Tutorial I

Outline

- Stable Marriage problem
- Assignment 1

Stable Marriage problem

- There are N men (denoted by capital letters) and N women (denoted by lower case letters)
- Each person has a preference list of the opposite sex
- A marriage is 1-1 correspondence between men and women
- An unstable marriage is that if there exist 2 marriage couples X-x and Y-y such that X desires y more than x and y desires X more than Y ➔ In this case, we say X-y is dissatisfied

Stable Marriage problem (con't)

Ex. N=4

- men

  A:abcd B:bacd C:adcb D:dcab

- women

  a:ABCD b:DCBA c:ABCD d:CDAB

- There is a unstable marriage:

  A-a, B-b, C-c, D-d

- C-d is dissatisfied

Stable Marriage problem (con't)

- Algorithm to stable marriage:
  - Men are numbered in some arbitrary order

1. The lowest numbered unmarried man X proposes to the most desirable woman x on his list who has not rejected him
2. The women x will accept the proposal if she is currently unmarried, or if her current mate Y is less desirable to her than X (Y reverts to the unmarried state)
3. Repeats 1 and 2, terminating when every person has married

Stable Marriage problem (con't)

- Another example:
  - A:abcd B:bacd C:adcb D:dcab
  - a:ABCD b:DCBA c:ABCD d:DCAB
  - Let the order of the men: ABCD
    - A proposes to a => A-a
    - B proposes to b => B-b
    - C proposes to a => a rejects C
    - C proposes to d => C-d
    - D proposes to d => d jilt C, D-d, C is single
    - C proposes to c => C-c
    - A-a, B-b, C-c, D-d

Stable Marriage problem (con’t)

- Dose the algorithm terminate?
  - Yes. At most n^2 proposals … [why?]
- The final marriage is stable?
  - Yes. Proof by contradiction.
    1. Let X-y be a dissatisfied pair. In the marriage, they are paired as X-x, Y-y
    2. Since X prefers y to x, he must have proposed to y before getting married to x
    3. So, y either “rejected X” or “accepted X and then jilt him later”. This implies that her mate must be more desirable to her than X (contradiction)

Stable Marriage problem (con't)

- Worst case of the algorithm: $O(n^2)$ proposals
- How to analysis the average case of the deterministic algorithm?
  - We assume that the men's lists are chosen independently and uniformly; the women's lists can be arbitrary but must be fixed in advance.
  - T_p is the number of proposals made during the execution of the algorithm
  - The running time is proportional to T_p.

  => difficult to analysis T_p

Stable Marriage problem (con't)

- By deferred decisions:
  - The entire set of random choices (of the men's lists) is not made in advance
  - At each step of the proposal, we fix only the random choices that must be revealed to the algorithm

Stable Marriage problem (cont'd)

- Original algorithm (with deferred decisions):

  When making a proposal, the man picks a random woman from those he has not proposed before

  - That is, sampling without replacement
- Simplified algorithm:

  When making a proposal, the man picks a random woman from all women

  - That is, sampling with replacement
- The outputs are the same
  - Reason: In simplified algorithm, once a man was rejected by a woman, if he propose to the same woman afterwards, he will still be rejected

Stable Marriage problem (con't)

- T_p: the number of proposals made by the original algorithm
- T_a: the number of proposals made by the simplified algorithm
- $T_p > m \Rightarrow T_a \geq T_p > m$
- So, $\Pr(T_p > m) \leq \Pr(T_a > m)$ for all m
  - we can make use of the above observation to find an upper bound of T_p

Stable Marriage problem (con’t)

- T_a: the number of proposal made (each proposal is made uniformly and independently to one of n women)
- The algorithm terminates once all women have received at least one proposal ➔ coupon collector’s problem
- Thus, expected number of proposals = O(n lg n)

Assignment 1 – problem 1

- The proof of principle of inclusion-exclusion

$$
\begin{aligned}
Pr\left(\bigcup_{i=1}^{n} E_i\right) &= \sum_{i=1}^{n} \Pr(E_i) - \sum_{i<j} \Pr(E_i \cap E_j) + \sum_{i<j<k} \Pr(E_i \cap E_j \cap E_k) \\
&\quad - \cdots + (-1)^{\ell+1} \sum_{i_1<i_2<\cdots<i_\ell} \Pr\left(\bigcap_{r=1}^{\ell} E_{i_r}\right) \\
&\quad + \cdots + (-1)^n \Pr\left(\bigcap_{i=1}^{n} E_i\right).
\end{aligned}
$$

- Hint: by induction

Assignment 1 – problem 2

- the values of F are stored in a lookup table, 1/5 of the lookup table entries are changed
- $F((x + y) \bmod n) = (F(x) + F(y)) \bmod m$
- Give input z, F(z)?
- Hint: If F(z) is changed, you never get correct answer. You can use the above formula.

Assignment 1 – problem 3

- Describe a randomized algorithm for finding an r-cut with minimum number of edges.
- Hint: r-cut is a general case of 2-cut.

Assignment 1 – problem 4

- The expected number of fixed points ($\pi(x)=x$) in permutation π

- Hint: Try to come up with a simple way to analyze the expected number

Assignment 1 – problem 5

- Interview problem:
  - First interview m candidates but reject them all
  - From the (m+1)th candidate onwards, hire the first candidate who is better than all of the previous candidates you have interviewed
- Hint: E_i be the event that the i th candidate is the best and we hire him