2022 21st International Symposium on Distributed Computing and Applications for Business Engineering and Science (DCABES)

Frequent and High Utility Itemsets Mining Based on Bi-Objective Evolutionary Algorithm with An Improved Mutation Strategy

Chongyang Li
School of Artificial Intelligence & Computer Science
Jiangnan University
Wuxi, China
674209810@qq.com

***Abstract*—Frequent and high utility itemsets mining (FHUIM) is one of the important tasks in pattern mining. In order to solve the exponential search space and parameter setting problems that traditional HUIM algorithms encountered, the task of FHUIM was reformulated as a bi-objective problem that can be solved by multi-objective evolutionary algorithms (MOEAs). However, the search efficiency of the MOEAs may become lower when the total distinct items, the number of transactions, and the average length of transactions in the database are larger. To further improve the efficiency of MOEAs for FHUIM, we proposed FHUIM based on bi-objective evolutionary algorithm with an improved mutation strategy (FHUIM-BOEA-IMS). In FHUIM-BOEA-IMS, an improved mutation strategy is proposed to make the items with higher support and utility more likely to be saved in population, by which the FHUIs are more likely to be searched. The results on four popular datasets show that the proposed FHUIM-BOEA-IMS has better performance than the compared baseline in the task of FHUIM in terms of the convergence and final solutions.**

***Index Terms*—Frequent and high utility itemsets, Mutation strategy, Bi-objective evolutionary optimization**

I. Introduction

Frequent itemsets mining (FIM) is an important task in pattern mining [1], which aims to find out the frequent-occurring itemsets in database. However, FIM can not discover the itemsets with high profits. High utility itemsets mining (HUIM) was therefore proposed [2], by which the itemsets with high profits can be searched.

However, HUIM may find the itemsets with low support and FIM may find the itemsets with low utility. To meet the need of users more properly, the problem of frequent and high utility itemsets mining (FHUIM) is proposed, which is used to discover the frequent-occurring itemset with high utilities [3, 4]. Several algorithms for solving the task of FHUIM were proposed, the traditional algorithms for FHUIM have to meet the parameter setting problem and the huge search space [4]. The multi-objective evolutionary algorithms (MOEAs) [3] for FHUIM are then proposed to address the problems in traditional FHUIM algorithms.

In [3], the FHUIM are modelled as a multi-objective problem and solved by an MOEA named MOEA-FHUI. MOEA-FHUI can optimize the support and utility at the same time, and can get multiple solutions in one run. MOEAs have many advantages in the task of FHUIM compared to the traditional algorithms. However, the MOEAs may be inefficient when the scale of the datasets are larger [3]. It is more difficult for MOEAs to discover the FHUIs when the number of distinct items and transactions are larger. In order to improve the efficiency of MOEAs for FHUIM to deal with the large dataset, a novel mutation strategy is introduced here. The items in the final solutions are only a small part of the total distinct items (sparse nature of FHUIs) [5], thus these items are more expected to be saved in population. According to the analysis above, an novel mutation strategy is proposed, by which the items in the final solutions are more likely to appear in the population and the itemsets with high supports and utilities can be found more efficiently. Based on the proposed mutation strategy, a bi-objective evolutionary algorithm for FHUIM (FHUIM-BOEA-IMS) is proposed in this study. In summary, here are the contributions of this paper:

(1) Based on the sparse nature of FHUIs, an improved mutation strategy is proposed. The proposed mutation strategy can make the itemsets with high supports and utilities more likely to be generated, by which the search efficiency of the algorithm can be improved.

(2) Based on the proposed mutation strategy, a bi-objective evolutionary algorithm FHUIM-BOEA-IMS is proposed for FHUIM. The experimental results on four data mining datasets shows that the introduced FHUIM-BOEA-IMS has better performance than the baseline MOEA for FHUIM on the convergence and the quality of final solutions.

The remainder of the paper is structured as follows. We review the related works of FIM, HUIM, and FHUIM in Section II. Section III gives the preliminaries and problem statement of FHUIM. The proposed FHUIM-BOEA-IMS algorithm is presented in Section IV. We give the experimental results and analysis in Section V. The conclusions are given in Section VI.

2473-3636/22/$31.00 ©2022 IEEE
DOI 10.1109/DCABES57229.2022.00034

II. Related Works

A. FIM and HUIM

FIM was first proposed in [1], which can find the frequent-occurring itemsets based on their support values. Based on the anti-monotone property of the itemsets, Apriori is an early algorithm for FIM [6], in which the search space is reduced. Owning to the enumeration tree in FP_growth [7], the objective evaluating time and the search space can be reduced to search the frequent itemsets.

The task of HUIM was proposed [2] to discover the itemsets that can bring high profit to users, in which the measure of utility was defined to describe the profits of itemsets. To solve the problem of HUIM, Two-Phase algorithm was proposed, in which transaction weighted utilization measure was introduced in search space pruning. UP-Growth algorithm was proposed [8] to solve the problem of huge number of candidates in Two-Phase algorithm, in which the HUIs can be searched in the UP-Tree structure. However, traditional HUIM algorithms have to deal with the parameter setting problem and the exponential search space. To solve the problems that traditional algorithms meet in HUIM, several evolutionary algorithms were proposed [9, 10], by which users do not need to specific the minimum utility threshold and several solutions with relative high utilities can be discovered in acceptable time.

B. FHUIM

To discover the itemsets with high supports and utilities, the problem of FHUIM was proposed [11], and a Two-Phase algorithm was proposed to solve FHUIM. However, huge number of solutions can be obtained by using the Two-Phase algorithm in [11]. TKU-Miner algorithm was proposed to mine the top-K highest quality itemsets, in which three parameters have to be provided. To solve the problems of huge search space and parameter setting in FHUIM, the problem of FHUIM was formulated as a multi-objective problem that optimized by the proposed MOEA-FHUI [3]. The minimum threshold and prior parameters are not required, and the multi solutions can be obtained in one run within an acceptable time in MOEA-FHUI.

III. Preliminaries

Let $D = \{T_1, T_2, \ldots, T_n\}$ be a transaction database with n transactions. The set of total distinct items is the itemset $I = \{i_1, i_2, \ldots, i_m\}$ in D. Each transaction $T_q \in D$ $(1 \leq q \leq n)$ is a subset of I. An internal utility $q(i_j, T_q)$ $(1 \leq j \leq m, 1 \leq q \leq n)$ and an external utility $p(i_j)$ $(1 \leq j \leq m)$ are associated with each item in the transaction T_q. An itemset X is a non-empty subset of I. The set of transactions that contains all the items in X is T_X. Table I gives a small transaction database D, the external utilities of a, b, and c are 3, 4, and 2, the internal utilities of a, b, and c in each transaction are shown in Table I. Based on the given data, we use some examples to further explain the related definitions of FHUIM.

TABLE I: An example database.

T_{ID}	Transaction	Transaction Utility
T_1	$a:1, b:2$	11
T_2	$a:2, b:1, c:3$	16
T_3	$a:3$	9

Definition 1: The support of an itemset X, which is denoted as $sup(X)$ and calculated as:

$$sup(X) = |\{T | X \subseteq T, T \in D\}| \tag{1}$$

For itemset $\{a, b\}$ in Table I, it is contained in transactions T_1 and T_2. The support of the itemset $sup(\{a, b\})$ is 2. In traditional model of FIM, a frequent itemset refers to the itemset that the support of it is no less than the user specified minimum threshold. If we set 2 as the minimum threshold, then the itemset $\{a, b\}$ is a frequent itemset.

Definition 2: The utility of an itemset X is defined as:

$$uti(X) = p(i) \times \sum_{T_q \in T_X} \sum_{i \in X} q(i, T_q). \tag{2}$$

The utility of the itemset $\{a, b\}$ is $uti(a, b) = 1 \times 3 + 2 \times 4 + 2 \times 3 + 1 \times 4 = 21$. The itemset $\{a, b\}$ can be a HUI when the user specified minimum utility threshold is no less than 21,

Definition 3: The problem of FHUIM is defined as:

$$F(X) = maximize\ [f_1(X), f_2(X)] \tag{3}$$

$$where \begin{cases} X \subseteq I \\ f_1(X) = \frac{sup(X)}{\max\limits_{i \in I} sup(i)} \\ f_2(X) = \frac{uti(X)}{\max\limits_{i \in I} uti(i)} \end{cases}$$

In this paper, the task of FHUIM is formulated as a bi-objective optimization problem for its advantages in parameter setting and the trade off between the time and quality of solutions [3].

IV. Proposed algorithm FHUIM-BOEA-IMS

In this section, the proposed FHUIM-BOEA-IMS is demonstrated in detail. The overall procedure of FHUIM-BOEA-IMS is presented first. Then the proposed mutation strategy is introduced.

A. Algorithm Overview

The proposed FHUIM-BOEA-IMS follows the framework of NSGA-II [12] and adopts the binary encoding to encode the individual. The pseudocode of FHUIM-BOEA-IMS is given in Algorithm 1.

By scanning the database (line 1), the length of each individual can be obtained, which depends on the number of distinct items in D. The support values and the utility values of all the items are then evaluated (line 2). FHUIM-BOEA-IMS used the initialization strategy in MOEA-FHUI, which used the 1-item itemsets with high supports and the

Algorithm 1 FHUIM-BOEA-IMS

```
Input : $N$, population size; $MG$, maximum number of
    generation; $D$, the transaction dataset;
Output : $FS$, the final solutions;
 1: $len$ ← Number of distinct items in $D$; // Length of
    chromosome
 2: $Item\_Sup$, $Item\_Uti$ ← Calculate supports and utilities
    for all 1-itemsets;
 3: Initialization // Initialization strategy in MOEA-FHUI
 4: Evaluate the fitness of individuals in $P_1$; // equ. (3)
 5: Perform non-dominated sorting and crowding distance
    sorting for individuals in $P_1$;
 6: $g = 1$;
 7: while $g < MG$ do
 8:     $S$ ← Perform binary tournament selection on $P_g$;
 9:     $C$ ← Generate offsprings with one point crossover and
    improved mutation from $S$;
10:     Evaluate the fitness of individuals in $C$;
11:     $P_{g+1}$ ← Remove duplicate individuals from $P_g \cup C$;
12:     Perform non-dominated sorting and crowding distance
    sorting for individuals in $P_{g+1}$;
13:     $g \leftarrow g + 1$;
14: end while
15: FS ← Select the non-dominated itemsets from $P_{MG}$;
```

randomly selected transactions in D as the initial population (line 3). The fitness values of the initial population are calculated and the efficient sorting techniques based on non-dominated and crowding distance are performed for offspring generation (lines 4-5). In the evolutionary process, binary tournament selection operator, one point crossover operator, and the proposed mutation strategy are appointed to generate offspring (lines 8-9). The fitness of the offspring are calculated, the offspring and parents are merged with the the duplicate individuals removed (lines 10-11). Then the efficient sorting techniques based on non-dominated and crowding distance are performed to find the solutions with higher fitness, which make up the new population (line 12). Finally, when the termination condition of FHUIM-BOEA-IMS is reached, the non-dominated individuals decoded into corresponding itemsets for recommendation (line 15).

B. Improved Mutation strategy based on sparse nature of FHUIs

In FHUIM, the items in final solutions are only a small part of the itemset I, most of the items in I are useless for FHUIs [5]. If the important items have more possibility of saving in the individuals, the efficiency of discovering FHUIs can be improved. Based on the analysis above, an improved mutation strategy is proposed.

The proposed improved mutation strategy is based on the single point mutation. Before using the proposed mutation strategy, the score of each items are calculated by adding the supports and utilities of the 1-item itemsets up. The higher the score, the higher the potential support and utility of the item bring to the itemset. For each individual, two different items are selected randomly. If two items are both not in or both in the individual, the mutation strategy makes the item with higher score in the individual by flipping specific bit, which makes the individual more likely to be better. If the item with higher score is in the individual and the other is not, then the strategy does nothing. Otherwise, either of the two corresponding bits is flipped.

TABLE II: Characteristics of the used datasets.

Dataset	Transactions	Items	AvgLen
Mushroom	8124	119	23
Accident_10%	34019	336	34
Connect	67557	129	43
USCensus_10%	100000	396	48

V. Experimental Results

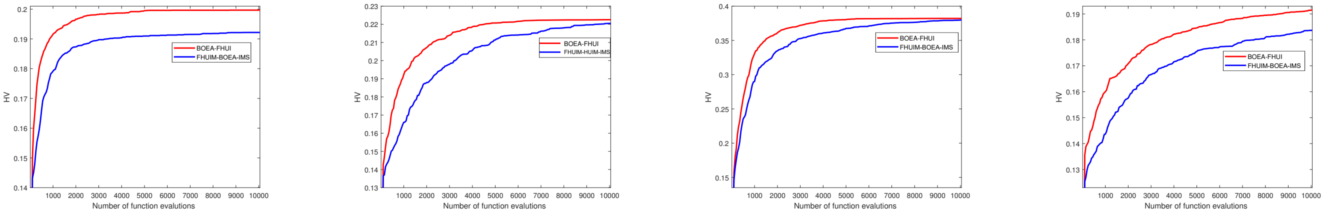
A. Experimental Settings

The proposed FHUIM-BOEA-IMS is compared with MOEA-FHUI for FHUIM in this section. Both FHUIM-BOEA-IMS and MOEA-FHUI use the framework of NSGA-II. Four datasets with different characteristic are adopted in experiment [13], which are shown in Table II. Both algorithms use binary tournament selection, one point crossover, and single point mutation (the proposed mutation strategy is based on the single point mutation) as their genetic operators. The mutation probability of the proposed FHUIM-BOEA-IMS is set to 1, and the mutation probability of MOEA-FHUI is as [3] provides. The number of function evaluations is set to 10000, the population size is set to 50. The hypervolume (HV) [14] was adopted as the performance metrics for the unknown Pareto fronts of FHUIM.

B. Comparison between FHUIM-BOEA-IMS and MOEA-FHUI

The comparison of convergence speed between FHUIM-BOEA-IMS and MOEA-FHUI is shown in Fig. 1. In Fig. 1, FHUIM-BOEA-IMS can coverage faster than MOEA-FHUI in all dataset. In Mushroom and USCensus_10%, the convergence speed of FHUIM-BOEA-IMS is much faster than MOEA-FHUI, and the final HV of FHUIM-BOEA-IMS is significantly larger than MOEA-FHUI. The number of total distinct items, the number of transactions, and the average length of transaction of USCensus_10% are the largest among the datasets, the better performance in USCensus_10% shows the efficient search ability of FHUIM-BOEA-IMS. For Connect and Accident_10%, The convergence speed of FHUIM-BOEA-IMS is obviously faster than MOEA-FHUI in first 5000 evaluations and the final HV of FHUIM-BOEA-IMS is no less than MOEA-FHUI, which reveals the search efficiency of FHUIM-BOEA-IMS.

Table III shows the HV results of FHUIM-BOEA-IMS and MOEA-FHUI in four datasets. For all datasets, the HV of FHUIM-BOEA-IMS is larger than MOEA-FHUI, which



(a) Mushroom

(b) Accident_10%

(c) Connect

(d) USCensus_10%

Fig. 1: Comparison of convergence speed of MOEA-FHUI and HUIM-BOEA-IMS.

TABLE III: HV results on all the four datasets by MOEA-FHUI and FHUIM-BOEA-IMS.

Dataset	Algorithm	
	FHUIM-BOEA-IMS	MOEA-FHUI
Mushroom	**2.00e-01±3.41e-03**	1.92e-01±3.93e-03
Accident_10%	**2.23e-01±3.15e-04**	2.21e-01±1.87e-03
Connect	**3.82e-01±1.30e-05**	3.80e-01±2.72e-03
USCensus_10%	**1.92e-01±4.43e-03**	1.76e-01±1.13e-01

shows the powerful search capability of FHUIM-BOEA-IMS. The standard deviation of FHUIM-BOEA-IMS is less than MOEA-FHUI in all datasets, which means FHUIM-BOEA-IMS is more stable than MOEA-FHUI.

VI. Conclusions

In this paper, in order to mine FHUIs more efficiently, we proposed a bi-objective evolutionary algorithm (FHUIM-BOEA-IMS). FHUIM-BOEA-IMS was based on a bi-objective itemsets mining problem model and can optimize the support and utility simultaneously. In FHUIM-BOEA-IMS, an improved mutation strategy was proposed, which can increase the number of high quality items in population. Experimental results demonstrated that, for the task of FHUIM, the proposed FHUIM-BOEA-IMS outperforms the baseline MOEA on the quality of final solutions and convergence speed.

References

[1] R. Agrawal, T. Imieliński, and A. Swami, "Mining association rules between sets of items in large databases," *SIGMOD Rec.*, vol. 22, no. 2, p. 207–216, 1993.

[2] H. Yao, H. J. Hamilton, and C. J. Butz, "A foundational approach to mining itemset utilities from databases," in *Proceedings of the 2004 SIAM International Conference on Data Mining*. SIAM, 2004, pp. 482–486.

[3] L. Zhang, G. Fu, F. Cheng, J. Qiu, and Y. Su, "A multi-objective evolutionary approach for mining frequent and high utility itemsets," *Applied Soft Computing*, vol. 62, pp. 974–986, 2018.

[4] L. Zhang, P. Luo, E. Chen, and M. Wang, "Revisiting bound estimation of pattern measures: A generic framework," *Information Sciences*, vol. 339, pp. 254–273, 2016.

[5] Y. Tian, R. Liu, X. Zhang, H. Ma, K. C. Tan, and Y. Jin, "A multi-population evolutionary algorithm for solving large-scale multi-modal multi-objective optimization problems," *IEEE Transactions on Evolutionary Computation*, p. 10.1109/tevc.2020.3044711, 2020.

[6] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules in large databases," in *Proceedings of the 20th International Conference on Very Large Data Bases*, ser. VLDB '94. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1994, p. 487–499.

[7] J. Han, J. Pei, and Y. Yin, "Mining frequent patterns without candidate generation," *SIGMOD Rec.*, vol. 29, no. 2, p. 1–12, 2000.

[8] V. S. Tseng, C.-W. Wu, B.-E. Shie, and P. S. Yu, "Up-growth: an efficient algorithm for high utility itemset mining," p. 253–262, 2010.

[9] J. C.-W. Lin, L. Yang, P. Fournier-Viger, J. M.-T. Wu, T.-P. Hong, L. S.-L. Wang, and J. Zhan, "Mining high-utility itemsets based on particle swarm optimization," *Engineering Applications of Artificial Intelligence*, vol. 55, pp. 320–330, 2016.

[10] Q. Zhang, W. Fang, J. Sun, and Q. Wang, "Improved genetic algorithm for high-utility itemset mining," *IEEE Access*, vol. 7, pp. 176 799–176 813, 2019.

[11] C. C. C. Yeh J S, Li Y C, "Two-phase algorithms for a novel utility-frequent mining model," in *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, 2007, pp. 433–444.

[12] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: Nsga-ii," *IEEE transactions on evolutionary computation*, vol. 6, no. 2, pp. 182–197, 2002.

[13] P. Fournier-Viger, J. C.-W. Lin, A. Gomariz, T. Gueniche, A. Soltani, Z. Deng, and H. T. Lam, "The spmf open-source data mining library version 2," in *Joint European conference on machine learning and knowledge discovery in databases*. Springer, 2016, pp. 36–40.

[14] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach," *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 4, pp. 257–271, 1999.