EECS 70 Discrete Mathematics and Probability Theory
Fall 2014 Anant Sahai

Homework 3

This homework is due September 22, 2014, at 12:00 noon.

1. Propose-and-Reject Lab

In this week's Virtual Lab, we will simulate the traditional propose-and-reject algorithm. You will find a simple implementation of a `Person` class, where each `Person` has an identification number, is either male or female, and has his or her own preference list, in the code skeleton for this question.

Please download the IPython starter code from Piazza or the course webpage, and answer the following questions.

(a) Using the example from Note 4, page 6, which is shown again below for your convenience, create one list which contains all the men, and one which contains all the women. Each man or woman needs to be created using the `Person` class.

Men	Women			
1	A	B	C	D
2	A	D	C	B
3	A	C	B	D
4	A	B	C	D

Women	Men			
A	1	3	2	4
B	4	3	2	1
C	2	3	1	4
D	3	4	2	1

(b) For this question, we've given you the code to run the traditional propose-and-reject algorithm where the men propose to the women. Your task is to create a new version by changing it to women making proposals. Are the final pairings for the two cases the same?

(c) This question will focus on exploring what happens when we randomize preferences. In the code skeleton, you will find an implementation of the function `create_random_lists`, which creates a random set of preferences.

We're interested in how often the men-propose and women-propose algorithms return the same pairings. Write a function that generates a random set of preferences for men and women, then runs each variant of the traditional propose-and-reject algorithm on that set of preferences. For lists of 4 people, how often do the men-propose and women-propose algorithms agree on the final stable pairings?

(d) Finally, we're going to explore how long the women-propose algorithm takes as a function of n. In lecture, we learn that the algorithm must terminate after at most n^2 days (we will soon prove a stricter bound in a later question). Write a function that returns how many days it takes for the traditional propose-and-reject algorithm to arrive at its stable solution using randomly-generated preference lists. Try it on some inputs of different size. How quickly does the number of days grow with the input? Does it grow linearly (at the same rate as n - maybe twice as fast or half as fast), quadratically, logarithmically? If you can't tell, try graphing some outputs by hand. You do not have to submit any graph, but it certainly would help defend your claim.

Reminder: When you finish, don't forget to convert the notebook to pdf and merge it with your written homework. Please also zip the `ipynb` file and submit it as `hw3.zip`.

2. Candy Problem

There are N students standing in a circle, facing the center. Each student is initially issued an even number of candies. Being the fair and honest students that they are, they come up with the following adjustment algorithm. First, each student gives half of his or her candies to the student on his or her left. Note that after this step, some students might have an odd number of candies. Next, those students with an odd number of candies will get one more candy from the teacher. The students repeat the adjustment algorithm and stop when everyone has the same number of candies.

(a) Run the algorithm for the case where there are 6 students and the initial number of candies, in clockwise order, is: {2, 4, 4, 2, 6, 8}.

(b) By the Well-Ordering Principle, before we begin the adjustment algorithm, there is a minimum number of candies possessed by any student and a maximum number of candies possessed by any student. Since these quantities are even, we let $2n$ denote the minimum number of candies and let $2m$ denote the maximum number of candies in the initial candy distribution. Prove that the number of candies possessed by each student is still between $2n$ and $2m$ after one iteration of the adjustment algorithm.

(c) Now, suppose the minimum number of candies that any student has immediately after iteration i is $2k$, and that p students have exactly $2k$ candies. Prove that, provided the algorithm doesn't terminate immediately after iteration i, less than p students will have exactly $2k$ candies immediately after iteration $i+1$.

(d) Does the adjustment algorithm always terminate in a finite number of iterations, or could students be trading candies with each other forever? Explain your reasoning.

3. Indifferent Attitudes

In the real world, it is perhaps a bit unrealistic to expect that each man and woman can provide a *strict* preference ordering of all his or her potential mates. To reflect this, let's suppose we allow each person's preference list to contain *ties*. Mathematically, a set W of k women forms a tie of length k in the preference list of man m if m does not prefer w_i to w_j for any $w_i, w_j \in W$ (i.e., m is *indifferent* between w_i and w_j), while for any other woman w not in W, either m prefers w to all women in W or m prefers all the women in W to w. A tie on a woman's list is defined analogously. We will call this the *Stable Marriage Problem with Ties*.

Recall that in the original Stable Marriage Problem, where preference lists are strictly ordered, it is always possible to find a stable matching, where a matching is stable if there is no man x and woman y such that x and y both prefer each other over their current partners. However, for the Stable Marriage Problem with Ties, 3 different types of stability are possible:

- **Weak stability.** A matching is *weakly stable* if there is no couple x and y, each of whom strictly prefers the other to their current partner in the matching.
- **Strong stability.** A matching is *strongly stable* if there is no couple x and y such that x strictly prefers y to his or her partner, and y either strictly prefers x to his or her partner or is indifferent between them.
- **Super-stability.** A matching is *super-stable* if there is no couple x and y, each of whom either strictly prefers the other to his/her partner or is indifferent between them.

It is then natural to ask whether these different types of stable matchings always exist in a given instance of the Stable Marriage Problem with Ties. Answer the following.

(a) For the Stable Marriage Problem with Ties, does a weakly stable matching always exist? Either prove the statement or provide a counterexample.

(b) For the Stable Marriage Problem with Ties, does a strongly stable matching always exist? Either prove the statement or provide a counterexample.

(c) For the Stable Marriage Problem with Ties, does a super-stable matching always exist? Either prove the statement or provide a counterexample.

(d) Assume we are given an instance of the stable marriage problem with ties, along with a weakly stable matching M for that instance. Upon getting married to their partners assigned in M, each person's preferences change slightly and the married partner becomes preferred over anyone they were tied with, but doesn't change in rank otherwise. Is M now super-stable?

4. Karl and Emma fight!

(a) Karl and Emma are having a disagreement regarding the traditional propose-and-reject algorithm. They both agree that it favors men over women. But they disagree about what, if anything, can be done without changing the ritual form of men proposing, women rejecting, and people getting married when there are no more rejections.

Karl mansplains: "It's hopeless. Men are obviously going to propose in the order of their preferences. It's male optimal so why would they do anything else? As far as the women are concerned, given that they face a specific choice of proposals at any given time, they are obviously going to select the suitor they like the most. So unless we smash the system entirely, it is going to keep all women down."

Emma says: "People are more perceptive and forward-looking that you think. Women talk to each other and know each other's preferences regarding men. They can also figure out the preferences of the men they might be interested in. A smart and confident woman should be able to do better for herself in the long run by not trying to cling to the best man she can get at the moment. By rejecting more strategically, she can simultaneously help out both herself and her friends."

Is Emma ever right? If it is impossible, prove it. If it is possible, construct and analyze an example (a complete set of people and their preference lists) in which a particular woman acting on her own (by not following the ordering of her preference list when deciding whether to accept or reject among multiple proposals) can get a better match for herself without hurting any other woman. Show how she can do so. The resulting pairing should also be stable.

(b) Karl and Emma have another disagreement! Karl claims that if a central authority was running the propose-and-reject algorithm then cheating the system might improve the cheater's chances of getting the more desirable candidate. The cheater need not care about what happens to the others.

Karl says: "Lets say there exists a true preference list. A prefers 1 to 2 but both are low on her preference list. By switching the reported preference order among 1 and 2, she can end up with 3 whom she prefers over 1 and 2 which wasn't possible if she did not lie. Isn't that cool?"

Emma responds: "That's impossible! In the traditional propose-and-reject algorithm switching the preference order 1 and 2 cannot improve A's chance to end up with 3."

Either prove that Emma is right or give an example of set of preference list for which a switch would improve A's husband (that is, she gets matched with 3), and hence proving Karl is right.

5. TeleBears

In the Course Enrollment Problem, we are given n students and m discussion sections. Each discussion section u has some number, q_u of seats, and we assume that the total number of students is larger than the total number of seats (i.e. $\sum_{u=1}^{m} q_u < n$). Each student ranks the m discussion sections in order of preference, and the instructor for each discussion ranks the n students. Our goal is to find an assignment of students to seats (one student per seat) that is *stable* in the following sense:

- There is no student-section pair (s,u) such that s prefers u to her allocated discussion section and the instructor for u prefers s to one of the students assigned to u. (This is like the stability criterion for Stable Marriage: it says there is no student-section pair that would like to change the assignment.)

- There is no discussion section u for which the instructor prefers some unassigned student s to one of the students assigned to u. (This extends the stability criterion to take account of the fact that some students are not assigned to discussions.)

Note that this problem is almost the same as the Stable Marriage Problem, with two differences: (i) there are more students than seats; and (ii) each discussion section generally has more than one seat.

(a) Explain how to modify the propose-and-reject algorithm so that it finds a stable assignment of students to seats.

(b) State a version of the Improvement Lemma (see Lecture Note 4) that applies to your algorithm, and prove that it holds.

(c) Use your Improvement Lemma to give a proof that your algorithm terminates, that every seat is filled, and that the assignment your algorithm returns is stable.

6. Long Courtship

(a) Run the traditional propose-and-reject algorithm on the following example:

Man	Preference List
1	$A > B > C > D$
2	$B > C > A > D$
3	$C > A > B > D$
4	$A > B > C > D$

Woman	Preference List
A	$2 > 3 > 4 > 1$
B	$3 > 4 > 1 > 2$
C	$4 > 1 > 2 > 3$
D	$1 > 2 > 3 > 4$

(b) We know from the notes that the propose-and-reject algorithm must terminate after at most n^2 proposals. Prove a sharper bound showing that the algorithm must terminate after at most $n(n-1)+1$ proposals. Is this instance a worst-case instance for $n = 4$? How many days does the algorithm take on this instance?

7. Better Off Alone

In the stable marriage problem, suppose that some men and women have standards and would not just settle for anyone. In other words, in addition to the preference orderings they have, they prefer being alone to being with some of the lower-ranked individuals (in their own preference list). A pairing could ultimately have to be partial, i.e., some individuals would remain single.

The notion of stability here should be adjusted a little bit. A pairing is stable if

- there is no paired individual who prefers being single over being with his/her current partner,
- there is no paired man and paired woman that would both prefer to be with each other over their current partners, and
- there is no single man and single woman that would both prefer to be with each other over being single.

(a) Prove that a stable pairing still exists in the case where we allow single individuals. You can approach this by introducing imaginary mates that people "marry" if they are single. How should you adjust the preference lists of people, including those of the newly introduced imaginary ones for this to work?

(b) As you saw in the lecture, we may have different stable pairings. But interestingly, if a person remains single in one stable pairing, s/he must remain single in any other stable pairing as well (there really is no hope for some people!). Prove this fact by contradiction.

8. Write Your Own Problem

Write your own problem related to this week's material and solve it. You may still work in groups to brainstorm problems, but each student should submit a unique problem. What is the problem? How to formulate it? How to solve it? What is the solution?