Peer Learning Through Targeted Dynamic Groups Formation

Dong Wei
CS, NJIT
dw277@njit.edu

Ioannis Koutis
CS, NJIT
ikoutis@njit.edu

Senjuti Basu Roy
CS, NJIT
senjutib@njit.edu

***Abstract*— Peer groups leverage the presence of knowledgeable individuals in order to increase the knowledge level of other participants. The 'smart' formation of peer groups can thus play a crucial role in educational settings, including online social networks and learning platforms. Indeed, the targeted groups formation problem, where the objective is to maximize a measure of aggregate knowledge, has received considerable attention in recent literature. In this paper we initiate a *dynamic* variant of the problem that, unlike previous works, allows the change of group composition over time while still targeting to maximize the aggregated knowledge level. The problem is studied in a principled way, using a realistic learning gain function and for two different interaction modes among the group members. On the algorithmic side, we present DyGroups, a generic algorithmic framework that is greedy in nature and highly scalable. We present non-trivial proofs to demonstrate theoretical guarantees for DyGroups in a special case. We also present real peer learning experiments with humans, and perform synthetic data experiments to demonstrate the effectiveness of our proposed solutions by comparing against multiple appropriately selected baseline algorithms.**

I. Introduction

Online social networks and learning platforms enable the formation of targeted groups for peer learning. As an example, peer learning associations[1], social Q&A sites[2], even crowdsourcing platforms[3] investigate how interaction between like-minded individuals can improve knowledge and understanding on a topic, or simply promote improved well-being of the individuals. Indeed, systematic targeted groups formation can leverage the presence of knowledgeable individuals in order to educate group participants on a myriad of topics, and support efforts to dispel rumors and misinformation.

A. Novelty

The importance of targeted groups formation has been recognized in the literature. Recent works have studied the effect of *targeted one-shot* groups formation to optimize peer learning [1], [2], where the objective is to form a set of groups to maximize some type of aggregate learning. These works view groups as *static*, in the sense that every individual is assumed to be a member of only one group through the end of the process. What is not studied *is the effect of time*, and more concretely the potential of allowing the formation of a targeted set of groups to be repeated a certain number of times, as opposed to a single shot process. We hypothesize that *dynamic* group formation will enable more individuals to 'learn from the best', can better utilize intermediate learning gains, and has the potential to improve overall peer learning outcomes, both in theory and practice. Ours is the **first** systematic effort to model this problem and study algorithms for it.

B. Practical Motivation

Imagine a peer learning scenario in a physical classroom or on an online learning platform. Separating the participants into equal-size groups for homework assignments or projects is common practice to ensure relatively similar workload among students, while enabling effective interaction among the peers [3]. However, in the case when there are multiple group homework assignments or projects, *fixed* groups may be not optimal. Research has shown that successful groups evolve naturally [4], [5], and that the ability of dynamically altering group composition results in groups that persist for longer [6], [7]. This suggests that dynamic groups may offer benefits. Going back to the classroom example, it would be intuitively better to change the group membership of the students across the assignments so that everyone gets the opportunity to learn from the best participants, with the hope that the total 'educational welfare' is maximized.

C. Technical Contributions

We undertake the first formal attempt to study dynamic group formation to optimize peer learning. We follow previous related works [1], [8]–[10] and adopt common definitions and assumptions used to quantify the single-shot group formation. We then introduce a vast generalization and study *the effect of time and how the flexibility of changing membership and learning from others can improve the outcome*. Specifically, we make the following contributions:

(i). We initiate the study of the targeted dynamic groups formation (referred to as Targeted Dynamic Grouping or TDG) problem. We assume that the process consists of a predefined number of rounds (α). Each round entails a grouping of the participants into groups of size (k), and a defined learning gain for each individual, controlled by a *linear* learning gain function f. We consider two different interaction modes of learning within each group. One induces a *star*, where the interaction of any group member is limited only to the most knowledgeable individual of that group. The other induces a *clique*-like structure, where all possible pairs of within-group interactions take place. Our goal is to find dynamic groupings that *maximize the aggregate learning gain* after α rounds.

(ii). We delve into an investigation of the nuances of our proposed problem. We present an algorithmic framework DyGroups that is greedy in nature, and highly scalable.

[1]https://peerlearningassociation.weebly.com/
[2]https://www.quora.com
[3]https://appen.com/

DyGroups runs in $\Theta(\alpha n)$ time for both *clique* and *star* interaction modes, where n is the number of members. The greedy approach comes with an *interesting twist*. In each round, there are multiple k-groupings that maximize the aggregate learning for that round. Among them, DyGroups selects the one that also maximizes the variance of the participants' skills after the round. We also present an in-depth proof of the optimality of the proposed algorithm in a special case. We prove that DyGroups can always find the overall optimal solution for the *star* interaction structure when $k = 2$. However, extending the optimality proof to more general cases ($k > 2$) appears to be a far more difficult and interesting problem.

(iii). We run two independent rigorous experiments on real fact-learning in peer groups comparing multiple baseline algorithms. Specifically, we recruited about 200 human subjects from Amazon Mechanical Turk and asked them to learn facts about COVID-19 through peer interaction. The experimental results corroborate two crucial hypotheses with statistical significance that are central to our formulation and proposed solutions. First, it validates that *workers' skills improve through peer interaction*. Second, it experimentally demonstrates that *changing group composition over time is important and that* DyGroups *outperforms baseline solutions*. Additionally, we run large scale synthetic data experiments and implement several baselines (along with recent related works [2], [8]) to validate the theoretical claims and scalability aspects of our proposed solutions.

This paper is organized as follows. Our model and problem formalization are discussed in Section II. Our algorithmic framework and its running time properties are discussed in III. Section IV contains theoretical proofs of our statements. Experiments using human subjects as well as large scale simulation studies are shown in Section V. The related work is introduced in Section VI. We present a discussion and outline future works in Section VII, and conclude in Section VIII.

II. Model & Definitions

We consider n participants who undergo a learning process in rounds, or time steps. Before step t each participant is associated with a positive real number, quantifying their *skill level*. In every round, k non-overlapping equi-sized groups are formed. Each participant interacts 1-on-1 only with members of their group. The learning outcome of a 2-person interaction is determined by the *learning gain function*. The total learning gain within a group is further determined by a specified *interaction mode*. After one round, skill levels are updated. The process continues inductively for α steps.

We now proceed to define the highlighted terms in the above description. We will be using the following example to illustrate the notions.

[Toy Example]. Imagine a small number of $n = 9$ students taking a course on Python Programming that comprises of $\alpha = 4$ assignments during the course of a semester. In every round we will be forming $k = 3$ disjoint groups of size 3. We assume that in the beginning of the course, the skills of the students in Python programming are as follows: $[0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9]$ [4].

Learning Gain Function for 2-Person Interactions. The learning outcome for a 2-person interaction between participants i, j depend only on $\Delta = |s_i - s_j|$. Specifically, if $s_i > s_j$, then after their interaction:

(i) s_i is unaltered, and **(ii)** s_j is updated to $s_j + f(\Delta)$

We call $f(\Delta)$ the learning gain function. In the rest of this paper we will work with linear functions $f(\Delta) = r\Delta$, where $r \in$(0,1) is a *learning rate* parameter that is part of the input [5].

As the learning function involves an asymmetry between the two parts, we let $f(i \rightarrow j)$ denote the skill gain of person j from their interaction with person i. Note that if the skill of i is lower than that of j, then $f(i \rightarrow j) = 0$.

In the Toy Example, with a learning rate or $r = 0.5$, a pairwise interaction between the 3-rd and the 9-th member with skills $0.3, 0.9$ respectively, s_9 remains unaltered at 0.9. On the other hand, s_3 becomes $0.3 + 0.5 \times (0.9 - 0.3) = 0.6$.

Interaction Modes, Group Learning Gain. In each round, participants are split into non-overlapping equi-sized groups. The size of the groups is fixed throughout the α rounds. Each participant has 2-person interactions with other participants from their group. The learning gain within the groups will be determined by the *interaction mode*, which will be the same for all groups, throughout the process. We consider two possible interactions modes:

(i) *Star Mode*. Every participant of the group learns from the highest-skilled member of the group, and skill levels are updated according to the learning gain function. Specifically if p_i is i^{th} highest-skill participant of a group x, the learning gain of group x is

$$g_{star}(x) = \sum_{p_j \neq p_1} f(p_1 \rightarrow p_j). \tag{1}$$

In the Toy Example, assume $[0.9, 0.5, 0.3]$ are assigned to a group and the interaction model is star model. In this case 0.9 is unaltered in the group, and $0.5, 0.3$ will both learn from 0.9 and are updated to $0.7, 0.6$ after the learning, assuming a learning rate $r = 0.5$. In this case the total learning gain of the group (soon to be defined formally) is 0.5.

(ii) *Clique Mode.* All possible pairwise interactions take place. Suppose that p_i is the participant with the i^{th} highest learning skill in a group x. That implies that p_i will learn and gain skill from $(i-1)$ persons. Then, we define the learning gain of a group x as follows:

$$g_{cliq}(x) = \sum_{p_i \in x} \frac{1}{i-1} \left(\sum_{p_j \neq p_i} f(p_j \rightarrow p_i) \right) \tag{2}$$

In plain words, the total gain for p_i is the average of its positive gains from 2-person interactions within its group. The

[4]The distribution of the initial skill values can be arbitrary. How to estimate numerical values for the initial skills of the individuals in a real situation is an orthogonal issue. In our experimental evaluation, we demonstrate a realistic way of estimating their initial skills.

[5]The case r=1 is relatively straightforward and we omit it.

averaging operation ensures that the order of skill levels is preserved within the group after the round, as it would be expected in practice.

In the TOY EXAMPLE, assume $[0.9, 0.5, 0.3]$ form a group. As before, 0.9 is unaltered. However, 0.3 learns not only from 0.9 but also from 0.5. Therefore, the new skill value for 0.3 after learning is $0.3+(0.5(0.5-0.3)+0.5(0.9-0.3))/2 = 0.5$. Since 0.5 only learns from 0.9, the new skill value of 0.5 is 0.7 as the same in the previous example. The overall group learning gain is 0.4.

Aggregated Learning Gain per Round. Given a grouping $\mathcal{G}_t$ of k groups at round t, the aggregated learning gain of the grouping under either mode is defined as:

$$\mathbf{LG}(\mathcal{G}_t) = \sum_{x=1}^{k} g(x). \tag{3}$$

Problem 1. Targeted Dynamic Grouping *(*TDG*): Given as input a set of n individuals and their skills, an integer k representing the number of groups, and an integer α representing the number of rounds, our goal is to compute a sequence of groupings $\mathcal{G}_1 \dots, \mathcal{G}_\alpha$ that maximizes the aggregated learning gain over α rounds:*

$$\max_{\{\mathcal{G}_1 \dots, \mathcal{G}_\alpha\}} \sum_{t=1}^{\alpha} \mathbf{LG}(\mathcal{G}_t)$$

III. ALGORITHMS AND RUNNING TIME

We start with the presentation of the generic algorithmic framework DYGROUPS. Then we instantiate it for the Star and Clique modes in Sections III-A and III-B respectively. To avoid disrupting the flow of ideas, we defer the formal statements and proofs for our claims to Section IV.

DYGROUPS is fairly simple; we take a greedy stride in solving the problem. Since the process of forming k groups is to be repeated over α rounds, in each round t, DYGROUPS calls subroutine DYGROUPS-LOCAL to form a grouping $\mathcal{G}_t$ of k groups so as to *locally* maximize $\mathbf{LG}(\mathcal{G}_t)$ at round t. The new skill values become part of the inputs for round $t+1$ and the process repeats. Algorithm 1 presents the pseudo-code.

Algorithm 1 DYGROUPS-MODE

input: Set S of n skills, where s_i is the skill of individual i, number of groups k, learning rate r, number of rounds α.
output: Collection of α groupings, each consisting of k equi-sized groups, where $\mathcal{G}^t$ is the grouping in round t
for $t = 1 : \alpha$ **do**
  $\mathcal{G}^t$ = DYGROUPS-MODE-LOCAL(S,k)
  S = UPDATE-SKILLS-MODE($\mathcal{G}^t$,S)
end for
return $\mathcal{G}_1, \mathcal{G}_2, \dots \mathcal{G}_\alpha$

Three remarks are in order:

a. DYGROUPS-MODE is a generic framework in the sense that it can be instantiated for different interaction modes, by instantiating the two subroutines it calls. We will see how to do that specifically for the Star and Clique modes.

b. The greedy approach is a reasonable choice in this framework. As we will see shortly, DYGROUPS-MODE-LOCAL can solve the round-local maximization problem very efficiently in both the Star and Clique modes. It thus lends to a highly scalable algorithm, both with respect to n and k. There is additional theoretical support to our choice. As we shall show in Section IV-C, DYGROUPS-STAR, does produce the optimal solution for $k = 2$.

c. The **running time** of routine DYGROUPS-MODE is clearly $O(\alpha(T_g + T_u))$, where T_g, T_u are the running times of DYGROUPS-MODE-LOCAL and UPDATE-SKILLS-MODE respectively.

We also note that storing a grouping G^t requires $\Omega(n)$ memory, and so each round of the algorithm requires $\Omega(n)$ time. This leads to the following lower bound on the running time of DYGROUPS-MODE.

Claim 1. DYGROUPS-MODE *requires* $\Omega(\alpha n)$ *time.*

A. DYGROUPS-STAR

We begin by noting that UPDATE-SKILLS-STAR has a very straightforward implementation. Each skill update takes $O(1)$ time, because each participant interacts only with the 'teacher' of their group. Thus the total running time is $O(n)$.

On the other had, designing DYGROUPS-STAR-LOCAL is a very interesting problem. Let us call the highest-skilled person in some group, the *teacher* of that group. We first observe that the learning gain is maximized if the teachers of the k groups are selected to be the k highest-skilled participants, and that is true *irrespectively* of the split of the remaining $n-k$ participants into the k groups. This is due to the linearity of the learning function. The proof of this claim appears in Section IV, Theorem 1.

We thus have to select from exponential number of locally optimal groupings in each round [6]. Our further insight is to select the grouping $\mathcal{G}$ that has the *maximum variance* of skill values among the locally optimal groupings. This is done as follows. Suppose $p_1, \dots, p_k$ are the k teachers. Recall that we have to assign the other $n - k$ participants to a teacher. In order to do that we split them into k provisional groups of size $s = n/k - 1$, solely based on their skill level: each person in provisional group i has skill equal or higher relative to every person in group $i + 1$. Then we form group i, by assigning the i^{th} provisional group to teacher p_i. The proof that this assignment maximizes variance appears in Section IV, Theorem 2. Algorithm 2 gives an implementation.

The **running time** of DYGROUPS-STAR-LOCAL is dominated by $O(n \log n)$ for the sorting step. It is easy to see that the remaining lines take $O(n)$ time. Thus, the overall running time of DYGROUPS-STAR is $O(\alpha n \log n)$, which notably is independent of k.

We now illustrate our discussion using the TOY EXAMPLE from Section II. We first follow a sequence of three groupings

[6]e.g. when $k = n/2$ there are $(n/2)!$ locally optimal solutions, and when $k = 2$ there are $\binom{n-2}{(n-2)/2}$ locally optimal solutions.

Algorithm 2 DYGROUPS-STAR-LOCAL

input: Set S of n skills, where s_i is the skill of individual i, number of groups k.
output: Grouping $\mathcal{G}$ consisting of k groups of size n/k.
X = sort(S,descending) // sorted skill values
Let p_i be the participant with skill $X[i]$
$t = k + 1; s = n/k - 1;$
for $i = 1 : k$ **do**
  Assign 'teacher' p_i to group g_i
  for j=1:s **do**
    Assign p_t to group g_i
    t = t+1
  end for
end for
return $\mathcal{G} = \{g_1, g_2, ..., g_k\}$

that selects an arbitrary locally optimal grouping in each round.

Round 1: $[0.9, 0.1, 0.2], [0.8, 0.3, 0.4], [0.7, 0.5, 0.6]$.
Updated Skills: $[0.9, 0.8, 0.7, 0.65, 0.6, 0.6, 0.55, 0.55, 0.5]$.
Round 2: $[0.9, 0.55, 0.5], [0.8, 0.6, 0.55], [0.7, 0.65, 0.6]$
Updated Skills: $[0.9, 0.8, 0.7, 0.675, 0.65, 0.7, 0.675, 0.725, 0.7]$
Round 3: $[0.9, 0.675, 0.65], [0.8, 0.7, 0.675], [0.725, 0.7, 0.7]$
Final: $[0.9, 0.8, 0.725, 0.7125, 0.7125, 0.75, 0.7375, 0.7875, 0.775]$
The total learning gain after 3 rounds is 2.4.

Let us now introduce how DYGROUPS-STAR runs.
Round 1: $[0.9, 0.6, 0.5], [0.8, 0.4, 3], [0.7, 0.2, 0.1]$
Updated Skills: $[0.9, 0.8, 0.7, 0.75, 0.7, 0.6, 0.55, 0.45, 0.4]$
Round 2: $[0.9, 0.7, 0.7], [0.8, 0.6, 0.55], [0.75, 0.45, 0.4]$
Updated Skills:$[0.9, 0.8, 0.75, 0.8, 0.8, 0.7, 0.675, 0.6, 0.575]$
Round 3: $[0.9, 0.8, 0.75], [0.8, 0.7, 0.675], [0.8, 0.6, 0.575]$
Final: $[0.9, 0.8, 0.8, 0.85, 0.825, 0.75, 0.7375, 0.70, 0.6875]$
The total learning gain after 3 rounds is 2.55.

While we do not know if DYGROUPS-STAR will in general produce the optimal grouping sequence, we do present a proof of this fact for $k = 2$, in Section IV-C . We note that forming two groups is natural in applications, such as peer programming, where one group does the programming and the other peer reviews.

Finally we would like to shortly discuss the insight that leads to the proof for $k = 2$. A further inspection of the above example can reveal that the two different sequences of groupings produce the same learning gain after the first 2 rounds. However the variance maximization policy leads to a *higher* 3^{rd}*-order teacher* in round 3. This availability of better teachers earlier in the process is what leads to DYGROUPS-STAR dominating other solutions.

B. DYGROUPS-CLIQUE

Let us begin with UPDATE-SKILLS-CLIQUE. By design, in the Clique mode, every person learns from all the higher-skilled persons in the group and so there are $O(t^2)$ interactions, where $t = n/k$ is the size of the group. However it is possible to calculate all the updated skills within the group in $O(t)$ time, leading to an $O(n)$ update algorithm. We prove this fact in Section IV, Theorem 3.

Similarly to DYGROUPS-STAR, DYGROUPS-CLIQUE finds a grouping that maximizes the gain for each round. This is formally stated in Theorem 3. The Clique mode definition leads us to a different grouping algorithm, which computes the *unique* grouping $\mathcal{G} = \{g_1, \ldots, g_k\}$ with the property that the j^{th}-ordered skill in g_i is greater than or equal to the j^{th}-ordered skill in g_{i+1} for each i, j. Algorithm 3 provides an implementation.

Algorithm 3 DYGROUPS-CLIQUE-LOCAL

input: Set S of n skills, where s_i is the skill of individual i, number of groups k.
output: Grouping $\mathcal{G}$ consisting of k groups of size n/k.
X = sort(S,descending) // sorted skill values
Let p_i be the participant with skill $X[i]$
$t = 1; s = n/k;$
for $j = 1 : s$ **do**
  for $i = 1 : k$ **do**
    Assign p_t to group g_i
    t = t+1
  end for
end for
return $\mathcal{G} = \{g_1, g_2, ..., g_k\}$

The **running time** of DYGROUPS-STAR-CLIQUE is dominated by $O(n \log n)$ for the sorting step. It is easy to see that the remaining lines take $O(n)$ time. Thus, the overall running time of DYGROUPS-CLIQUE is $O(\alpha n \log n)$, which notably is independent of k.

Let us illustrate the algorithm with TOY EXAMPLE.
Round 1: $[0.9, 0.6, 0.3], [0.8, 0.5, 0.2], [0.7, 0.4, 0.1]$
Updated: $[0.9, 0.8, 0.75, 0.7, 0.65, 0.55, 0.525, 0.425, 0.325]$
Round 2: $[0.9, 0.7, 0.525], [0.8, 0.65, 0.425], [0.75, 0.55, 0.325]$
Updated: $[0.9, 0.8, 0.8, 0.75, 0.725, 0.6625, 0.65, 0.575, 0.4875]$
Round 3: $[0.9, 0.75, 0.65], [0.8, 0.725, 0.575], [0.8, 0.6625, 0.4875]$
Final: $[0.9, 0.825, 0.8, 0.8, 0.7625, 0.7375, 0.73125, 0.66875, 0.609375]$
The total learning gain after 3 rounds is 2.334375.

IV. PROOFS

All formal statements in this Section refer back to the informal discussion in Section III.

A. DyGroups-Star

Theorem 1. *Suppose $S = [s_1, \ldots, s_n]$ is a set of skills in descending order. Then: (a) An optimal grouping into k equi-sized groups, i.e. a grouping that maximizes the learning gain, must assign $s_1, \ldots, s_k$ to different groups. (b) Every grouping that assigns $s_1, \ldots, s_k$ is optimal.*

Proof. Given a group g let $s_{g,i}$ denote the i^{th} highest skill in g. Also, let $t = n/k$ denote the size of the groups. (a) For the sake of contradiction, suppose that $\mathcal{G}$ is optimal and that it does not assign $s_1, \ldots, s_k$ to the same group. That implies that there are two groups $g, g' \in \mathcal{G}$ such that $s_{g,2} > s_{g',1}$. Let

C_g be the cumulative value of the $t-2$ smallest skills in g, and let C'_g be the cumulative value of the $t-1$ smallest skills in g'. The total gain A in the given grouping is given by:

$$r\left[(s_{g,1}-s_{g,2})+((t-2)s_{g,1}-C_g)+((t-1)s_{g',1}-C_{g'})\right]$$

We can now swap the two skills g_2, g'_1 between g and g'. In this new grouping, the new total gain B is

$$r\left[(s_{g,1}-s_{g',1})+((t-2)s_{g,1}-C_g)+((t-1)s_{g,2}-C_{g'})\right]$$

The statement is meaningful when $t>2$. Given that $s_{g,2}>s_{g',1}$, $t>1$, we get that $B>A$, which is a contradiction.
(b) Suppose now that $\mathcal{G}$ is an arbitrary optimal grouping. Let g, g' be two arbitrary groups in G, and let $s_{g,i}, s_{g',j}$ two skills from g, g' different than the top skills (i.e. $i,j \neq 1$). Swapping $s_{g,i}$ and $s_{g',j}$ between the two groups leaves the total gain invariant, as follows from a simple application of the definition. Now if $\mathcal{G}$ and $\mathcal{G}'$ are two different arbitrary optimal groupings, then they must agree on the leaders of the k groups, by part (a). One can then apply a sequence of swaps such as the one described above, to transform $\mathcal{G}$ into $\mathcal{G}'$, while leaving the gain invariant along the sequence. Therefore the total gain of $\mathcal{G}$ and $\mathcal{G}'$ must be the same. □

Theorem 2. *The output $\mathcal{G}$ of* DyGroups-Star-Local *is the grouping of highest variance among the groupings that maximize the learning gain on input set S.*

Sketch. For the sake of contradiction suppose $\mathcal{G}'$ is another optimal grouping. Let $\mathcal{G}'=\{g'_1,\dots,g'_k\}$. By Theorem 1, since $\mathcal{G}'$ is optimal, the highest skills $s_1\geq\ldots\geq s_k$ of its groups are fixed, and they are the same in the corresponding groups of $\mathcal{G}$. Since $\mathcal{G}\neq\mathcal{G}'$, there must be some $i>j$ such that g'_i contains skill s', and g'_j contains skill s, with $s<s'$.

Now let $\mathcal{G}''$ be the grouping after swapping s' and s in $\mathcal{G}'$. By Theorem 1, the mean skill μ is the same in $\mathcal{G}'$ and $\mathcal{G}''$ after the update. The variance after the update in $\mathcal{G}'$ is:

$$A=C+(s+r(s_i-s))-\mu)^2+(s'+r(s_j-s'))-\mu)^2$$

where C is a sum of squares, each coming from one of the other $n-2$ updated skill values. Similarly the variance after the update in $\mathcal{G}''$ is:

$$B=C+(s'+r(s_i-s'))-\mu)^2+(s+r(s_j-s))-\mu)^2$$

We can now calculate $B-A$, using the facts that $s_i>s_j$ and $s<s'$, and get $B>A$. This is a contradiction. □

B. DyGroups-Clique

Theorem 3. Update-Skills-Clique *can be implemented in $O(n)$ time.*

Proof. Let $t=n/k$ be the size of one group. We fix an arbitrary group, and assume that the initial set of descending-sorted skills is $s_1,\dots,s_t$. Now let $c_i=\sum_{j=1}^{i}s_j$. Clearly each c_{i+1} can be computed from c_i with one addition. Hence computing all c_i's takes $O(t)$ time. Let $s'_1,\dots,s'_t$ be the updated set of skills. Clearly, we have $s'_1=s_1$ since the highest-skilled person does not learn from anyone in the group. Also, for $i>1$, using our definitions from Section II we have

$$s'_{i+1}=s_i+r(c_i-is_{i+1})/i$$

Thus, having computed the c_i's, the calculation of all new skills requires $O(t)$ time. Doing that for k groups gives a total running time of $O(n)$. □

Theorem 4. DyGroups-Local-Clique *produces a grouping that maximizes the gain for the input set of skills.*

The proof follows a similar reasoning with that of the proof of Theorem 1. However the calculations are rather lengthy, and so we omit it due to space constraints.

C. DyGroups-Star For Two Groups

Terminology: when we say a grouping is *locally-optimal* we mean that maximizes the gain for that round only, but possibly not for the entire process. If it is not locally optimal, then we say it is *non-locally optimal*.

Theorem 5. DyGroups-Star *is optimal for the TDG problem when $k=2$.*

The proof is rather non-trivial and is presented in several steps. We first present an equivalent yet alternative objective function of the TDG problem that is pivotal for the proof. We then present a set of helper lemmas that are crucial and finally prove Theorem 6.

An equivalent objective function: Let $s_1^0,\dots,s_n^0$ be the input skill values in decreasing order. The TDG objective is:

$$\text{Maximize}\sum_{t=1}^{\alpha}\mathbf{LG}(\mathcal{G}_t)$$

Assume the skill value of a member i after α rounds is s_i^α, and s_i^0 is the initial skill value of i. Then, the objective function can be written as:

$$\text{Maximize}\sum_{i=1}^{n}s_i^\alpha-s_i^0$$

We thus convert the input to the distance to the highest-skilled member: $0, b_2^0, b_3^0\dots, b_n^0$, where $b_i^0=s_1^0-s_i^0$. In the Toy Example the $s_1,..,s_9$ are $[0.9,0.8,0.7,0.6,0.5,0.4,0.3,0.2,0.1]$. Therefore, $b_1,\dots,b_9$ are $[0,0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8]$.

Therefore, the equivalent objective function of the TDG problem is

$$\text{Minimize}\ \sum_{i=1}^{n}b_i^\alpha \tag{4}$$

Assume $D=\sum_{i=1}^{n}b_i^0$. Since the most skilled member will always lead a group, we can assume b_x^0 is the teacher in the other group. In each group, there are $\frac{n}{2}-1$ members learning from the leader. Assume i is one of them and x is the leader in i's group. The skill value s_i^1 after one round will be:

$a_i^0 + r(a_x^0 - a_i^0)$. So, the skill distance b_i^1 after this round will be:

$$\begin{aligned} s_1 - s_i^1 = & \\ & = s_1 - s_i^0 - r(s_x^0 - s_i^0) \\ & = b_i^0 - r(b_i^0 - b_x^0) \\ & = (1-r)b_i^0 + rb_x^0 \end{aligned}$$

Therefore, the aggregated skill distance after the first round will be:

$$\begin{aligned} & \sum_{\substack{i \in [2,\ldots,n] \\ i \neq x}} (1-r)b_i^0 + (\frac{n}{2}-1)rb_x^0 + b_x^0 \\ = & \sum_{\substack{i \in [2,\ldots,n] \\ i \neq x}} (1-r)b_i^0 + (\frac{n}{2}-1)rb_x^0 + (b_x^0 - rb_x^0) + rb_x^0 \\ = & \sum_{i \in [2,\ldots,n]} (1-r)b_i^0 + \frac{n}{2}rb_x^0 \\ = & (1-r)D + \frac{n}{2}rb_x^0 \end{aligned}$$

By this, we can assume b_x^i is the teacher of the second group for the round i ($i < \alpha$) and adapt the equation above recursively. According to this, we can re-write the objective function as:

$$\text{Minimize } \frac{n}{2}r\sum_{i=1}^{\alpha} b_x^i(1-r)^{\alpha-i} + D(1-r)^{\alpha} \tag{5}$$

Since the second part is constant, the problem becomes to find a series of groupings which maximize the skill value of the second teacher in the group. For the purpose of the proof, we will stick to this alternative objective function for the remainder of this subsection.

Lemma 1. *There exist $2\binom{n-2}{\frac{n}{2}-1}$ local optima in each round.*

Proof. According to the *Star* mode, the teacher of group-1 is the highest skilled individual (s_1) and the teacher of group-2 is the second highest skilled individual (with skill s_2). The assignment of the remaining individuals inside these two groups does not interfere with the learning gain function. Since there are such $2\binom{n-2}{\frac{n}{2}-1}$ possible assignments. Therefore, we have $2\binom{n-2}{\frac{n}{2}-1}$ local optimals in each round. □

Lemma 2. DYGROUPS-STAR *in conjunction with* DYGROUPS-STAR-LOCAL *is not worse than any other solutions that produce local optima in each round.*

Proof. Assume X is the series of groupings that are produced by DYGROUPS-STAR with DYGROUPS-STAR-LOCAL and there is another solution X' that also contains local maximum groupings but different from X. Therefore, in X', there must be at least one grouping that is not present in X. For a local maximum solution, the teacher of the second group is the second most skilled person. According to the objective function, if X' is a better solution than X, X' should have higher second skilled person at some rounds.

Let's assume that such a scenario occurs in round t.This entails individual i attains higher skill value in X' than in X and this skill value exceeds the second highest skill value in round $t-1$. It means either i achieves higher learning gain in X' than X, or i has higher skill value in X' before $t-1$.

In DYGROUPS-STAR-LOCAL, the 3rd to $\frac{n}{2}$ individuals are placed in the first group, which ensures the largest skill increase. Therefore, the aforementioned scenario can not happen. Thus, DYGROUPS-STAR in conjunction with DYGROUPS-STAR-LOCAL will not produce any worse objective function than any other local maximum solution. □

Theorem 6. DYGROUPS-STAR *is not worse than any other solution that contains non-local optima.*

Proof. The overall proof consists of multiple steps. First, we assume a case when the alternative solution with objective value X' contains only one non-local optima. Then, we extend the proof to the case when the alternative solution contains multiple (y) local solutions that are not *local optima*.

(Only one non-local optima:) First, let's assume there is a solution X' that contains only one *non-local optimum grouping at round t ($t < \alpha$)*. Other than that, DYGROUPS-STAR and this other solution both produce local optima in the first $t-1$ rounds. Let $b_i^{t'}$ be the skill difference of the i-the member in the t-th round (recall Equation 4). Let $b_x^{t'}(x > 2)$ be the skill difference of the second teacher. So, the objective function value of X' will be:

$$\begin{aligned} D(1-r)^{\alpha} + r\frac{n}{2}(\sum_{i=1}^{t-1} b_2^{i'}(1-r)^{\alpha-i} + b_x^{t'}(1-r)^{\alpha-t} \\ + b_2^{t+1'}(1-r)^{\alpha-t-1} + \sum_{i=t+2}^{\alpha} b_2^{i'}(1-r)^{\alpha-i}) \end{aligned}$$

For our solution X, the objective value is:

$$\begin{aligned} D(1-r)^{\alpha} + r\frac{n}{2}(\sum_{i=1}^{t-1} b_2^{i}(1-r)^{\alpha-i} + b_2^{t}(1-r)^{\alpha-t} \\ + b_2^{t+1}(1-r)^{\alpha-t-1} + \sum_{i=t+2}^{\alpha} b_2^{i}(1-r)^{\alpha-i}) \end{aligned}$$

Since $b_2^{t'} \leq b_x^{t'}$, $b_2^{t'}$ is placed in the first group t. Therefore, $b_2^{t'}$ becomes to $(1-r)b_2^{t'}$ after round t. Noted that, $b_2^{t'} \leq b_i^{t'}, i \geq 3$, so $(1-r)b_2^{t-1'} \leq (1-r)b_i^{t-1'}, i \geq 3$. That means $(1-r)b_2^{t-1'}$ is the teacher of the second group($b_2^{t+1'}$) of X' at round $t+1$.

The minimum skill difference (that is the b value where smaller is better) that the second teacher can attain is $b_2^{i'}, i < t$ of X' is b_2^i. So, we can rewrite the objective value of X' as:

$$\begin{aligned} D(1-r)^{\alpha} + r\frac{n}{2}(\sum_{i=0}^{t-1} b_2^{i}(1-r)^{\alpha-i} + b_x^{t}(1-r)^{\alpha-t} \\ + b_2^{t}(1-r)^{\alpha-t} + \sum_{i=t+2}^{\alpha} b_2^{i'}(1-r)^{\alpha-i}) \end{aligned}$$

Then, there are two possible cases: **(Case 1:)** b_2^t stays as the second teacher in X from t to α; **(Case 2:)** A new second teacher shows up in round u ($u > t$) with $b_3^t(1-r)^{u-t}$.

(Case 1:) The objective value of X is:

$$D(1-r)^{\alpha} + r\frac{n}{2}(\sum_{i=1}^{t-1} b_2^i(1-r)^{\alpha-i} + b_2^t(1-r)^{\alpha-t} + b_2^t(1-r)^{\alpha-t-1} + \sum_{i=t+2}^{\alpha} b_2^{t-1}(1-r)^{\alpha-i}) \tag{6}$$

The objective value for X' is:

$$D(1-r)^{\alpha} + r\frac{n}{2}(\sum_{i=1}^{t-1} b_2^i(1-r)^{\alpha-i} + b_x^t(1-r)^{\alpha-t} + b_2^t(1-r)^{\alpha-t} + \sum_{i=t+2}^{\alpha} b_2^t(1-r)^{\alpha-i+1}) \tag{7}$$

So, Equation 6 - Equation 7 is:

$$r\frac{n}{2}(b_2^t - b_x^t(1-r)^{\alpha-t})$$

Since there is no position change in the second teacher, $b_2^t \leq b_x^t(1-r)^{\alpha-t}$. Therefore, Equation 6 - Equation 7 ≤ 0 and proved.

(Case 2:): In this case, $(1-r)^{u-t}b_3^t \leq b_2^t$. In addition, since there is no change in the second teacher before round u, $b_2^t \leq (1-r)^{u-t-1}b_i^t, i \geq 3$.
Therefore, $(1-r)b_2^t \leq (1-r)^{u-t}b_i^t, i \geq 3$. Overall, the objective function value of X is:

$$\begin{aligned}
&D(1-r)^{\alpha} + r\frac{n}{2}(\sum_{i=0}^{t-1} b_2^i(1-r)^{\alpha-i} + b_2^t(1-r)^{\alpha-t} \\
&+ \ldots + b_2^t(1-r)^{\alpha-u+1} + (1-r)^{u-t}b_3^t(1-r)^{\alpha-u} \\
&+ \min(b_4^t(1-r)^{u-t+1}, b_2^t(1-r))(1-r)^{\alpha-u-1} \\
&+ \sum_{i=u+2}^{\alpha} b_2^i(1-r)^{(\alpha-i)})
\end{aligned}$$

Similarly, the objective value of X' needs to be discussed considering two cases:

The third highest skilled individual is the teacher of the second group:

$$\begin{aligned}
&D(1-r)^{\alpha} + r\frac{n}{2}(\sum_{i=0}^{t-1} b_2^i(1-r)^{\alpha-i} + b_3^t(1-r)^{\alpha-t} \\
&+ b_2^t(1-r)^{\alpha-t} + \ldots + b_2^t(1-r)^{\alpha-u+1} \\
&+ \min(b_4^t(1-r)^{u-t+1}, b_2^t(1-r))(1-r)^{\alpha-u-1} \\
&+ \sum_{i=u+2}^{\alpha} b_2^{i'}(1-r)^{\alpha-i})
\end{aligned}$$

We can observe that since X' has the same second skill value as X at the round $u+1$, and X' adopts the local maximum groupings after round t, so X and X' has the same objective value in this case.

Any other individual with b_x^t is the teacher of the second group:

$$\begin{aligned}
&D(1-r)^{\alpha} + r\frac{n}{2}(\sum_{i=0}^{t-1} b_2^i(1-r)^{\alpha-i} + b_x^t(1-r)^{\alpha-t} \\
&+ b_2^t(1-r)^{\alpha-t} + ... + b_2^t(1-r)^{\alpha-u+1} \\
&+ min(b_3^t(1-r)^{u-t+1}, b_2^t(1-r))(1-r)^{\alpha-u-1} \\
&+ \sum_{i=u+2}^{\alpha} b_2^{i'}(1-r)^{\alpha-i})
\end{aligned}$$

As before, X' is not better than X after u. In fact, it could be easily shown, X' gets worse (with higher value of the objective function in Equation 4), as we consider any other individuals with larger b_x^t as the teacher.

Overall, the optimal solution that only contains one round of non-local maximum grouping cannot achieve better objective value than ours. We can also observe two facts, for round $u > t$:

1. When $b_2^{u'} > b_2^u$, the objective value of X' is the same or worse than X (Case 1, no swap after the round t).

2. When $b_2^{u'} = b_2^u$, the objective values of two methods are equal (Case 2 after the swap round).

Multiple y one non-local optimas: Next we prove the case when there exists y non-local optimas in X'. Before the second non-local maximum grouping and after the the first non-local maximum grouping, the aforementioned two facts hold.

Assume the i-th non-local maximum grouping is adopted by X' at the round t_i. When $b_2^{v'} = b_2^v(t_1 < v < t_2)$, it is the exact same as the previous discussions: X' is not better than X before the round u_2. According to the algorithm, it always put the $3rd$ to $\frac{n}{2}$ members into the first group. Therefore, according to the lemma 2, this property will hold until the next non-local maximum grouping happens in X'.

Assume the second leader of X' at the round t_i is $b_{x_i}^{t_i}$, and X changes the second leader at the round z, so $b_2^{v'} > b_2^v(t_1 \leq v \leq z)$. There are y non-local maximum groupings in X' before the round z. Then, the objective value of X' will be:

$$\begin{aligned}
&D(1-r)^{\alpha} + r\frac{n}{2}(\sum_{i=0}^{t_1-1} b_2^i(1-r)^{\alpha-i} \\
&+ b_{x_1}^{t_1}(1-r)^{\alpha-t_1} + (1-r)b_2^{t_1}(1-r)^{\alpha-t_1-1} \\
&+ \ldots + (1-r)b_2^{t_1}(1-r)^{\alpha-t_2+1} \\
&+ b_{x_2}^{t_2}(1-r)^{\alpha-t_2} + \cdots + b_2^{\alpha'})
\end{aligned} \tag{8}$$

And the objective value of X is:

$$\begin{aligned}
&D(1-r)^{\alpha} + r\frac{n}{2}(\sum_{i=0}^{t_1-1} b_2^i(1-r)^{\alpha-i} \\
&+ b_2^{t_1}(1-r)^{\alpha-t_1} + b_2^{t_1}(1-r)^{\alpha-t_1-1} \\
&+ ... + b_2^{t_1}(1-r)^{\alpha-t_2+1} \\
&+ b_2^{t_2}(1-r)^{\alpha-t_2} + \cdots + b_2^{\alpha})
\end{aligned} \tag{9}$$

Therefore, Equation 9 - Equation 8 is:

$$\sum_{i=0}^{y-1} b_2^{t_1}(1-r)^{\alpha-z+i} - \sum_{i=1}^{y} b_{x_i}^{t_i}(1-r)^{\alpha-t_i} \tag{10}$$

Since our algorithm always pick the second skilled person as the leader of the second group, $b_2^{t_1} \leq b_{x_1}^{t_1}(1-r)^{t_2-t_1-1}$, and $b_2^{t_1} \leq b_{x_2}^{t_2}$, and so on.. Therefore, Equation 10 is smaller or equal to zero, and X is not worse than X'. Overall, X is not worse than any optimal solution that contains any non-local maximum groupings. Hence, the proof. □

Proof. (of Thorem 5) With Lemmas 1 and 2, we prove that although there exists many local optima, DYGROUPS-STAR in conjunction with DYGROUPS-STAR-LOCAL is not worse than any other local optima solutions considering the objective function in Equation 5. From Theorem 6, we prove that DYGROUPS-STAR is not worse than any solution that does not produce local optima. Combining these three, we therefore prove that DYGROUPS-STAR produces global optima for the TDG problem. □

V. EXPERIMENTAL EVALUATION

We now present our experimental evaluation of DYGROUPS. We perform two experiments with human subjects and then we demonstrate the quantitative and runtime performance of DYGROUPS on synthetic data.

A. Human Subjects Experiments

The main purpose of this study is to experimentally examine: a. The effectiveness of peer learning, i.e. whether individual skills improve through interactions with peers. b. The effectiveness of DYGROUPS, relative to baseline solutions.

We consider an application on learning facts from peers through targeted dynamic groups formation.We present two experiments with human subjects. They are similar but they have been conducted independently; also the second experiment is more extensive. The experiments employ real workers hired on Amazon Mechanical Turk (AMT) to learn facts related to COVID-19 from their peers.

AMT Setup: We deploy multiple-choice questions as a Human Intelligent Task (HIT) consisting of facts and rumors about COVID-19. Those are shown to the workers as tasks[7]. Workers are paid \$5 if they stick with the entire learning process. Each deployment was accessible for 24 hours and 1 hour is allotted to each worker.

Skill Assessment: Each HIT consists of 10 questions. The HIT questions also comprise a skill assessment test; the skill of each participant is set to be equal to the number of their correct answers, divided by 10.

[7]
Sample questions:
- What is the longest incubation time of COVID-19 in the record?
A. 14 days, B. 19 days, C. 20 days, D. More than 20 days

- Which action will help to prevent COVID-19?
A. Wash your hands regularly and thoroughly
B. Taking a hot bath, C. Drinking alcohol, D. None of the above

Experiment-1. We recruit $N = 64$ individuals. They first undergo PRE-QUALIFICATION: Workers are assigned individual assessment tests to estimate their skill level. Based on the qualification outcome, we split them into two Populations A,B each containing $n = 32$ participants. The split is random, under the constraint that the two populations have very similar skill distributions, and in particular the same average skill. Population A follows DYGROUPS with a learning rate of $r = 0.5$ and $k = 4$, while Population B follows a baseline solution KMEANS (details in Section V-B). During the learning process, each Population alternates between these two steps: [GROUP-FORMATION]: Worker groups are formed, following the respective policy. The workers are asked to answer the questions collaboratively, by consulting with the rest of their peers in their group. [POST-ASSESSMENT]: Another test, akin to PRE-QUALIFICATION is performed to estimate the new skill of the individuals - and compute the learning gain after each round. The experiment consists of $\alpha = 3$ rounds.

Experiment-2. This is identical to Experiment-1 except it is conducted using N=128, that are split into four Populations of size $n = 32$ following DYGROUPS and the baselines KMEANS, LPA and PERCENTILE-PARTITIONS, further discussed in Section V-B. The experiment consists of $\alpha = 2$ rounds.

Parameter justification: The choice of parameters r=0.5 and k=4 in our experiments is not arbitrary. Before running the actual experiment, we have made several initial deployments, where we hired workers of varying expertise from AMT and formed random groups of different size: small groups of size $2, 3, 4, 5$, and large groups of size $10, 12, 15$, and let them interact across multiple rounds. We have conducted pre-assessment and post-assessment tests on these deployments that steered us in the choice of parameters. From these initial deployments we have learned that for these assigned fact checking tasks, the new skill that individuals acquire after interaction with another higher skilled peer *is on average half of the difference of skills between between them prior to interaction*. We also found that groups are most interactive and manageable when they contain $4-5$ people, and that worker engagement tends to dissipate if works are asked to participate in too many rounds. This leads us to set $r = 0.5$, k=4, and α=2 or 3 for the actual experiments. We also have observed that the one day time window is good enough for each round, and the workers do not need to spend more than one hour overall.

Summary of Results.

- *Observation I.* The aggregated skill improves with peer interaction (75% confidence interval), i.e. peer learning is effective. This can be seen in Figures 1 and 4(a) that show that the learning gain of PRE-QUALIFICATION and POST-ASSESSMENT scores after each round.
- *Observation II.* DYGROUPS outperforms the baselines with statistical significance (Figures 4(a) and 1). Interestingly, DYGROUPS outperforms even after the first round, which shows that it is very competitive even as a single-shot group formation algorithm.

We also note two serendipitous features of DYGROUPS that are worth of further investigation:

- *Observation III.* DYGROUPS has higher worker retention than other baselines (Figures 3, 4(b)). This anecdotally indicates that under the same monetary rewards, the rate of skill improvement may be an important factor towards retaining participants in the process.
- *Observation IV.* As the amount of total skill left to be learned decreases with α, we expect that the aggregate learning must have a negative second derivative. However, in Figure 2, aggregated learning gain appears to increase linearly in the first rounds of DYGROUPS. This indicates that the learning rate may accelerate during the first rounds.

B. Synthetic Data Experiments

1) Experimental Setup: Experiments are implemented in C++ and performed on a machine with Intel i5 CPU and 4GB Memory. In experiments involving randomness, we average over 10 different runs. Due to space constraints, we present a representative subset of our results.

Baseline Algorithms. We note that there are no prior works on the dynamic groups formation problem. The closest related works are [2], [8], both of which focus on the one-shot grouping problem. We thus design a range of baseline algorithms, each employing a different grouping scheme applied for α rounds:

- RANDOM-ASSIGNMENT. Groups are selected randomly.
- PERCENTILE-PARTITIONS. Groups are computed using an algorithm from [8]. The algorithm involves a parameter p, which is set to 0.75, following the discussion in [8].
- LPA. This uses an algorithm from [2].
- K-MEANS. This is an alternative grouping heuristic that we devise as a baseline. The algorithm picks k random participants as group 'centers' and assigns the rest to their nearest group, that is not completely full.

We also implement BRUTE-FORCE, an exponential-time algorithm that solves the TGD problem optimally. Naturally, the algorithm can be run only for very small values of n and k, and α.

Parameters. We vary the following seven parameters: number of partipants:(n), number of groups:(k), number of rounds:(α), interaction mode:(star/clique), distribution of the initial skill values, learning rate:(r).

Distribution. We generate the initial skill values of people using the log-normal and Zipf distributions. Both are guaranteed to produce positive skill values (unlike the normal distribution). We set the mean $\mu = e$ and the standard deviation $\sigma = \sqrt{e}$ for the log-normal distribution. On the other hand, the shape parameters of Zipf distribution are set to 2.3 and 10.

Summary of results. In the immediately subsequent sections, we can see the following.

- *Section V-B2*. DYGROUPS is superior compared to other baselines in terms of improving the aggregated learning gain, under a range of parameter settings.
- *Section V-B3*. BRUTE-FORCE matches DYGROUPS-STAR for $k = 2$ and small values of α, n as predicted by our theoretical results.
- *Section V-B4*. DYGROUPS methods induces significant learning gains relative to RANDOM-ASSIGNMENT.
- *Section V-B5*. DYGROUPS allows higher 'inequality' among participants relative to RANDOM-ASSIGNMENT.
- *Section V-B6*. DYGROUPS is highly scalable and therefore suitable for large scale real world applications.

2) Effectiveness Experiments: We review experiments on the effectiveness of DYGROUPS.

Default Parameters. Unless otherwise noted, $k = 5$, $n = 10000$, $\epsilon = 0.05$, $r = 0.5$, $\alpha = 5$, *star* mode, with log-normally distributed initial skills. Deciding appropriate values for these parameters is often times application dependent. In our synthetic data experiments, these default values are decided based on the outcome of our real data experiments.

Varying n-[Figures 5(a,b)]. We record the aggregate learning gain **LG** as a function of n, for both initial skill distributions. The results demonstrate that aggregate learning gain increases with increasing n. DYGROUPS convincingly outperforms all other baselines.

Varying k-[Figures 6(a,b)]. We record the learning gain as a function of the group size k. DYGROUPS outperforms other baselines. We also notice that **LG** decreases with increasing k. This is expected since with a higher number of groups, not all groups get to have expert peers and therefore the learning gain decreases.

Varying α-[Figures 7(a,b)]. As before, DYGROUPS convincingly wins. As expected, a higher α induces a higher aggregate learning gain.

Varying r-[Figures 8-9]. We record the aggregate learning gain as a function of the learning rate r. We can observe that DYGROUPS outperforms in the clique model for all of r values. In the special case of $r = 1$, by definition of the star mode, it takes $\log_{n/k}(n)$ rounds to make everyone reach the highest skill value for DYGROUPS and LPA.

3) Star Interaction Mode with $k = 2$: We experimentally validate our theoretical claim presented in Section IV-C. We compare BRUTE FORCE with DYGROUPS-STAR, where we set $\alpha \in [1, 4]$, $n \in \{4, 6, 8\}$ and the skill values are picked from $[0, 1]$ uniformly. We run 1000 different experiments with the parameters picked at random. In all of them DYGROUPS-STAR agrees with BRUTE-FORCE, i.e. it maximizes the aggregate learning gain.

4) Learning Gain Relative to Random Groupings: In Figure 10 we plot the ratio of the learning gain of DYGROUPS methods vs that of the RANDOM-ASSIGNMENT, as a function of α and n. Specifically, for a fixed $n = 10000$ we let α range over $\{2, 4, 6, 8, 16, 32, 64\}$. For a fixed $\alpha = 10$ we let n range over $\{10, 10^2, 10^3, 10^4, 10^5, 10^6\}$. It can be seen that DYGROUPS achieve up to 30% higher learning gain relative to random groupings over a small number of rounds. Interestingly, DYGROUPS-STAR is comparable to DYGROUPS-CLIQUE under this special case, and thus the simpler star mode may be a good proxy for the clique mode.

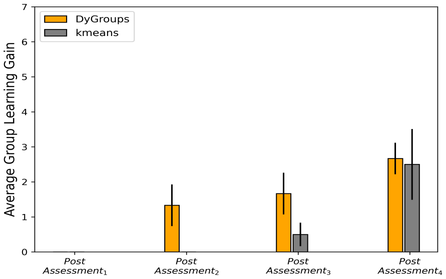


Fig. 1: Experiment-1: Learning gain across rounds

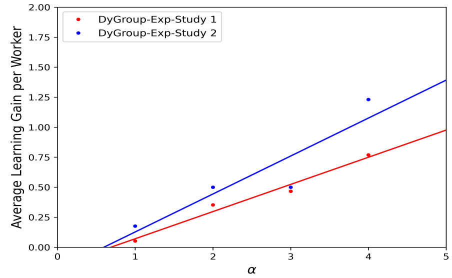


Fig. 2: Linear fit to learning gain

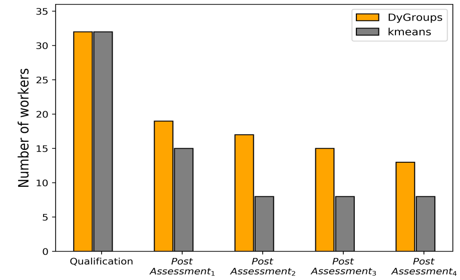
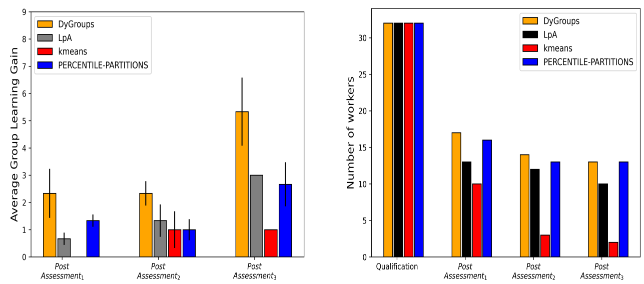
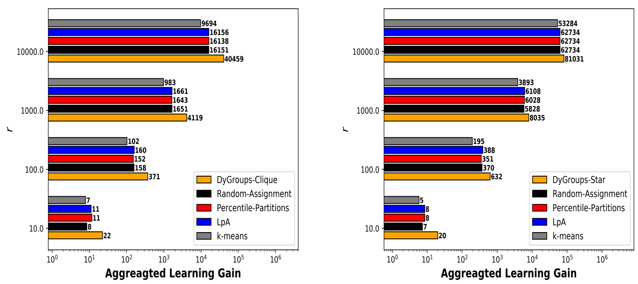


Fig. 3: Experiment-1: Worker retention



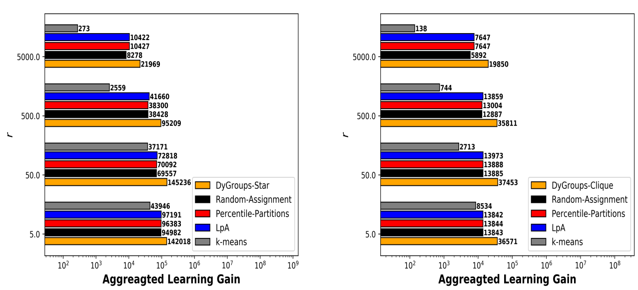
(a) Learning gain across rounds (b) Worker retention

Fig. 4: Experiment-2: Results



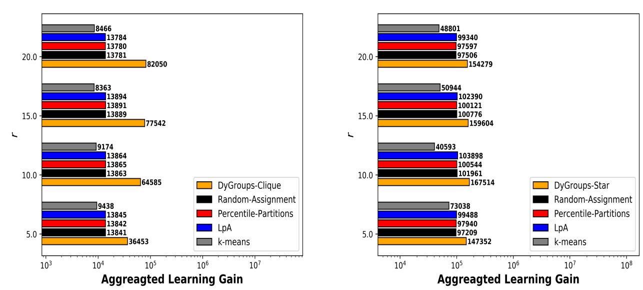
(a) Clique, log-normal (b) Star, Zipf

Fig. 5: Aggregate Learning gain - varying n



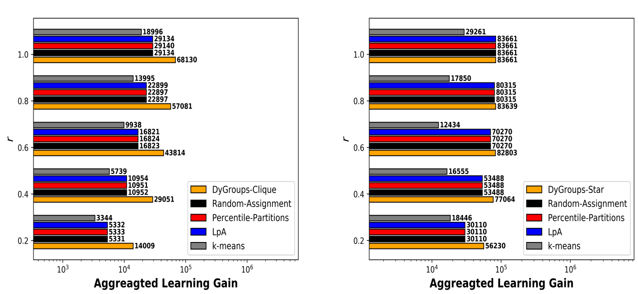
(a) Star, log-Normal (b) Clique, Zipf

Fig. 6: Aggregate Learning gain - varying k



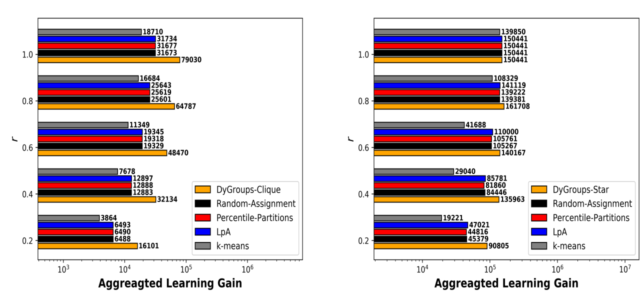
(a) Clique, Zipf (b) Star, log-normal

Fig. 7: Aggregate Learning gain - varying α



(a) Clique, Zipf (b) Star, Zipf

Fig. 8: Aggregate Learning gain - varying r



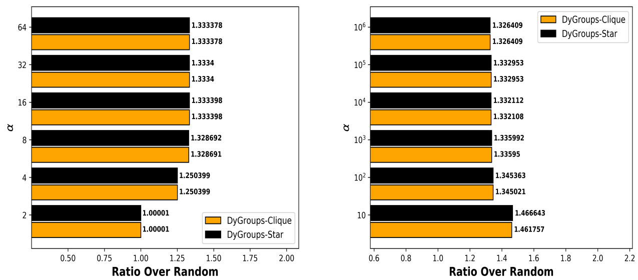
(a) Clique, log-normal (b) Star, log-normal

Fig. 9: Aggregate Learning gain - varying r

5) Fairness: We measure the *inequality* of the distribution of skills in DYGROUPS vs that in RANDOM-ASSIGNMENT. For this experiment we use $r = 0.1$. We use two metrics: the CV-coefficient of variation [8], and the well-known **Gini** coefficient [9]. Inequality **drops** with both methods (Figure 11(b)),

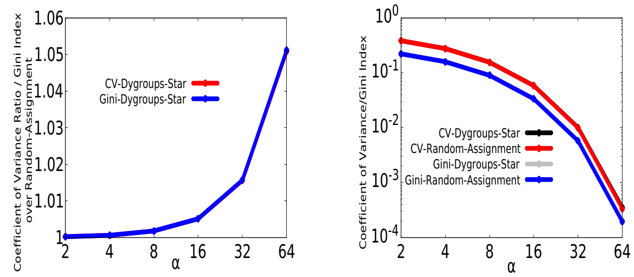
[8]CV is the ratio of the average by the standard deviation of skills.

[9]The Gini coefficient is $G = \frac{\sum_{i>j} |s_i - s_j|}{n \sum_i |s_i|}$, where s_i are skill levels.



(a) Varying α (b) Varying n

Fig. 10: Learning gain relative to RANDOM-ASSIGNMENT



(a) Inequality Ratios (b) Inequality Measures

Fig. 11: Inequality relative to RANDOM-ASSIGNMENT

something which may be expected due to the fact that there is an upper bound in the skill level. However, in Figure 11(a) we plot the **ratio** of the CV and the Gini index in DYGROUPS-STAR and RANDOM-ASSIGNMENT. We observe that DYGROUPS-STAR allow a higher inequality relative to RANDOM-ASSIGNMENT in all rounds, and that the gap between the two methods appears to be *widening* over time.

6) Running Time Experiments: The running time of both DYGROUPS variants is dominated by the time to sort the skill values. This leads to excellent scaling behavior, shown in Figures 12,13. In practice, the time to run DYGROUPS is negligible. For instance, performing 5 rounds on $n = 10^5$ participants takes 0.18 sec.

VI. RELATED WORK

We are unaware of any existing works that present models and algorithms for targeted groups formation in multiple rounds. In this section, we present the relevant existing works.

Groups formation: Designing quantitative models and algorithms for groups formation to optimize learning through peer interaction is first studied by Agrawal et al. [1]. The authors propose learning gain models and algorithms for the one-shot groups formation problem. Esfandiari et al. [2], adapt the learning gain functions of [1] but also add the additional dimension of affinity to optimize peer learning. Besides [1], [2], another of our prior works studies group formation to optimize recommendation [11].

Related problems have been considered by the operation research community; the problem is always formalized as an Integer Programming Problem (ILP) and often solved using simulated annealing [12], branch-and-cut [13] or genetic algorithms [14]. From an algorithmic standpoint, Anagnostopoulos et al. [15] present a general framework for a task-assignment problem and a series of approximation algorithms with theoretical guarantees. In addition, Anagnostopoulos et al. [16], study how to form teams for a series of arriving tasks without overwhelming any expert, and there is some communication between teams. Lappas et al. [17], introduce a team formation problem that also involves skill requirements and communication costs. Rangapuram et al. [18] propose approximation algorithms for solving a constrained matching problem via the densest subgraph problem. Sanaz et al. [19], [20] solve the non-overlapping teams formation problems. Especially in [19], the authors aim to maximize the potential of students' learning in online classes.

Unlike these existing works, we study peer learning in a dynamic setting where group composition changes over time. While we adapt learning gain models of [1], [2], we note that the existing solutions do not extend to TDG.

Information Diffusion/ Gossip Propagation: TDG is in some sense a *diffusion* problem: knowledge is diffused via pair-wise interactions, and the objective is to attain the maximum possible diffusion (as measured by the total skill) in a specified number of rounds. Diffusion problems are very well studied in various contexts, including information diffusion maximization in social networks (e.g. [5], [21]–[24]), and gossip propagation (e.g. [25], [26]). A tangential topic is the problem of influence maximization, introduced by Kempe, Kleinberg, and Tardos [27].

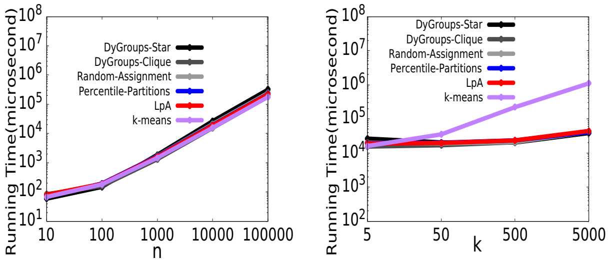
However, all these works assume the presence of a graph topology or network. Conversely, TDG *assumes a fully connected underlying network, and instead asks for a controlled utilization of its resources over time in a group-like manner.*

VII. DISCUSSION AND FUTURE WORK

We believe that our study can stimulate further research in this space, along the following directions.

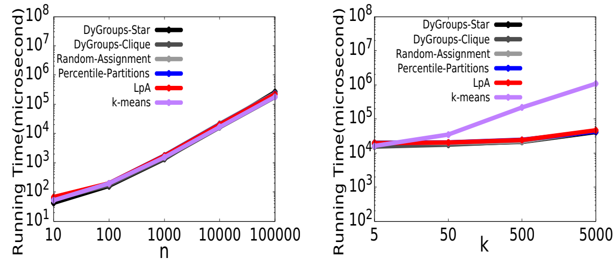
Alternative formulations. Our formulation assumes equal-size groups and linear learning gain. These settings are directly adopted from related works [1], [8], [28], [29], including education literature. We note that DYGROUPS can be adapted for the case when groups have varying sizes. Of course, more complicated models are conceivable. A particularly interesting problem is to study settings where the learning gain depends on additional factors that capture "intrinsic learning ability", e.g. a time-evolving affinity among individuals [8] that impact learning, or different learning rates for the participants. One possible way to model the former problem is to solve a bi-criteria optimization problem, with the goal of forming dynamic groups where both affinity and skill evolves across rounds. Another interesting question, motivated by Observation III in Section V-A is the impact of *retention* on the aggregate learning gain. A faster overall learning gain may still higher satisfaction among participants, and thus create a positive feedback loop.

DYGROUPS for more groups. Recall that we proved the optimality of DYGROUPS-STAR for the case k=2. Clearly,



(a) Varying n (b) Varying k

Fig. 12: Running time (in micro sec), Star, log-normal



(a) Varying n (b) Varying k

Fig. 13: Running time (in micro sec), Clique log-normal

DYGROUPS can still be used to solve the problem when $k > 2$ and we conjecture that DYGROUPS-STAR is still optimal. That said, the computational complexity of the problems for different interaction modes is an open problem for larger values of k, possibly of independent theoretical interest.

Other learning gain functions. As long as the learning gain function is concave, DYGROUPS can be adapted to solve TDG. Our initial research suggests that for non-linear concave learning gain functions, DYGROUPS is not optimal, thus raising questions on the approximability of the optimal, or other theoretical guarantees.

Fairness. We have only scratched the surface with respect to fairness. We believe that studying bi-criteria optimization problems with respect to fairness and learning gain is an extremely interesting theoretical and practical issue, even within the scope of relatively limited peer learning models.

VIII. CONCLUSION

We initiate the study of peer-learning processes in *rounds*. The problem is motivated by practical considerations of groups in online social networks or offline classroom learning. In each round, the participants are split into groups and learn from interactions within their group. The objective is to find a *sequence* of groupings that will maximize the total knowledge at the end of the process. We introduce a model and an associated algorithmic framework DYGROUPS for this problem. Using the insights we gain from our theoretical study, we design experiments with human subjects that provide evidence corroborating our hypothesis that the choice of groupings can indeed significantly impact the total amount of learning.

ACKNOWLEDGMENT

Dong Wei and Senjuti Basu Roy are supported by the NSF, CAREER Award #1942913,IIS #2007935, IIS #1814595, and by the Office of Naval Research Grant No:N000141812838. I. Koutis is supported by the NSF, CCF Awards #2039863 and #1813374.

REFERENCES

[1] R. Agrawal *et al.*, "Grouping students in educational settings," in *SIGKDD*, pp. 1017–1026.
[2] M. Esfandiari *et al.*, "Optimizing peer learning in online groups with affinities," in *SIGKDD*, 2019, pp. 1216–1226.
[3] H. Rahman *et al.*, "Optimized group formation for solving collaborative tasks," *The VLDB Journal*, vol. 28, no. 1, pp. 1–23, 2019.
[4] L. Backstrom *et al.*, "Group formation in large social networks: membership, growth, and evolution," in *SIGKDD*, 2006, pp. 44–54.
[5] J. Ugander *et al.*, "Structural diversity in social contagion," *PNAS*, vol. 109, no. 16, pp. 5962–5966.
[6] G. Palla *et al.*, "Quantifying social group evolution," *Nature*, vol. 446, no. 7136, pp. 664–667, 2007.
[7] P. Holme *et al.*, "Temporal networks," *Physics reports*, vol. 519, no. 3, pp. 97–125, 2012.
[8] R. Agrawal *et al.*, "Grouping students for maximizing learning from peers." *EDM*, 2017.
[9] N. M. Webb, "Peer interaction and learning in small groups," *International journal of Educational research*, vol. 13, no. 1, pp. 21–39, 1989.
[10] J. A. Bargh *et al.*, "On the cognitive benefits of teaching." *Journal of Educational Psychology*, vol. 72, no. 5, p. 593, 1980.
[11] S. Basu Roy *et al.*, "From group recommendations to group formation," in *SIGMOD*, 2015, pp. 1603–1616.
[12] A. Baykasoglu *et al.*, "Project team selection using fuzzy optimization approach," *Cybernetics and Systems*, vol. 38, no. 2, pp. 155–185, 2007.
[13] A. Zzkarian *et al.*, "Forming teams: an analytical approach," *IIE transactions*, vol. 31, no. 1, pp. 85–97, 1999.
[14] H. Wi *et al.*, "A team formation model based on knowledge and collaboration," *Expert Systems with Applications*, vol. 36, no. 5, pp. 9121–9134, 2009.
[15] A. Anagnostopoulos *et al.*, "Power in unity: forming teams in large-scale community systems," in *CIKM*, 2010, pp. 599–608.
[16] ——, "Online team formation in social networks," in *WWW*, 2012, pp. 839–848.
[17] T. Lappas *et al.*, "Finding a team of experts in social networks," in *SIGKDD*, 2009, pp. 467–476.
[18] S. S. Rangapuram *et al.*, "Towards realistic team formation in social networks based on densest subgraphs," in *WWW*, 2013, pp. 1077–1088.
[19] S. Bahargam *et al.*, "Team formation for scheduling educational material in massive online classes," *arXiv preprint arXiv:1703.08762*, 2017.
[20] ——, "The guided team-partitioning problem: Definition, complexity, and algorithm," *arXiv preprint arXiv:1905.03037*, 2019.
[21] C. Wang *et al.*, "Scalable influence maximization for independent cascade model in large-scale social networks," *DMKD*, vol. 25, no. 3, pp. 545–576, 2012.
[22] K. Saito *et al.*, "Prediction of information diffusion probabilities for independent cascade model," in *KES2008*. Springer, 2008, pp. 67–75.
[23] A. Goyal *et al.*, "Simpath: An efficient algorithm for influence maximization under the linear threshold model," in *ICDM*. IEEE, 2011, pp. 211–220.
[24] A. Guille *et al.*, "Information diffusion in online social networks: A survey," *ACM Sigmod Record*, 2013.
[25] P. G. Lind *et al.*, "Spreading gossip in social networks," *Physical Review E*, vol. 76, no. 3, p. 036117, 2007.
[26] J. Tang *et al.*, "Gossip-based scalable directed diffusion for wireless sensor networks," *IJCS*, 2011.
[27] D. Kempe *et al.*, "Maximizing the spread of influence through a social network," in *SIGKDD*, pp. 137–146.
[28] C. K. Jackson *et al.*, "Teaching students and teaching each other: The importance of peer learning for teachers," *American Economic Journal: Applied Economics*, vol. 1, no. 4, pp. 85–108, 2009.
[29] R. Hertz-Lazarowitz *et al.*, *Learning to cooperate, cooperating to learn*. Springer Science & Business Media, 2013.