Maximizing Satisfaction in Group Formation

Sean Munson
Franklin W. Olin College of Engineering
Needham, MA 02492
Email: sean.munson@students.olin.edu

Grant Hutchins
Franklin W. Olin College of Engineering
Needham, MA 02492
Email: grant.hutchins@students.olin.edu

***Abstract*— Maximizing satisfaction in how people are grouped is problem in which, once a certain number of people are reached, it is not possible to examine all possible solutions. A number of heuristics were explored to approximate maximum satisfaction for different size teams for social and academic situations. These approximations produce sets of teams for which the predicted satisfaction is much improved over the sets of teams generated by random methods.**

I. Introduction

Many academic, business, and other organizations require the formation of groups for the completion of tasks or for smaller meetings. Often, the selection of these groups does not take into consideration much more than one person's arbitrary judgement of who works well together. Other times, the groups might even be completely random.

Collecting preference data from each member of an organization and following some optimized algorithm might save time and hopefully prevent problems within groups. Moreover, collecting data over time on the success of groupings and linking these data to the preferences of members could help inform grouping decisions in the future.

II. Methodology

A. Data Source

Data was collected from a 32-student discrete mathematics class at Olin College of Engineering. Students were asked to evaluate their desire – positive, negative, or neutral – to be teamed with each other student in four situations:

- a group of four to complete homework for the discrete mathematics course,
- a pair to complete a project, such as this one, for the same class,
- a pair to take a cross-country road trip, and
- a group of four to go out to dinner.

Additionally, data on work styles and whether or not students had previously worked with each other was collected.

Out of the 32 students in the class, 31 completed the survey. The remaining person was assumed to have neutral preferences, as would be done if this system was used to create actual groups. Once groups were generated, students were asked to return and evaluate the proposed teams on a scale of 1 to 5.

B. Satisfaction

The preferences expressed on the survey can be expressed as a weighted graph, in which each person represents a vertex and the preferences are directed edges. Positive preferences are given a weight of one, negative preferences are given a weight of negative one. Neutral preferences are disregarded. With each of the groups formation tasks, the goal was to maximize the number of "ideal" teams, or groups in which each person wants to work with each other person in the group. Each proposed group was given a score based on the edges contained within this group; i.e. a group of four in which each person indicated they wanted to work with each other person would receive a score of 12; a group of four in which each person indicated they did not want to work with each other person would receive a score of -12.

Achieving maximum satisfaction turns out to be an NP-Complete problem. Each possible group's score would have to be compared to each other possible group's score to find the maximum. In our example with 32 students, there are

$$\frac{\binom{32}{4}\binom{28}{4}\binom{24}{4}\cdots\binom{4}{4}}{8!} = 2.6462 \times 10^{119} \qquad (1)$$

possibilities for combinations of groups of four and

$$\frac{\binom{32}{2}\binom{30}{2}\binom{28}{2}\cdots\binom{4}{2}}{16!} = 3.0037 \times 10^{233} \qquad (2)$$

possible combinations of groups of two. It took approximately two minutes to randomly generate and score 200,000 teams; needless to say it would not be feasible to run all possible combinations. Instead, approximations must be used.

It is also worth noting that there is an upper boundary whether one attempts to brute-force maximum happiness or to approximate it: a if a set of teams has the maximum possible score, the process can stop; there may be other sets of teams that also could achieve this score but since the maximum possible satisfaction has already been achieved, it does not matter.

III. Approximations Based on Preferences Towards Others

Consider forming teams of three from among nine people whose preferences are expressed in the directed, weighted graph shown in Figure 1. We can use preference data to attempt to intelligently generate teams using greedy algorithms.

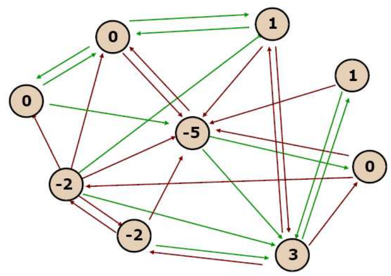


Fig. 1. Sample graph of nine people with preferences towards each other. Popularity of each person is indicated on the vertex.

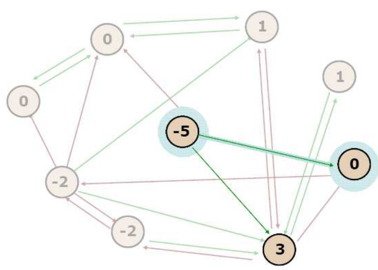


Fig. 2. First steps in chain approximation.

Approximations evaluated started by calculating the "popularity" of each person. A person's popularity is the sum of the weights of the edges towards their vertex. The popularity of the people represented in this set is shown in Figure 1.

A. Chain Approximation

The first approximation tried was a simple chain of preferences. The starting vertex for the chain to be built in this example is the least popular person, though most popular was also evaluated. Once the least popular person is identified (in this case the person with a popularity of negative five), their preferences are examined. If there are people they want to work with (a +1 edge), the least popular of these is selected and added to the team. If there are not, the least popular of the people they feel neutral about is added. Finally, if they were to feel negatively about all remaining available people, the most popular remaining person would be added. Completion of the first selection is shown in Figure 2.

The preferences of the newly added person are then examined and a third person is added according to the process previously described. In the example, the new team member feels neutrally about all but one person with whom they would prefer not to be in a team, and so the least popular person towards whom they feel neutrally is added. At this point, the team has three people. A new team is started by finding the least popular available person and building a new chain off of that person. The concluding step to form the first team and the initial member of the second team are shown in Figure 3(a); the final outcome is shown in Figure 3(b).

1) Variations: A number of variations on this particular chain approximation are possible; some were evaluated. Among those tried were giving priority to the most popular

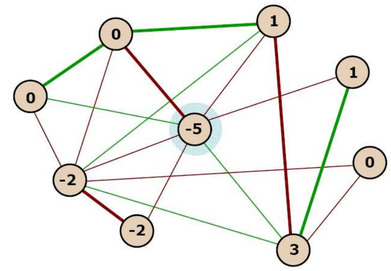


Fig. 4. First steps in group approximation.

person instead of the least popular person when starting a new team or deciding among possible preferred people. It is also possible to substitute "pickiness" – the sum of the preferences a person expresses towards others rather than of those expresses towards the person – for popularity in this process.

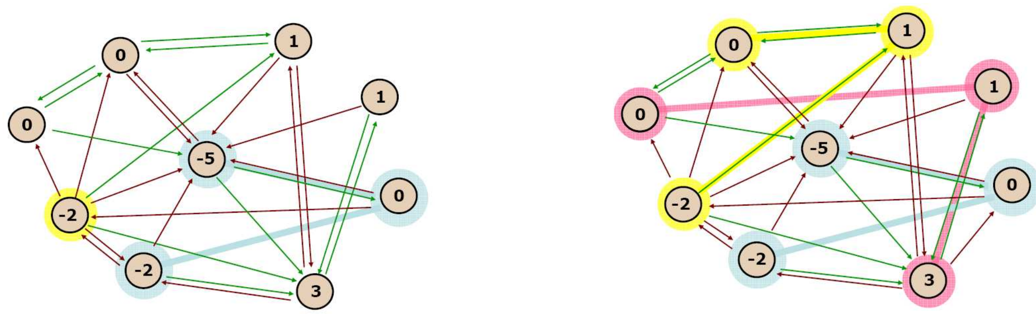
2) Evaluation: The strengths of this approximation are its speed and simplicity. It also does attempt to give all but the last pick in each team someone else with whom they want to work. However, a number of problems are readily apparent. As seen when adding the third person to the first team in our example (Figure 3(a)), people sometimes want to work with someone who would prefer not to work with them. Since the chain approximation only takes into consideration the preferences expressed in one direction at any time, the feelings of potential additions towards anyone in already in the team are not considered. Also not considered are the feelings of others already in the group towards potential additions.

B. Group As a Person Approximation

The weaknesses of the chain approximation directed the authors to develop an approximation that would consider a greater number of preferences at a time. The process for this approximation starts much the same as the chain approximation, by calculating each person's popularity. Once this is complete, edges are collapsed, or reduced. First, directionality is removed. Then, the edge weights between each pair of vertices are added together to produce an appropriate new single edge weight. Two negative edges results in a new -2 edge, a positive edge and a negative edge produce a neutral edge, and two positive edges produce a +2 edge. As with the chain approximation, a number of options exist for selecting the first person; for this example, we will again use the least popular person. The example data set, collapsed and with the first pick, is shown in Figure 4.

Next, instead of following the chain approximation's approach of selecting the person towards whom the initial pick has the most positive feelings, the person with whom the the initial pick has the most positive feelings is selected. If there are multiple people in this category, the least popular person is selected as in the chain approximation (Figure 5(a)).

The two people on the team are then merged into one vertex; the edges of these two vertices are then combined by the previous rules for collapsing vertices (Figure 5(b)). The



(a) Final step in formation of first team, plus initial member of second team

(b) Complete chain approximation teams.

Fig. 3. Additional Chain Approximation Steps

process is then repeated: out of the people with whom the existing group members share the most positive feelings, the least popular is added (Figure 5(b)).

Once a a complete team is formed, the process starts again with the least popular available person to build an additional team.

1) Variations: As with the chain approximations, a number of variations on this process are possible. Most popular, instead of least popular, people could be used as the seed for each team, or pickiness could be substituted for popularity.

2) Evaluation: This heuristic represents a significant improvement – at least in theory – over the chain algorithm by considering preferences all of the people in the team and all of the potential additions to a team when making a decision. It does, however, explore only one option in all of the possible sets of teams, a disadvantage shared with the chain approximation.

IV. Hill Climbing

To explore more feature space, a hill-climbing stage, with some features of genetic algorithms, was developed. A set of teams generated randomly or by the any of the variants of the chain or group approximations is the seed for the hill-climbing phase. In this stage, each team is first scored based on the edges between its members. The half of the teams with the highest scores are locked-in, while the others are designated for modification; this is a "survival of the fittest" sort of approach.

Among the teams designated for modification, each person is given a score based on how they affect the satisfaction of the team: how do the other team members feel about them plus how do they feel about the other team members. In each team, the person who gets the lowest score is removed. The removed individuals are then randomly reinserted into the teams.

The modified set teams of teams is then scored; if it has a higher score than the previous high scorer, it replaces it as the best known solution. Regardless of whether or not it receives a better score, the process continues with this new set of teams: each team is re-scored, the new scores may mean different teams are locked or unlocked, individual scores are recalculated, and individuals reshuffled. Each tree is followed for 100 shuffles. After 100 shuffles, the process restarts from the original seed to explore a new tree. The best result found during these 100,000 explorations is returned.

The hill climbing stage was almost always able to produce a set of teams with a higher predicted level of satisfaction than the input seed. Improvements were most noticeable for the chain approximation; results tended to be only marginally better for the group approximation.

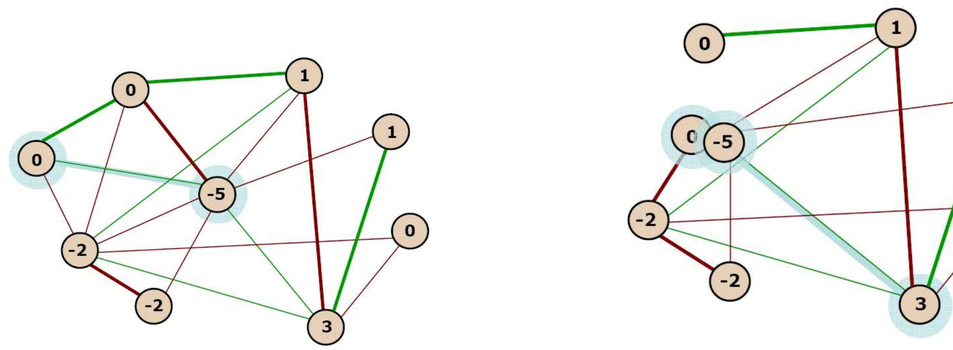
V. Nearest Neighbor Approximation

Another algorithm, the Nearest Neighbor Approximation, was used to create groups and pairs of students. In this method, student preferences are not considered. Instead, the data collected from work groups are considered.

Three questions were asked to the students about their work habits:

- I prefer and tend to get group work done...
  - well before it is due. (2 points)
  - at the last minute. (1 point)
  - it doesn't really matter. (0 points)
- I prefer to do group work...
  - with each person responsible for distinct parts. (1 point)
  - all together as a group. (2 points)
  - however; it doesn't really matter. (0 points)
- How focused do you tend to be when doing group work?
  - I tend to be completely focused on getting it done quickly. (0 point)
  - I am silly a bit, or goof off sometimes, but am generally pretty focused. (1 point)
  - If I'm having fun talking about other things with my team member, it's fine if the project takes twice as long (2 points)

The answers from these three questions were represented as 3-tuples, spread out over the space of a cube as shown in Figure 6.



(a) Second step in group approximation

(b) Third step in group approximation

Fig. 5. Additional Group Approximation Steps

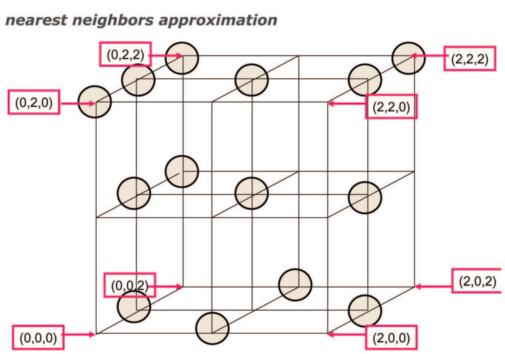


Fig. 6. Cube representation of 3-tuples ranging from (0,0,0) to (2,2,2) which represent different responses to the work habits questions. Each corner of the cube is labeled. Each dot represents one or several students. Note that not every node contains students

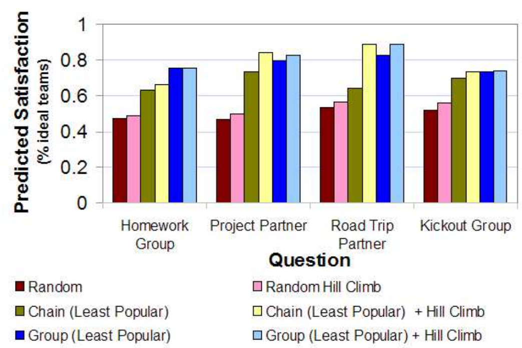


Fig. 7. Comparison of the predicted satisfaction of teams formed by the six algorithms involving personal preferences

The algorithm takes advantage of the fact that a cube has 8 corners (each representing the possible 3-tuples containing only 0's and 2's) and that 32 students would form into 8 teams of 4.

Eight separate rankings of students were created based on their distances from each corner of the cube. For example, students with a response 3-tuple of $(2, 0, 2)$ would have the highest rank in the $(2, 0, 2)$ ranking, but the lowest rank in the $(0, 2, 0)$ ranking.

The algorithm begins by iterating once over each corner of the cube. Each corner of the cube defines a team. Each team selects the highest-ranked unselected student using its corner's ranking. This selection is somewhat analogous to assigning "captains" to select the remainder of their teams. The algorithm then continues iterating through the teams, each time adding its highest-ranked unselected student. After the fourth iteration, the teams are complete.

A variation of this algorithm selects pairs. The first two iterations assign eight pairs, which are then finalized. The third and fourth iterations assign eight additional pairs. Note that this method is identical to taking the teams of four and splitting them in half.

Since the only evaluations of the quality of teams come from the second part of the student survey, the effectiveness of this algorithm is difficult to properly measure. The usefulness of similar work styles among members of a team might not be clear until after the team has gone through a few work experiences, especially among students who do not yet have opinions about one another. Also, students with identical work style opinions might happen to have highly negative opinions of each other, contributing to overall team dissatisfaction.

In retrospect, the weighting of the first two questions is not properly ordered. Opposing choices should not be adjacent in these cases since they represent the two ends of a spectrum. Ideally, the "doesn't matter" choices would be weighted as 1 so that they would be close in distance to the 0 and 2 choices. Most likely, renumbering the choices properly should make the resultant teams more representative of the choices of the students.

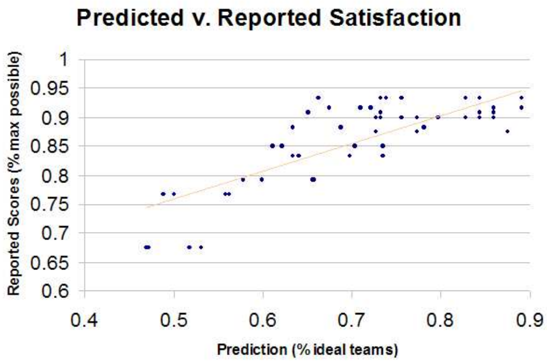


Fig. 8. Correlation graph between the predicted satisfaction of teams and the reported satisfaction. Data are from the six algorithms involving personal preferences

	Preference	Nearest Neighbor	Random
Question 1	3.6	3.0	2.7
Question 2	4.3	3.0	3.5
Question 3	4.2		3.5
Question 4	3.9		3.1

Fig. 9. Average student rating of teams formed by different algorithms on a 1-5 scale

VI. RESULTS

The average predicted satisfaction for teams from each of four personal preference algorithms, plus random are compared in Figure 7. Note that the random hill-climb method does not actually follow the process described above but simply picks the best of 200,000 randomly generated sets of teams.

25 of the 32 students completed a double-blind team evaluation survey. Each student was shown teams and pairings from each of the six student preference algorithms and from the nearest neighbor algorithm, padded with a few random teams. The students gave a rating between 1 and 5 for each teaming, with 1 representing the most negative opinion and 5 representing the most positive opinion. This opinion represents students' initial reactions to teams made up of people they already knew and with whom in many cases had prior work experience.

From these data, actual reported team satisfaction was determined by averaging all of the responses from a given set of teams. These data are compared with the predicted values in Figure 8. Confirming a correlation between predicted and actual scores was critical in determining whether or not predicted satisfaction was an acceptable measure of the acceptability of a given set of teams. Where there too many subtle nuances created by combining a group of four that go beyond what was expressed in the preferences survey?

Overall, the reported scores were higher than those predicted, implying that students were more satisfied with the results than expected. In addition, the teams which the model predicted would have low satisfaction also reported lower satisfaction.

The preference-based algorithms always outperformed both the work habit-based algorithms and the random teams. Not once in the 200,000 random sets of teams generated for each scenario did a random set of teams result in greater satisfaction than even the worst team generated by one of the preference-based algorithms. The nearest neighbor algorithm performed comparably to the random teams, but the incorrect ordering of the work habit results probably caused this number to be artificially low. Overall, the preference-based algorithms seem to outperform the other methods and thus probably make sense to use in organizations that currently do not employ an informed teaming methodology.

VII. FUTURE WORK

This work represents a first start at evaluating some methods for group formation in both social and academic settings. At first pass, these methods appear to offer significant improvements over randomly generated teams. Not studied, however, was how the proposed teams might perform: does working with the people one prefers produce better performing teams?

To consider this question, a longer term study that monitors performance of generated teams would be particularly helpful. This data is also necessary to truly evaluate the performance of any work-habits based teams. A longer term study would also allow for performance of past teams to be factored into the generation of new teams.

Beyond the desire for a longer term study, a number of variations of the existing algorithms can be tried, including those that weight preferences based on prior experience or perhaps those that use work-habits instead of popularity to break ties in the preferences-based approximations.

Additionally, all approximations considered do not distinguish in quality between a set of teams in which everyone is somewhat satisfied and a set of teams in which half of the people are very satisfied and half are very unsatisfied as they would produce the same overall score. The distribution of satisfaction for given team groups merits consideration and it may be found that modifying the approximations to actively avoid extreme dissatisfaction in a group of teams, even at the cost of some net satisfaction, is preferred.

VIII. CONCLUSION

A number of heuristics were explored to approximate maximum satisfaction in teams for social and academic situations. These approximations generate sets of teams in which the team members are predicted to be, and report being, happier than in existing teams. The software developed could be deployed with little modification to support team generation in classrooms, or to support a longer term study.

ACKNOWLEDGMENT

The authors would like to thank Drs. Sarah Spence and Allen Downey of Franklin W. Olin College of Engineering for their assistance in this project. Additionally, the authors are grateful for the time the students of the discrete mathematics course spent completing the survey.