TA Allocation Policy - Click Here for TA Allocation Software

Instructions on how to use this application

Assignment of TAs for our Courses

The TA allocation to a course is based on the principle of stable marriage and is a modified form of Gale & Shapley's algorithm.

What forms a stable marriage in our case?

The algorithm takes in a list of students and courses with their preferences for the courses and students, respectively. It gives a stable allocation based on a student's preference for a course and a course's "preference" for a student. In our stable allocation, there is no such scenario:

1. **Student s1 gets a TAship for Course c1 but actually prefers c2**

1. **A Course c2 gets a Student s2 as a TA, but it actually "prefers" another Student s1**

We selected stable marriage allocation since it has one of the highest welfare gains of all allocation algorithms (it means it's good).

Our solution works in two phases.

1. **Generate TA Quotas for each course**

1. **Allocate TAs to a course.**

---

1. Precondition

The application is executed only after course allocation to all the faculty members is complete.

---

2. Generate TA Quotas.

Generation of TA quota means that the algorithm decides how many TAs (FD, HD, PhD together) are required for this course. For this purpose, it uses the following attributes of a course:

- a) Course Type: CDC without lab, CDC with Lab, Elective, Elective with Lab/Assignment, First Degree or Higher Degree or WILP

- b) Typical course strength *(n)*: Since we have to allocate TAs even before registration, it is impossible to know the exact course strength. In view of this, we use historical enrollment data starting from the previous semester.

Each course (c) attribute has a weight (w) associated with it, which plays a role to decide the number of TAs required. A CDC with Lab has the highest weightage, followed by Elective with Lab => CDC without Lab => Elective.

First, the normalized weight of course c is calculated: wavg_c = (n_c*w_c)/sum (n_i*w_i) for all courses. Next, the TA quota is calculated as Total_TA*wavg_c.

For instance, there are a total of 4 courses. Course A is CDC with lab and w_A= 1.5. Suppose that this course has 120 students enrolled. Thus, n_A*w_A = 120*1.5=180. Suppose that n*w factor of the other three courses, B, C, and D, are 180, 150, and 120, respectively. In that case, the normalized weight for A wavg_A = 180/(180+180+150+120)=0.286. Suppose that we have a total 20 FDTA available for the department. The course A gets 20*0.286=5.72 i.e. 6 FDTAs. We apply a similar computation to obtain the number of HD and PHDTAs for course A, once we know the total number of HD and PHDTAs available.

Other factors exist, such as the fact that FDTAs can’t be assigned to an HD course. A first degree has more weightage for FDTAs than HDTAs and PhDTAs. These are also considered while computing the quota. We do not elaborate them in detail for brevity.

---

3. Allocate TAs to a course

Students' preferences are taken using a web-based GUI (check our intranet). A student has 5 preferences in decreasing order, spread across FD and HD level courses.

Course preferences are computed based on the **CGPA** and **subject grade** of that student in that particular course. If two or more FD students have identical CGPA and subject grades, the tie is broken randomly.

- 3.1 Steps
- 1. Round 1:
    - a. [propose] The algorithm tries to allocate a course to each student based on his/her first preference, and each subject gets some proposals.
    - b. [choose] Each subject does its choosing based on its preference, with each subject choosing the top k number of students as it's provisional TAs; where k is the quota of TAs allocated to it.
    - c. [reject] The students rejected by subjects go to the next round.
- 2. Round 2:
    - a. [propose] the students propose their second preferences in the next round.
    - b. [choose] Now, the subjects again do choosing, mixing the new students with its provisional TAs, and again selecting top k students.
    - c. [reject] The students rejected here go to the next round.
- 3. The algorithm terminates when it runs out of students to proceed to the next round or when the students in the proposing pool have exhausted their filled preferences (that is, they are rejected by all the filled subjects).

The abovementioned steps are applied to FDTAs, HDTAs, and PHDTAs individually. Since no CGPA or subject grade data is available for an HD or a PhD student, a course has NO preference for an HD or a PhD student. Thus, for PhD and HD students, the algorithm first allocates a course to an HD/PHD student based on his/her first preference. Next, a course will randomly select a set of HD/PHD students from its list of proposals. The rejected students go to the next round for their second preference and so on.

---

4. Final Allocation

There is no guarantee that every student will get a TAship or a course will get a TA. Thus, it is possible that a course does not get any TA at all or that a student remains unallocated. The solution's output is reviewed, and such corner cases are handled manually before submitting the allocation to the instructions office.
