Multi-fair Capacitated Students-Topics Grouping Problem

Tai Le Quy[1(✉)], Gunnar Friege[2], and Eirini Ntoutsi[3]

[1] L3S Research Center, Leibniz University Hannover, Hannover, Germany
tai@l3s.de
[2] Institute for Didactics of Mathematics and Physics, Leibniz University Hannover, Hannover, Germany
friege@idmp.uni-hannover.de
[3] Research Institute CODE, University of the Bundeswehr Munich, Munich, Germany
eirini.ntoutsi@unibw.de

Abstract. Group work is a prevalent activity in educational settings, where students are often divided into topic-specific groups based on their preferences. The grouping should reflect students' aspirations as much as possible. Usually, the resulting groups should also be balanced in terms of protected attributes like gender, as studies suggest that students may learn better in mixed-gender groups. Moreover, to allow a fair workload across the groups, the cardinalities of the different groups should be balanced. In this paper, we introduce a *multi-fair capacitated* (MFC) grouping problem that fairly partitions students into non-overlapping groups while ensuring balanced group cardinalities (with a lower and an upper bound), and maximizing the diversity of members regarding the protected attribute. To obtain the MFC grouping, we propose three approaches: a greedy heuristic approach, a knapsack-based approach using vanilla maximal knapsack formulation, and an MFC knapsack approach based on group fairness knapsack formulation. Experimental results on a real dataset and a semi-synthetic dataset show that our proposed methods can satisfy students' preferences and deliver balanced and diverse groups regarding cardinality and the protected attribute, respectively.

Keywords: Fairness · Grouping · Knapsack · Educational data · Nash social welfare

1 Introduction

Teamwork plays a vital role in educational activities, as students can work together to achieve shared learning goals while learning about leadership, higher-order thinking, and conflict management [6]. A common approach to group students into teams is as follows: the instructor provides a list of topics, projects, tasks, etc. (shortly: *topics*), according to which the different non-overlapping groups of students should be formed. The grouping procedure can be performed randomly or based on students' preferences [14] typically expressed as a ranking over the provided topics. Or, the instructor just says: "Find yourself into

© The Author(s) 2023
H. Kashima et al. (Eds.): PAKDD 2023, LNAI 13935, pp. 507–519, 2023.
https://doi.org/10.1007/978-3-031-33374-3_40

groups"; in this case, a grouping is not random and does not consider students' preferences w.r.t. topics but it is triggered by social connections. The common case in educational settings is the grouping w.r.t. students' preferences.

The grouping process should consider various requirements. First, students' preferences should be taken into account (i.e., *student satisfaction*). A grouping is considered satisfactory if it can satisfy the students' preferences as much as possible. Second, the groups should be balanced in terms of their cardinalities, so all students share a similar workload (i.e., *group cardinality*) because when groups have unequal sizes, and the minority group is smaller than a critical size, the minority cohesion widens inequality [17]. Third, the instructor might be interested in fair-represented groups w.r.t. some protected attributes like gender or race [8] (i.e., *group fairness*), as studies suggest that mixed-gender grouping may have a positive effect on groups' performance [4].

These requirements have been discussed in the related work but are typically treated independently. For example, fairness w.r.t. workload distribution and students' preferences has been discussed in group assignments [6], assignment of group members to tasks [14] or students to projects [19]. Student satisfaction is typically assessed as the number of topics staffed [11] or the sum of the utilities of the topics assigned to students based on the ranking of preferences chosen by students [12]. The group cardinality can be satisfied by the heuristic method [15], or the hierarchical clustering approach [9]. However, providing a grouping solution that simultaneously satisfies all three requirements is hard [19].

To this end, we introduce *multi-fair capacitated (MFC) grouping* problem that aims to ensure fairness of the resulting groups in multiple aspects. In particular, we target fairness in terms of i) maximizing students' satisfaction, ii) ensuring fairness in group representation w.r.t. the protected attribute, and iii) balancing group cardinalities. For the satisfaction aspect, we employ the Nash social welfare notation [16]; for the fairness w.r.t protected attribute we use the balance score notion [3]. To solve the MFC problem, we propose three approaches: i) a greedy heuristic algorithm; ii) a knapsack-based approach that reformulates the assignment step as a maximal knapsack problem; iii) an MFC knapsack model based on the group fairness knapsack formulation [18].

2 Related Work

Agrawal et al. [1] proposed the problem of grouping students in a large class w.r.t. the overall gain of students. Miles et al. [14] investigated the problem of assignment of group members to tasks w.r.t. the workload distribution. Concerning a diversity of features such as skills, genders, and academic backgrounds, Krass et al. [8] investigated the problem of assigning students to multiple non-overlapping groups. However, students' preferences were not considered. To consider both efficiency and fairness, Magnanti et al. [12] solved a CPLEX integer programming formulation with two objectives: maximizing the total utility computed by the rank of student's preferences (efficiency) and minimizing the number of students assigned to the projects which they do not prefer (fairness). Besides, Rezaeinia et

al. [19] introduced a lexicographic approach to prioritize the goals. The efficiency objective is computed based on the utility, similar to [12]. A related problem is the problem of assigning reviewers to papers [7]. Each reviewer can be assigned to several papers, and each paper can be assigned to several reviewers [7]. However, in the students grouping problem, we attempt to generate non-overlapping groups [8], where each student can be assigned to only one group [19].

The knapsack problem formulation has been used for finding good clustering assignments [9] without students' preference and the minimum capacity of a group (cluster) is not considered. Recently, Stahl et al. [20] introduced a fair knapsack model to balance the price given by the data provider and the suggested price by the customer. Fluschnik et al. [5] proposed three concepts of fair knapsack (individually best, diverse and fair knapsack) to solve the problem of choosing a subset of items where the total cost is not greater than a given *budget* while taking into account the preferences of the voters. Fairness of the knapsack is measured by the Nash social welfare (or Nash equilibrium) [16]. The group fairness definition for the knapsack problem was investigated recently by Patel et al. [18]. In their study, each item is characterized by a *category*, their goal is to select a subset of items such that the total value of the selected items is maximized, and the total weight does not surpass a given weight while each category is *fairly* represented.

3 Problem Definition

Let $X = \{x_1, x_2, \cdots, x_n\}$ be a set of n students, $T = \{t_1, t_2, \cdots, t_m\}$ be a set of m topics. For an integer n we use [n] to denote the set $\{1, 2, \cdots, n\}$. Each student can choose h topics as their preference ($h \ll m$). The students' preferences are stored in matrix $wishes$. Row $wishes_i$ contains the list of h topics preferred by student i. We use a matrix V to record the student's level of interest in the topics. The preference of topic t_j chosen by student x_i is represented by a number v_{ij}. The more preferred topic will have a higher value of v_{ij}. Matrix V is computed as: $V_{i,wishes_{io}} = h/o$ with $o \in [h]$, where o indicates the order of preferences. Likewise, each topic t_j can be chosen by several students. A priority matrix W consists of values computed based on the registration time, where w_{ij} represents the priority of student x_i on topic t_j. Students who register earlier will have a higher value of w_{ij}. If the topic t_j is not preferred by student x_i then $v_{ij} = 0$ and $w_{ij} = 0$.

Let $\psi : V \times W \rightarrow \mathbb{R}$ be the aggregate function of matrices V and W. For each student x_i, we define a *welfare* value w.r.t. topic t_j: $welfare_{ij} = \psi(v_{ij}, w_{ij})$. In detail, $\psi(v_{ij}, w_{ij}) = \alpha v_{ij} + \beta w_{ij}$, where α and β are the parameters indicating the weight of each component. Figure 1 illustrates a dataset with 5 students and 4 topics. The matrix $welfare$ is computed with $\alpha = 1$ and $\beta = 1$ (preferences and registration time are equally considered).

The goal of a grouping problem is to distribute n students into k disjoint groups $\mathcal{G} = \{G_1, G_2, \cdots, G_k\}$, $(k \leq m)$, that maximizes the students' preferences w.r.t. the registration time, formulated by the objective function:

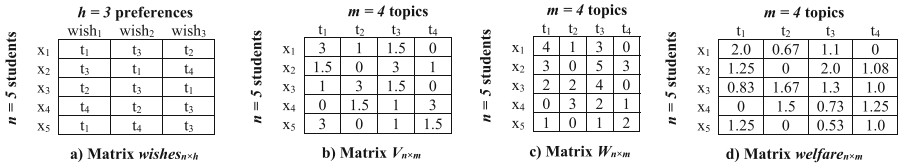

Fig. 1. A dataset with matrices $wishes$, V, W and $welfare$.

$$\mathcal{L}(X, \mathcal{G}) = \prod_{r=1}^{k} (1 + \sum_{i=1}^{n} welfare_{ij_r} \times y_{ij_r}) \tag{1}$$

In other words, the goal is to maximize the product of the total *welfare* obtained from each group G_r. In Eq. 1, a set of indexes $J = \{j_1, j_2, \cdots, j_k\}$ of k selected topics is defined as $J = \{j | x_i \in G_r, welfare_{ij} > 0\}$, $\forall r \in [k]$. Variable y_{ij_r} is the flag of x_i; $y_{ij_r} = 1$ if x_i is assigned to the group of topic t_{j_r}, otherwise $y_{ij_r} = 0$. Equation 1 is the representation of the Nash social welfare [16] function[1]. Therefore, we can call a grouping satisfactory if it maximizes the product in the objective function $\mathcal{L}(X, \mathcal{G})$. Furthermore, we add one to the sum $\sum_{i=1}^{n} welfare_{ij_r} \times y_{ij_r}$ to avoid the phenomenon that the sum of *welfare* in a certain group might be zero.

Fairness of Grouping w.r.t. a Protected Attribute: Assume that each student is characterized by a binary protected attribute $P = \{0, 1\}$, where 0 is the protected group (e.g., $gender = female$) and 1 is the non-protected group (e.g., $gender = male$). $\varphi : X \to P$ is the demographic category to which the student belongs. Fairness of a grouping $\mathcal{G}$ w.r.t. protected attribute [3] is computed as:

$$balance(\mathcal{G}) = \min_{\forall G_r \in \mathcal{G}} balance(G_r) \tag{2}$$

where fairness of a group G_r is the minimum ratio between two categories:

$$balance(G_r)_{\forall G_r \in \mathcal{G}} = \min \left(\frac{|\{x \in G_r \mid \varphi(x) = 0\}|}{|\{x \in G_r \mid \varphi(x) = 1\}|}, \frac{|\{x \in G_r \mid \varphi(x) = 1\}|}{|\{x \in G_r \mid \varphi(x) = 0\}|} \right) \tag{3}$$

Capacitated Grouping: Inspired by the capacitated clustering problem [15], we call a grouping *capacitated* if the cardinality of each group G_r, i.e., $|G_r|$, is between a given lower bound $C^l \geq 0$ and an upper bound $C^u \geq C^l$.

Definition 1. MFC grouping problem. We describe the MFC problem as finding a grouping $\mathcal{G} = \{G_1, G_2, \cdots, G_k\}$ that distributes a set of students X into k groups corresponding to k topics, and satisfies the following requirements:
1) The assignment is fair, i.e., maximizing students' satisfaction (Eq. 1);
2) The balance of each group G_r is maximized, i.e., the fairness constraint w.r.t. the protected attribute (Eq. 2);
3) The cardinality of each group $G_r \in \mathcal{G}$ is bounded within $[C^l, C^u]$.

[1] The Nash social welfare was defined as $\prod_{v_i \in V}(1 + \sum_{a \in S} u_i(a))$ [5] (the typical formula is $\prod_{v_i \in V} \sum_{a \in S} u_i(a)$, where v_i is a voter in a set of voters V, a is an item of the knapsack S, and $u_i(a)$ represents the extent to which v_i enjoys a. The knapsack S is fair if that product is maximized.

4 Methodology for the MFC Grouping Problem

To solve the MFC grouping problem, we first propose a greedy heuristic algorithm (Sect. 4.1); then we formulate the assignment phase as a vanilla maximal knapsack (Sect. 4.2) or a group fairness knapsack problem (Sect. 4.3).

4.1 A Greedy Heuristic Approach

We apply a 2-phase greedy strategy (Algorithm 1). Step 1: we maximize the students' preferences by assigning them to their most preferred topic. If a topic is preferred by many students we select the student who has the highest *welfare* value (lines 4, 5). Step 2: we adjust the assignment to satisfy the requirements by *GroupAdjustment* function (Algorithm 2). The number of students w.r.t. protected attribute $(p_0^l, p_0^u, p_1^l, p_1^u)$ are computed based on the resulting groups' cardinalities (C^l, C^u) and the balance score θ (line 2). If there exists ungrouped students, we try to assign them to the existing groups (lines 3 - 6). If all groups are full, we choose the most prevalent topic preferred by the remaining ungrouped students and assign them to such a topic (lines 7 - 11). We disband groups containing too few students and assign those ungrouped students to other groups until all groups have the desired capacity (lines 13 - 18).

Complexity: Step 1 consumes $\mathcal{O}(n \times h)$ and step 2 costs $\mathcal{O}(C^l \times n \times m)$ as the algorithm has to deal with every group having cardinality less than C^l. As $C^l \ll n$ and $C^u \ll n$, the complexity of the greedy heuristic model is $\mathcal{O}(n \times m)$.

Algorithm 1: Greedy heuristic algorithm

Input: X: a set of students; n: #students; h: #preferences; m: #topics; C^l, C^u: capacities ; matrices $wishes_{n\times h}$, $V_{n\times m}$, $W_{n\times m}$; θ: balance score
Output: A grouping with k groups

1 $groups \leftarrow \emptyset; welfare \leftarrow \psi(V, W)$;//Step 1: Assign students to groups;
2 **for** $i \leftarrow 1$ **to** n **do**
3 **for** $j \leftarrow 1$ **to** h **do**
4 **if** *(topic* $wishes_{ij}$ *is the most preferred topic of student* i*) and (*$welfare_{i,wish_{ij}}$ *is the highest value among students choosing topic* $wishes_{ij}$*) and (*$len(groups[wishes_{ij}] < C^l)$*)* **then**
5 $groups[wishes_{ij}]$.append(i);
6 GroupAdjustment($groups$) //Step 2: Adjustment;
7 **return** $groups$;

4.2 A Knapsack-Based Approach

The assignment of the greedy heuristic approach can be detrimental to students' satisfaction because there may be some students who have no more topics to be assigned. Therefore, we propose an approach to select the most suitable students for each topic by a *maximal knapsack* problem [13]. Let *capacity* be a cardinality array with $capacity_i = 1, \forall i \in [n]$; $welfare_{ij} = \psi(v_{ij}, w_{ij})$ and the indexes of k

Algorithm 2: Group adjustment algorithm

Input: *groups*: a set of groups; n: #students; h: #preferences; m: #topics; C^l, C^u: capacities; θ: balance score
Output: An adjusted grouping

Function `GroupAdjustment(`*groups*`):`
  $p_0^l \leftarrow \left\lceil \frac{C^l}{\frac{1+\theta}{\theta}} \right\rceil$; $p_0^u \leftarrow \left\lceil \frac{C^u}{\frac{1+\theta}{\theta}} \right\rceil$; $p_1^l \leftarrow C^l - p_0^l; p_1^u \leftarrow C^u - p_0^u + 1$;
  for $i \leftarrow 1$ **to** n **do**
    for $q \leftarrow 1$ **to** m **do**
      if *($i \notin groups[q]$) and len($groups[q] < C^l$) and (($n\_students\_0 < p_0^l$) or ($n\_students\_1 < p_1^l$))* **then**
        groups[q].append(i);
  while *len(unassigned_students)* > 0 **do**
    id $\leftarrow$ the most prevalent topic preferred by remaining students;
    for $i \in$ *unassigned_students* **do**
      if *len($groups[id]$) $< C^u$ and (($n\_students\_0 < p_0^u$) or ($n\_students\_1 < p_1^u$))* **then**
        groups[*id*].append(i);
  n_items $\leftarrow 1$;
  while *(cardinalities of all groups* $\notin [C^l, C^u]$*)* **do**
    if *n_items* $< C^l$ **then**
      Resolve the groups with cardinality *n_items*;
      if *($n\_students\_0 < p_0^u$) or ($n\_students\_1 < p_1^u$)* **then**
        Assign ungrouped students to the remaining groups having cardinality $< C^u$;
      n_items $++$;
return *groups*;

topics $J = \{j_1, j_2, \cdots, j_k\}$ will be chosen for the resulting groups. For each topic $t_{j_r} \in T$, $\forall r \in [k]$, i.e., r is the index of the selected knapsack, the goal is to select a subset of students (G_r), such that:

$$\text{maximize} \sum_{i=1}^{n} welfare_{ij_r} \times y_{ij_r} \text{s.t.} \quad \begin{cases} \sum_{i=1}^{n} capacity_i \times y_{ij_r} \leq C^u \text{ or} \\ \sum_{i=1}^{n} capacity_i \times y_{ij_r} \leq C^l \end{cases} \tag{4}$$

where $y_{ij_r} = 1$ if x_i is assigned to the group of topic t_{j_r}, otherwise $y_{ij_r} = 0$.

In other words, for each selected topic, we find a set of students that maximizes the total *welfare*, while the total *capacity*, is within the given bounds. The pseudo-code is described in Algorithm 3 with two steps. Step 1: we find the most suitable candidates among the unassigned students by the solution of a maximal knapsack problem [13] for each topic. We use dynamic programming to solve the maximal knapsack problem (Eq. 4). Step 2 is presented in Algorithm 2 which performs a fine-tuning of the assignment.

Complexity: In step 1, the complexity is $\mathcal{O}(m \times n \times C^u)$ since it costs $\mathcal{O}(n \times C^u)$ for each topic to solve the knapsack problem. The running time of step 2 is $\mathcal{O}(C^l \times n \times m)$. Therefore, the complexity is $\mathcal{O}(n \times m)$.

Algorithm 3: Knapsack-based algorithm

Input: X: a set of students; n: #students; h: #preferences; m: #topics; C^l, C^u: capacities; matrices $wishes_{n\times h}$; $V_{n\times m}$; $W_{n\times m}$.
Output: A grouping with k groups

```
1  groups ← ∅ //Step 1: Assign students to groups ;
2  welfare ← ψ(V, W) ;
3  for id ← 1 to m do
4      capacity ← get_capacity(unassigned_students);
5      values ← get_welfare(unassigned_students, welfare);
6      n_items ← len(unassigned_students);
7      if n_items > 0 then
8          if n mod C^l = 0 then
9              selected_students ← knapsack(values, capacity, n, C^l);
10         else
11             selected_students ← knapsack(values, capacity, n, C^u);
12         groups[id] ← selected_students;
13     GroupAdjustment(groups) //Step 2: Adjustment;
14 return groups;
```

4.3 An MFC Knapsack Approach

In the knapsack-based approach, the fairness constraint w.r.t. the protected attribute is not directly considered in the knapsack formulation. Inspired by the knapsack problem with *group fairness* constraints of Patel et al. [18], we propose an *MFC knapsack* algorithm to find the group of suitable students, which satisfies the MFC problem' requirements. The goal of the MFC knapsack is to select a subset of student (G_r), such that:

$$\text{maximize} \sum_{i=1}^{n} welfare_{ij_r} \times y_{ij_r} \text{s.t.} \quad \begin{cases} \sum_{i=1}^{n} capacity_i \times y_{ij_r} \leq C^u \text{ or} \\ \sum_{i=1}^{n} capacity_i \times y_{ij_r} \leq C^l \\ balance(G_r) \text{ is maximized} \end{cases} \tag{5}$$

where $y_{ij_r} = 1$ if x_i is assigned to the group of topic t_{j_r}, otherwise $y_{ij_r} = 0$.

We use dynamic programming to solve the MFC knapsack problem (Algorithm 4). The input parameters include a set of unassigned students $\mathcal{S} \subseteq X$. A dynamic programming table $\mathcal{A}(p, s, w)$ is used to record the total welfare of the first s students in the set $\mathcal{S}$ with capacity w on group p, $\forall p \in \{0, 1\}$, e.g., $\{male, female\}$ w.r.t. protected attribute (line 3, 4). Then, we construct table $\mathcal{B}(p, w)$ to find the total welfare with capacity w w.r.t. the protected attribute. The number of students in the protected group and the non-protected group is computed based on a given balance score θ (line 6). We apply a two-phase approach to solve the MFC grouping problem. Step 1, we assign students to groups based on the MFC knapsack's solution. We replace the *knapsack* function in Algorithm 3 with the new *MFC knapsack* function (Algorithm 4). Step 2, we use the group adjustment algorithm (Algorithm 2) to fine-tune the assignment.

Complexity: The MFC knapsack takes $\mathcal{O}(n \times C^u)$ for each topic. To solve the MFC problem, step 1 consumes $\mathcal{O}(m \times n \times C^u)$, and step 2 costs $\mathcal{O}(C^l \times n \times m)$. Therefore, the complexity of the MFC knapsack approach is $\mathcal{O}(n \times m)$.

Algorithm 4: MFC knapsack algorithm

Input: $\mathcal{S} = \{x_1, x_2, \ldots, x_z\}$: a set of unassigned students; C^l, C^u: capacities; $welfare_{n \times m}$: a welfare matrix; θ: balance score
Output: An optimal total welfare value

1 $avg = \frac{\sum_{i=1}^{n} welfare_{ij_r}}{(C^l + C^u)/2}$;
2 Let $\mathcal{A}(p, s, w), \forall p \in \{0, 1\}$, be the total welfare of the first s students in the set $\mathcal{S}$ with capacity w on group p ;
3 Initialize $\mathcal{A}(p, 0, w) \leftarrow 0$; $\mathcal{A}(p, s, 0) \leftarrow 0$;
4 $\mathcal{A}(p, s, w) \leftarrow max\{\mathcal{A}(p, s-1, w), \mathcal{A}(p, s-1, w-1) + \sum_{i=1}^{s} welfare_{ij_r}\}$;
5 Let $\mathcal{B}(p, w)$ be the total welfare of group p with capacity w;
6 $p_0^l \leftarrow \left\lceil \frac{C^l}{\frac{1+\theta}{\theta}} \right\rceil$; $p_0^u \leftarrow \left\lceil \frac{C^u}{\frac{1+\theta}{\theta}} \right\rceil$; $S_0 \leftarrow \{x \in \mathcal{S} | \varphi(x) = 0\}$; $S_1 \leftarrow \{x \in \mathcal{S} | \varphi(x) = 1\}$;
7 $\mathcal{B}(0, w) \leftarrow max\{\mathcal{A}(0, |S_0|, w) | p_0^l \leq w \leq p_0^u\}$;
8 $\mathcal{B}(1, w) \leftarrow max\{\mathcal{B}(0, w') + \mathcal{A}(1, |S_1|, w - w') | C^l - p_0^l \leq w - w' \leq C^u - p_0^u, p_0^l \leq w' \leq p_0^u$, and $\frac{w'}{w - w'} \geq \theta\}$;
9 **return** $argmax\{\mathcal{B}(1, w) | min\{\mathcal{B}(1, w) - avg\}\}$;

5 Evaluation

5.1 Datasets

We evaluate our proposed methods on two variations of the student performance dataset [10] and a real data science dataset collected at our institute (Table 1).

Real Data Science Dataset. This dataset is collected in a seminar on data science at our institute. Students have to register 3 desired topics out of 16 topics. The advisor assigns students into groups based on their preferences and the registration time. The data contain demographic information of students (*ID, Name, Gender*) with their preferences (*wish1, wish2, wish3*), registration time (*Time*) and priority matrix W represented by 16 attributes (*T1, ..., T16*).

Student Performance Dataset[2]. The dataset consists of demographic, including the protected attribute *gender* which is used in the evaluation, school-related attributes and grades of students in Mathematics and Portuguese subjects of two Portuguese schools in 2005 - 2006. Because there is no given information about the topics and preferences of students in the original dataset, we create a *semi-synthetic* version by generating preferences and topics. For each student, we randomly generate h different preferred topics. Then, for each topic, we list the students who select the topic and randomly generate (different) priorities and store them in m attributes (matrix W). Hence, the *semi-synthetic* version has $(h + m)$ new attributes apart from the original attributes.

[2] https://archive.ics.uci.edu/ml/datasets/Student+Performance.

Table 1. An overview of the datasets.

Dataset	#instances	#attributes	Protected attribute	Balance score
Real data science	24	23	Gender (F: 8, M: 16)	0.5
Student - Mathematics	395	33	Gender (F: 208, M: 187)	0.899
Student - Portuguese	649	33	Gender (F: 383; M: 266)	0.695

5.2 Experimental Setup

Parameter Selection. We set the number of wishes $h = 3$ for the student performance dataset in order to be consistent with the real data science dataset. The number of topics, $m = 200$ and $m = 325$, are set for the student performance dataset - Mathematics and Portuguese subjects, respectively, to ensure that each group has at least 2 students. Besides, we set the parameters $\alpha = 1.0$ and $\beta = 1.0$, i.e., each component has the same weight. The balance scores θ are computed based on the datasets (Table 1). Furthermore, since the real data science dataset is very small, our methods are evaluated with the lower bound C^l in the range of $(2, \ldots, 8)$. Regarding the student performance dataset, we set $C^l = (2, \ldots, 18)$, as the average number of students per group should not exceed 20 [21]. The upper bound C^u is set as $C^u = C^l + 1$ for all datasets.

Baseline. The CPLEX integer programming model which considers both efficiency and fairness [12].

Evaluation Measures. We report the results on the following measures:

- Nash Social Welfare. The Nash social welfare is computed by Eq. 1. However, the number of groups (k) is determined during the group assignment process, i.e., k is different for the same set C^l, C^u, for each method. Hence, we normalize the Nash social welfare of the final grouping by $Nash = log_k \mathcal{L}(X, \mathcal{G})$.

- Balance. The fairness in terms of the protected attribute (Eq. 2).

- Satisfaction Level. It is computed by the ratio of the number of satisfied students, i.e., the students are assigned to their preferred topic, out of the total number of students: $Satisfaction = \frac{|\{i | wishes_{ip} = k, i \in groups_k, p \in [h]\}|}{n}$.

5.3 Experimental Results

Real Data Science Dataset. In Fig. 2, we present the performance of proposed methods on various evaluation measures. The MFC knapsack method is better in terms of the Nash social welfare and satisfaction level (Fig. 2-a, c). In terms of fairness w.r.t. protected attribute, the MFC knapsack method outperforms others when a group has at least 4 people (Fig. 2-b). CPLEX fails to assign students while maintaining only a constant number of groups (Fig. 2-d).

Student Performance - Mathematics Dataset. The knapsack-based approach outperforms others regarding Nash social welfare and satisfaction level in most experiments (Fig. 3-a, c). The satisfaction level tends to decrease because students have only a limited number of preferences (3 topics). When the group's

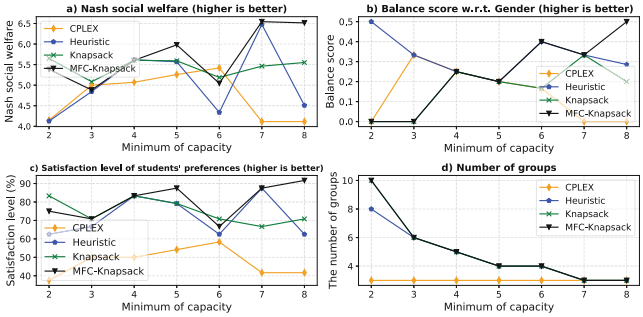

Fig. 2. Performance of methods on the real data science dataset.

cardinality increases, the desired topics become more diverse, and it is challenging to satisfy most students. In terms of fairness w.r.t. protected attribute (*gender*), the knapsack-based and MFC knapsack methods tend to achieve a higher balance score in comparison to the heuristic method (Fig. 3-b). When groups' cardinality is less than 4, the greedy heuristic and MFC knapsack methods tend to create more groups than the knapsack-based method (Fig. 3-d). The CPLEX method cannot return a solution when the groups' cardinality is less than 9 and it also fails since it is not possible to assign all students to groups.

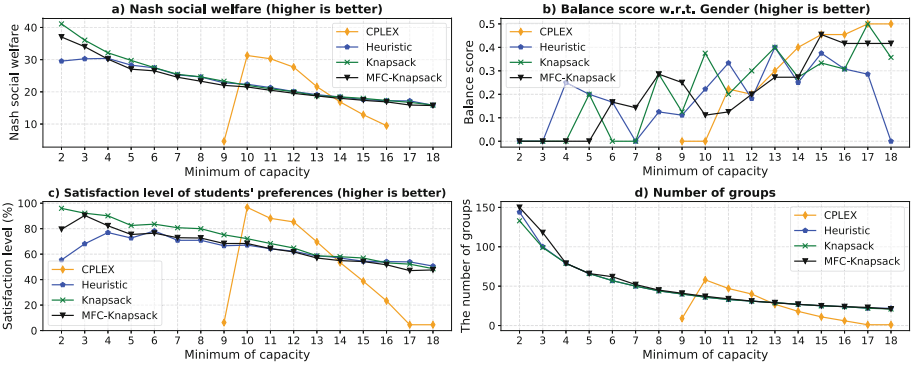

Fig. 3. Performance of methods on the student performance - Mathematics dataset.

Student Performance - Portuguese Dataset. The knapsack-based method once again demonstrates the ability to create groups with higher Nash social welfare and satisfaction level than others in many cases (Fig. 4-a and Fig. 4-c). Regarding fairness w.r.t. gender, a higher and more stable balance score is observed in the grouping generated by the MFC knapsack model (Fig. 4-b). The main reason for this phenomenon can be attributed to the model's emphasis on maximizing the balance constraint w.r.t. protected attribute. Besides, the

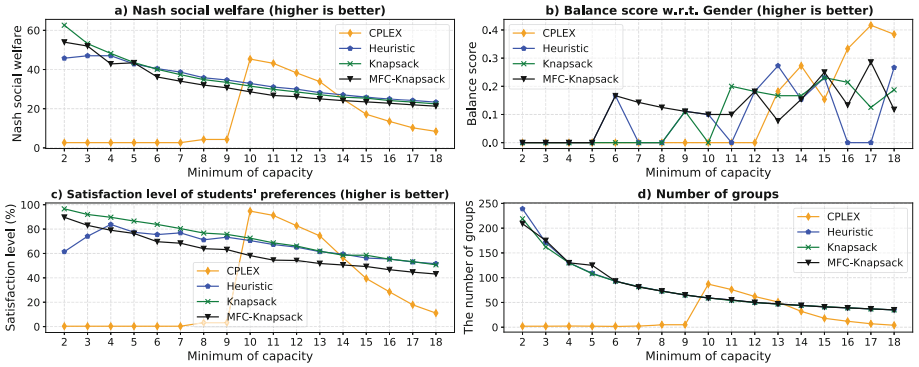

Fig. 4. Performance of methods on the student performance - Portuguese dataset.

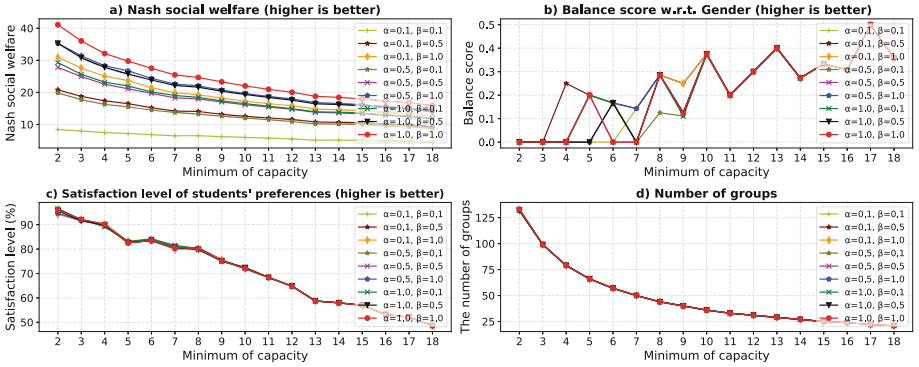

Fig. 5. Impact of α, β parameters on the knapsack-based model (student performance - Mathematics dataset).

MFC knapsack and greedy heuristic models divide students into more groups (Fig. 4-d) while the CPLEX also cannot assign all students to groups.

Impact of Parameters. The influence of α, β parameters is illustrated in Fig 5. The knapsack-based model shows the best performance with the combination of $\alpha = 1.0$ and $\beta = 1.0$.

Summary of Results. In general, the knapsack-based approach outperforms other models regarding Nash social welfare and satisfaction level. The MFC knapsack method shows its preeminence in terms of fairness w.r.t. gender in many cases, especially when the resulting groups have more members. However, in some cases, the knapsack-based approach tends to create fewer groups than the greedy heuristic method, i.e., the groups' cardinality is higher, which has both advantages and disadvantages. On the one hand, the larger groups can produce more ideas in brainstorming and discussions [2]. On the other hand, the group's performance may decline with the increase in the group's size [22].

6 Conclusions and Outlook

In this work, we introduced the MFC grouping problem that ensures fairness in multiple aspects: i) in terms of student satisfaction and ii) regarding the protected attribute and maintaining the groups' cardinality within the given bounds. We proposed three methods: the greedy heuristic approach that prioritizes the students' preferences in the assignment; the knapsack-based approach with the assignment step is formulated as a maximal knapsack problem; the MFC knapsack method considers fairness, cardinality, and students' preferences in the MFC knapsack formulation. The experiments show that our methods are effective regarding student satisfaction and fairness w.r.t. the protected attribute while maintaining cardinality within the given bounds. In the future, we plan to extend our approach to more than one protected attribute, as well as to further investigate the groups' characteristics w.r.t. students' abilities, and other definitions with different aspects of fairness in the educational settings.

Acknowledgements. The work of the first author is supported by the Ministry of Science and Culture of Lower Saxony, Germany, within the Ph.D. program "LernMINT: Data-assisted teaching in the MINT subjects".

References

1. Agrawal, R., Golshan, B., Terzi, E.: Grouping students in educational settings. In: KDD, pp. 1017–1026 (2014)
2. Bouchard Jr, T.J., Hare, M.: Size, performance, and potential in brainstorming groups. J. Appl. Psychol. **54**(1p1), 51 (1970)
3. Chierichetti, F., Kumar, R., Lattanzi, S., Vassilvitskii, S.: Fair clustering through fairlets. In: NeurIPS, pp. 5036–5044 (2017)
4. Fenwick, G.D., Neal, D.J.: Effect of gender composition on group performance. Gender Work Organization **8**(2), 205–225 (2001)
5. Fluschnik, T., Skowron, P., Triphaus, M., Wilker, K.: Fair knapsack. In: AAAI, vol. 33, pp. 1941–1948 (2019)
6. Ford, M., Morice, J.: How fair are group assignments? a survey of students and faculty and a modest proposal. J. Inf. Technol. Educ. Res. **2**(1), 367–378 (2003)
7. Hartvigsen, D., Wei, J.C., Czuchlewski, R.: The conference paper-reviewer assignment problem. Decis. Sci. **30**(3), 865–876 (1999)
8. Krass, D., Ovchinnikov, A.: The university of Toronto's rotman school of management uses management science to create MBA study groups. Interfaces **36**(2), 126–137 (2006)
9. Le Quy, T., Roy, A., Friege, G., Ntoutsi, E.: Fair-capacitated clustering. In: The 14th International Conference on Educational Data Mining, pp. 407–414 (2021)
10. Le Quy, T., Roy, A., Vasileios, I., Wenbin, Z., Ntoutsi, E.: A survey on datasets for fairness-aware machine learning. WIREs Data Min. Knowl. Discov. **12**(3) (2022)
11. Lopes, L., Aronson, M., Carstensen, G., Smith, C.: Optimization support for senior design project assignments. Interfaces **38**(6), 448–464 (2008)
12. Magnanti, T.L., Natarajan, K.: Allocating students to multidisciplinary capstone projects using discrete optimization. Interfaces **48**(3), 204–216 (2018)

13. Mathews, G.B.: On the partition of numbers. Proc. Lond. Math. Soc. **1**(1), 486–490 (1896)
14. Miles, J.A., Klein, H.J.: The fairness of assigning group members to tasks. Group Organization Manage. **23**(1), 71–96 (1998)
15. Mulvey, J.M., Beck, M.P.: Solving capacitated clustering problems. Eur. J. Oper. Res. **18**(3), 339–348 (1984)
16. Nash, J.F.: The bargaining problem. Econometrica **18**(2), 155–162 (1950)
17. Oliveira, M., Karimi, F., Zens, M., Schaible, J., Génois, M., Strohmaier, M.: Group mixing drives inequality in face-to-face gatherings. Commun. Phys. **5**(1) (2022)
18. Patel, D., Khan, A., Louis, A.: Group fairness for knapsack problems. In: AAMAS, pp. 1001–1009 (2021)
19. Rezaeinia, N., Góez, J.C., Guajardo, M.: Efficiency and fairness criteria in the assignment of students to projects. Annals of Operations Research, pp. 1–19 (2021)
20. Stahl, F., Vossen, G.: Fair knapsack pricing for data marketplaces. In: ADBIS, pp. 46–59. Springer (2016)
21. Urbina Nájera, A.B., De La Calleja, J., Medina, M.A.: Associating students and teachers for tutoring in higher education using clustering and data mining. Comput. Appl. Eng. Educ. **25**(5), 823–832 (2017)
22. Yetton, P., Bottger, P.: The relationships among group size, member ability, social decision schemes, and performance. Organ. Behav. Hum. Perform. **32**(2) (1983)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (http://creativecommons.org/licenses/by/4.0/), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.