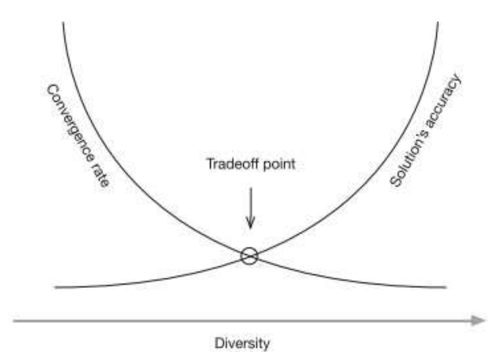
1 Introduction

Hybrid algorithms are two or more algorithms that run together and complement each other to produce a profitable cooperation from their integration. These algorithms are commonly known ashybrid metaheuristics (HMs).

The hybridization of EAs is popular, partly due to its better performance in handling noise, uncertainty, vagueness, and imprecision. For simplicity here, instead of using HM, we prefer to use the general term hybrid algorithms to refer to the similar notion. There are in fact two prominent issues of EAs in solving global and highly nonconvex optimization problem. These are:

- **Premature convergence:** The problem of premature convergence results in the lack of accuracy of the final solution. The final solution is a feasible solution close to the global optimal, often regarded as satisfactory or close-to-optimal solution.
- **Slow convergence:** Slow convergence means the solution quality does not improve sufficiently quickly. It shows stagnation or almost flat on a convergence graph (either a single iteration or the average of multiple iterations).

Fig. 1 Compromising accuracy and convergence rate



2. What Is Hybrid Algorithm?

A hybrid algorithm is an algorithm that combines two or more other algorithms that solve the same problem, and is mostly used in programming languages like C++, either choosing one (depending on the data), or switching between them over the course of the algorithm. This is done to combine desired features of each, so that the overall algorithm is better than the individual components.

"Hybrid algorithm" does not refer to simply combining multiple algorithms to solve a different problem – many algorithms can be considered as combinations of simpler pieces – but only to combining algorithms that solve the same problem, but differ in other characteristics, notably performance.

2.1. The Past

Evolutionary algorithms (EAs) are stochastic global optimizers that mimic the metaphor of biological evolution. They are always population-based algorithms that learn from the past searches by using a group of individuals or agents. These algorithms often possess behaviors

inspired by social or biological behaviors in the natural world. Loosely speaking, there are three categories of EAs, which are:

(I) Evolutionary Programming (EP)

(ii) Evolutionary Strategies (ES

(iii) Genetic Algorithms (GA)

2.2 The Present

The present developments tend to provide some improvements based on the extensive developments in last few decades, and researchers are still actively trying to design new hybrid algorithms. For example, in Rodriguez et al. developed hybrid metaheuristic by integrating an EA with Simulated Annealing (SA). In their review, they found that there were at about 312 publications indexed by ISI Web of Science that utilized both EA and SA algorithms. In comparison, there were only 123 publications that hybridized EAs with other metaheuristics such as the greedy search, iterated local search, descent gradient, and tabu search. However, Rodriguez et al.'s survey was limited to EAs and SA methods.

In the current literature, hybrid algorithms seem widely developed. Using Particle Swarm Optimization (PSO) as an example, the combination of PSO with other auxiliary search techniques seems highly effective in improving its performance. Genetic algorithm hybrids (or use of genetic operators with other methods) are by far the most widely studied. Genetic operators such as selection, crossover, and mutation have been integrated into PSO to produce better candidates. Differential evolution, ant colony optimization and conventional local search techniques have been used to combine with PSO.

2.3 The Future

Many new algorithms have been developed in recent years. For example, the bioinspired algorithms such as Artificial Bee Colony Algorithm (ABC), Bat Algorithm (BA), Cuckoo Search (CS), Firefly Algorithm (FA), Flower Pollination Algorithm (FPA), Glowworm Swarm Algorithm (GlowSA), Hunting Search Algorithm (HSA), Eagle Strategy (ES), Roach Infestation Optimization (RIO), Gravitational Search Algorithm (GravSA), Artificial Fish School Algorithm (AFS), Bacterial Evolutionary Algorithm (BEA), Artificial Plant Optimization Algorithm (APO), Krill Herd Algorithm (KHA) and others.

The list is expanding rapidly. These algorithms may possess entities and some novel characteristics for hybridization that remain to be discovered soon. However, it is worth pointing out that simple, random hybridization should not be encouraged. In a keynote talk by Xin-She Yang at the 15th EU workshop in metaheuristics and engineering (15th EU/ME) in Istanbul, Turkey in 2014, Yang warned about the danger of random hybridization.

Suppose there are n algorithms, if one chooses 2 k n to produce a hybrid, there will be

$$C_n^k = \binom{n}{k} = \frac{n!}{k!(n-k)!},$$

combinations. For n = 20 and k = 2, there will be 190 hybrids, while for n = 20 and k = 10, there will be 184,756 hybrids, which might produce 184,756 random (and almost useless) hybrids. Considering other combinations of k = 2, 3, …, 19, there will be about 2n = 1,048,576 hybrids. This estimate comes from the sum of binomial coefficients

$$\sum_{k=0}^{n} \binom{n}{k} = 2^n.$$

As it is really challenging to produce a good hybrid, we will try to summarize some observations and developments concerning hybrid algorithms in a very informal manner in the rest of this chapter.

3. Hybrid Recommendation Algorithm and Recommendation Strategy Based on Content and Collaborative Filtering

3.1. Research on the Hybrid Recommendation Model Based on Content and Collaborative Filtering

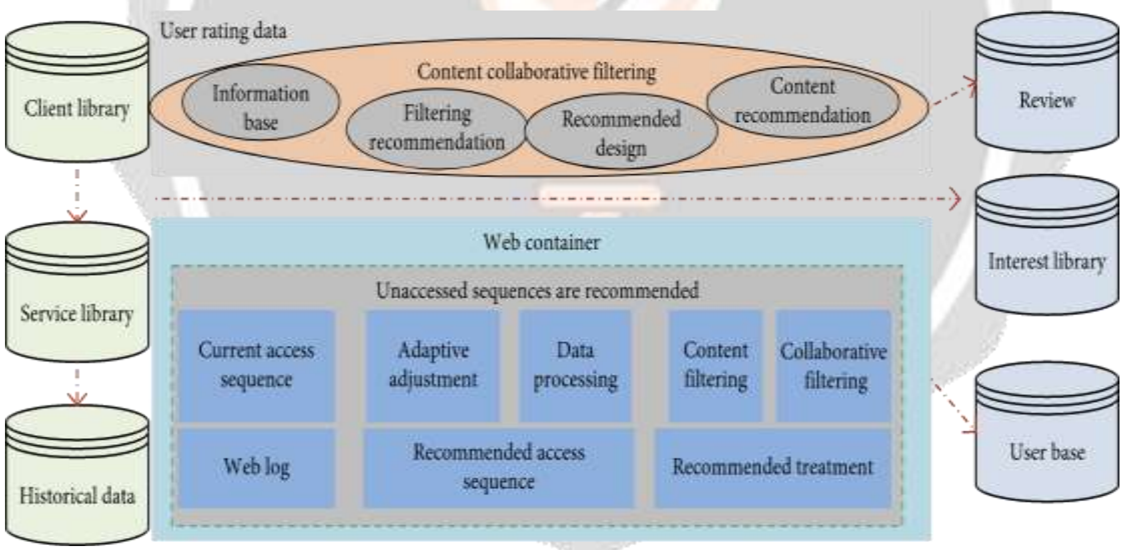


Figure 1

Schematic diagram of a hybrid recommendation model based on content and collaborative filtering.

The whole recommendation is divided into two modules, namely, content filtering recommendation module and collaborative filtering recommendation module, both of which are invisible to users. The preparation process of the dataset of the recommendation algorithm is as follows: first, the user's interest is extracted from the shopping history data of the user and the topic vector and feature vector preprocessed by the network log, and the data processing is established based on the recommendation module of content filtering. Then, based on user interest characteristics, user rating data, and current access sequence data, a recommendation module based on collaborative filtering is constructed to extract the nearest neighbors of the user and the nearest neighbors of the current access sequence (item). Then, it integrates two recommendation weighted sum calculation modules for the similarity calculation model of mixed recommendation (i.e., recommendation processing) to generate the recommended top visit sequence. The web server recommends the sequence to the user and accesses the recommendation sequence on the user, the adaptive adjustment of the recommended model, and the idle speed value of the feedback information to obtain the best recommendation data.

To realize personalized recommendation service, we must first collect the user's personal information and establish the user's interest characteristic description model. The ratio of the time spent browsing a web page to the number of characters on the page can effectively reveal the user's interest, which is related to the categories of information, and these categories are determinable and stable. Users browse the Web information including each page clicks, stay time, access, and order, and each page URL can be found in the Log of the proxy server, and the user visited Web pages can be found in the Cache of the proxy server, so you can go through the Web mining way to get the user's interest.

The optimal feature items are those words with the largest mutual information amount with the related text set Rel (Q). The logarithmic mutual information amount between the words and the related text set is calculated by the following formula:

The cosine similarity between user preference document and project document is

The higher the calculated similarity, the more preference the users have for this feature. The biggest problem facing TF-IDF is the choice of features. The content category of users is based on the similarity between user interests, that is, the similarity between user feature vectors. Here, the commonly used method of cosine of included angle is selected. The similarity of user interests is

Clustering is carried out according to the similarity between user feature vectors so that users with similar interests can be classified into one group for easy processing. Meanwhile, for new product information documents, a list of recommended users can be obtained by judging their categories. It is assumed that the classification of user sets is controlled manually, so the recommender systemclustering method can be adopted.

If the recommended access sequence is judged to be related to the user's interest, it will browse its relevant information, and then, the recommended access sequence will become the current accesssequence. When adjusting the model vector, the interest topic vector can be extracted from the current access sequence, and the feature vector can be extracted from the user's shopping history data and the Web log (the Web log has changed accordingly). The new model vector is obtained by the weighted sum operation of the topic vector and feature vector. Let the weights be A', B', C',and D', respectively.

3.2. Improved Content and Collaborative Filtering Algorithm Recommendation System Based on *K*-Means Clustering

A collaborative filtering recommendation algorithm based on *K*-means clustering is proposed. The new algorithm has two components: offline and online. When offline, first, users are clustered according to their characteristic data to form several clustering clusters. When online, the clustering cluster to which the target user belongs to is determined according to the similarity between the target user and each clustering center, to find the nearest neighbor in the cluster. Then, based on the preference of the nearest neighbor group to the project, we can predict the interest preference of the target users and finally get the recommendation.

The specific idea is to apply *K*-means clustering to collaborative filtering. For the whole user space, the similarity between users and the clustering center is calculated according to users' purchasing habits and scoring characteristics (that is, the user-item scoring matrix), and the clustering cluster is assigned to users according to the principle of nearest distance, thus the whole user space can be divided into several small groups. Based on the scoring characteristics of all users in each cluster, a virtual user is generated for each cluster. As the representative of all users in the cluster, the rating of the virtual user to the project can be the average of all users in the cluster to the project. At this point, the project ratings of all virtual users form a new search space (virtual user- project rating matrix), which replaces the original user-project rating matrix. When online recommendation is made, it only needs to calculate the similarity between target users and all virtual users, determine the cluster to which the target users belong according to the similarity level, search for neighbors in the cluster, and generate recommendation. The algorithm flow is shown in Figure 2.

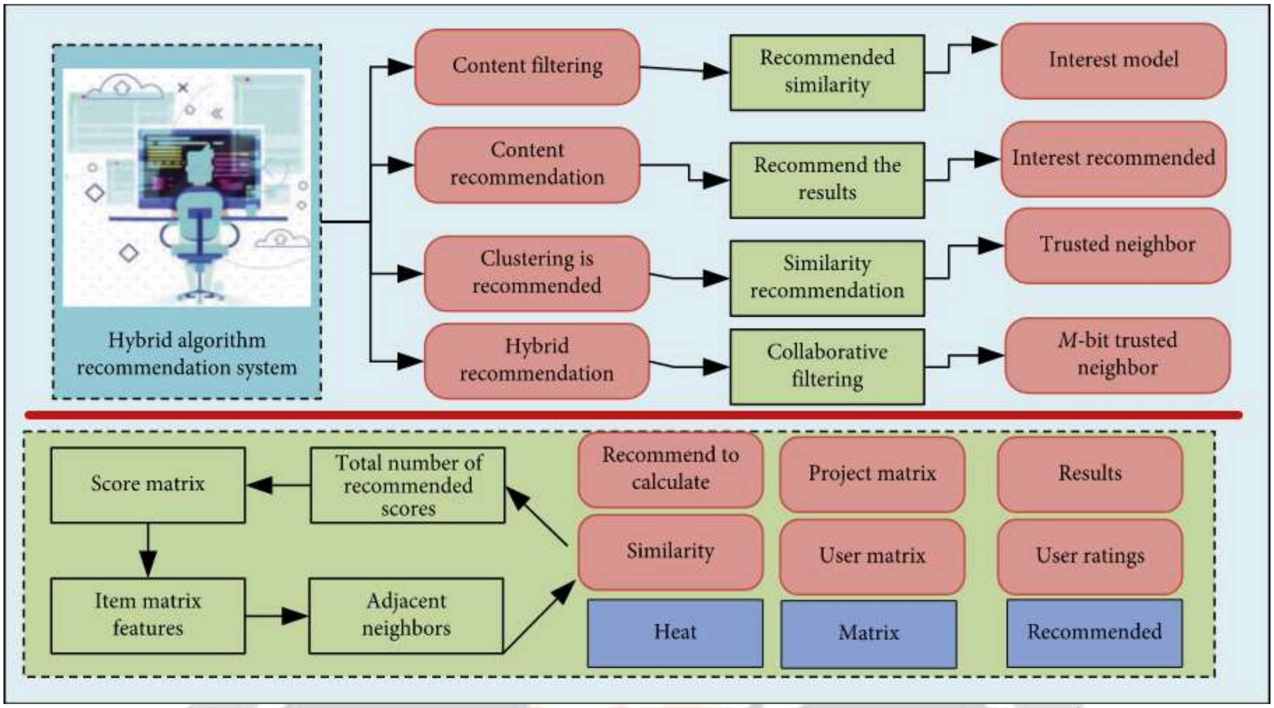


Figure 2

Recommendation system optimization diagram of the hybrid algorithm.

For a large recommendation system, there will be a lot of user and project data. However, only a small fraction of the total project space has been evaluated by users, which is known as the data sparsity problem. By clustering, the data dimension can be reduced. After neighborhood users are identified, the degree of preference of target users for unrated items can be predicted based on neighborhood users' preferences. The prediction scoring formula for the project is as follows:

When the hybrid recommendation algorithm starts to run, this article first uses the judgment conditions to process the user score data on the item. The total score of the project is less than 20 users. It is recommended to use them solely based on the content filtering method, which is also considered to be collaborative. The filtering recommendation algorithm for low-scoring data is effective in the fact that there are few user recommendations. The content-based recommendation algorithm is directly used to recommend equivalent items for users through item feature matching. This will result in mediocre recommendations, but it will also avoid the risk of invalid recommendations due to collaborative filtering of similar users not being able to find them. Of course, the value of 20 here is not fixed. In practical application, it can be adjusted according to the situation.

In addition, in the process of establishing the hybrid algorithm model proposed in this paper, the calculation method is not rigid with the traditional algorithm, but improved or innovated on the basis of the traditional calculation method, which is mainly reflected in the following points:(1)A method to optimize the user similarity calculation formula by using project heat was proposed(2)In order to present users' preferences more stereoscopic, the table-oriented feature extraction is carried out in the content-based recommendation algorithm, and the square-one method for

calculating users' similarity using the interest model is presented(3)According to the characteristics of the algorithm in this paper, a method to derive the weight coefficients of different features by using variance is proposed

The purpose of the content-based recommendation algorithm is to effectively filter out the third category of users whose interests are different from those of the target users, and the work required in this process includes three steps. The first step is to extract project features. The second step is to establish the user interest model.

4. Motivations for Hybridization

A hybrid algorithm, two or more algorithms are collectively and cooperatively solving a predefined problem.

- **Unified purpose hybrids:** Under this category, all sub-algorithms are utilized to solve the same problem directly; and different sub- algorithms are used in different search stages. Hybrid metaheuristic algorithms with local search is a typical example. The global Search explores the search space, while the local search is utilized to refine the areas that may contain the global optimum. Unified purpose hybrids. Under this category, all sub-algorithms are utilized to solve the same problem directly; and different sub- algorithms are used in different search stages. Hybrid metaheuristic algorithms with local search is a typical example. The global search explores the search space, while the local search is utilized to refine the areas that may contain the global optimum.

- **Multiple purpose hybrids:** One primary algorithm is utilized to solve the problem, while the sub-algorithm is applied to tune the parameters for the primary algorithm. For example, PSO can be applied to find the optimal value of mutation rate in GAs.

5. Taxonomy of Hybrid Algorithms

The goal of the general taxonomy is to provide a mechanism to allow comparisons of hybrid algorithms in a qualitative way. It is hoped that the categories and their relationships to each otherhave been chosen carefully enough to indicate areas in need of future work as well to help classify future work. Among existing taxonomies in other domains, one can find examples of flat and hierarchical classification schemes. The taxonomy could usefully be employed to classify any hybrid optimization algorithm. Metaheuristics are a general heuristic applicable to a large optimization problem.

Hybrid algorithms can be grouped into two categories. 5.1 Collaborative Hybrids

This involves a combination of two or more algorithms that work in sequence or in sequence. Thecontributing weight of each participating algorithm can be taken as a fraction and a half in the simplest case.

- **Multi-stage:** There are two stages involved in this case. The first algorithm acts as the global optimizer whereas the second algorithm performs local search. The first algorithm can explore the search space globally to locate promising area of convergence. Then the second algorithm will perform a deep local search like climbing a hill and a simplex descending path. A challenging problem in using such a system is knowing when to switchto a second algorithm.
- **Sequential:** In this structure, both algorithms are used separately until one integration process is met. For convenience, both algorithms will be used with the same number of repetitions before moving on to the next algorithm.
- **Parallel:** Two algorithms are used simultaneously, using the same number of people. One of the algorithms may be applied to the previously specified percentage of the algorithm.

Fig.1 Collaborative framework of hybrid algorithm, depicting multi-stage, sequential, andparallel structures

5.2 Interactive hybrids

In this feature, a single algorithm is subordinate, embedded in the expert metaheuristic. At this stage, the contributing weight of the second algorithm is estimated at 10-20%. This involves the installation of a deceptive user from the second algorithm to the main algorithm. For example, many algorithms have used a variable operator from GA to PSO, which has resulted in what is called Genetic PSO or Mutated PSO. Some may include gradient techniques such as hill climbs, steep descents, and Newton Raphson in the main algorithm.

There are two approaches:

- **Full manipulation:** All demographics are used regularly. This function can be integrated with existing source code, usually as subroutine / sub function.
- **Partial manipulation:** In this deception, only a fraction of the total population is accelerated using local search methods such as gradient methods. Choosing the right component and the right candidate to accelerate poses a major challenge in ensuring the success of this hybrid structure.

Fig. 2 Integrative structure of a hybrid algorithm, with full and partial manipulations

6. Disadvantages and Challenges of Hybrid Algorithms

- **Naming Convention**

(a) bit confusing to other researchers: HPSO-BFGS

(b) it may be interesting to compare the names of Hybrid GA-PSO (collaborative) to Mutated PSO (integrative), though those two hybrids combined GA with PSO.

- **2- Complexity of Hybrid Algorithm**
- **3- Computational Speed**

7. Recommendations for Future Developments

From the above analysis and observations, we can highlight some insights that should be useful to any future developments in this area as follows:

i) Simpler algorithm is preferred than more complex algorithms. Einstein once said: "Everything should be made as simple as possible, but not simpler." In the same sense, algorithms should be made as simple as possible.

ii) Shorter names are much preferred in the scientific community. Names should be as short as possible and 3–4 letters for the abbreviation.

iii) New hybrids (either collaborative or integrative hybrids) should have a clear structure that is easier for implementations. Ideally, any combination should be based on clear thinking, novel feature, and insightful mechanisms, which is more likely to produce better hybrids overall

8. Conclusion

In this chapter, we reviewed a wide range of hybrid algorithms and investigated the motivations of their developments. We have also categorized these algorithms, based on hybridization techniques. In addition, some drawbacks were discussed concerning hybridization. Recent examples of hybrid algorithm from the literature have been presented, with a summary of some prominent applications. Finally, some suggestions were recommended that can be useful to the future developments of hybrid algorithms.

References:

https://www.sciencedirect.com/topics/computer-science/hybrid-algorithm
https://en.wikipedia.org/wiki/Hyper-heuristic

Hybrid Evolutionary Algorithms: Methodologies, Architectures, and Reviews Crina Groshan &Ajith Abraham

A Taxonomy of Hybrid Metaheuristics: E.-G. TALBI Laboratoire d' Informatique Fundamentalde Lille, URA CNRS 369, Cite Scientifique, 59655 Villeneuve d' Asch Cedex, France

A Classification of Hyper-heuristic Approaches Edmund K. Burke, Matthew Hyde, Graham Kendall, Gabriela Ochoa, Ender¨Ozcanand John R. Woodward