The Communication Complexity of Finding a Stable Marriage

A Tale of Passion and Greed

Will Rosenbaum[1] Rafail Ostrovsky[2]

[1]Department of Mathematics
UCLA

[2]Department of Comupter Science
UCLA

Spring 2014

Contents

Contents

The Stable Marriage Problem

- n men and n women wish to form a set of n couples (i.e. a matching)
- matching should have the property that no players have an incentive to divorce (stability)
- each player (privately) holds preferences of players of the opposite gender
- stable matchings have property that no pair of unmatched players mutually prefer each other to their assigned partners

Preferences and Matchings

Denote sets of men and women by

$$Y = \{m_1, m_2, \ldots, m_n\} \quad \text{and} \quad X = \{w_1, w_2, \ldots, w_n\}.$$

Definition

For $p \in Y \cup X$, a **preference for** p is a linear order $<_p$ on players of opposite gender. For $w \in X$, w **prefers** m to m' if

$$m <_w m'.$$

An **instance of the stable marriage problem** is a triple (Y, X, P) where P is a set of a preference for each $p \in Y \cup X$.

Definition

A **matching** is a bijection

$$M : Y \leftrightarrow X.$$

We will often associate M with its graph: $M \subset Y \times X$.

Stable Matchings

Definition

Let (Y, X, P) be an instance of the stable marriage problem and M a matching, and $(m, w) \in Y \times X$. We say (m, w) is a **blocking pair** if $(m, w'), (m', w) \in M$ with $m \neq m'$, $w \neq w'$ and

$$w <_m w' \quad \text{and} \quad m <_w m'.$$

That is, m and w are not paired in M but mutually prefer each other to their assigned matches.

Definition

A matching M is **stable** (with respect to P) if it contains no blocking pairs.

Gale-Shapley (GS) Algorithm

Do stable matchings always exist for a given P?
Yes! Gale-Shapley (GS) Algorithm [Gale & Shapley, 1962]:

1. each man proposes to most preferred woman
2. women receiving proposals reject all but most preferred
3. rejected men propose to next most preferred
4. repeat steps 2 and 3 until no new proposals are made

Theorem (GS62)

The matching formed by un-rejected proposals in the GS algorithm is stable.

Stable Matching Example

For $n = 3$, men and women's preferences given by

$$\begin{array}{llll} m_1: & 1 & 2 & 3 \\ m_2: & 1 & 3 & 2 \\ m_3: & 2 & 1 & 3 \end{array} \qquad \begin{array}{llll} w_1: & 1 & 3 & 2 \\ w_2: & 1 & 2 & 3 \\ w_3: & 3 & 2 & 1 \end{array}$$

GS Algorithm produces the matching

$$M = \{(m_1, w_1), (m_2, w_3), (m_3, w_2)\}$$

What if women propose to men? Then GS produces

$$M' = \{(m_1, w_1), (m_2, w_2), (m_3, w_3)\} .$$

There can be many stable matchings!

Efficiency of GS Algorithm

- GS algorithm terminates after at most n^2 proposals
- can take $n^2 - \mathcal{O}(n)$ "rounds" of proposals to terminate
- order of proposals/rejections reveals a player's preferences
- in worst case, almost all players' preferences are revealed

Question

Is it possible to find stable matchings without revealing (most) of the players' preferences in the worst case?

To answer this question, we require tools from communication complexity...

Contents

Communication Complexity

Context:

- two players, Alice and Bob, wish to compute $f : X \times Y \to \{0, 1\}$
- Alice knows $x \in X$, Bob knows $y \in Y$
- both players can compute f from its input
- how much must Alice and Bob communicate to determine $f(x, y)$?

Communication Protocols

First formalized by Yao in 1979
(Deterministic) communication protocol Π consists of

- rooted binary tree
- internal vertices labeled by a player (Alice or Bob) and a boolean function of that player's input
- edges labeled by a value in $\{0, 1\}$
- leaves labeled by a value in $\{0, 1\}$

Communication Protocols

Alice and Bob execute the protocol by traversing the tree from root to leaf:

- if vertex v is labeled (p_v, f_v), player p_v computes f_v on their input z
- p_v announces value of $f_v(z)$ to other player
- both players follow the edge labeled by $f_v(z)$
- continue until reaching a leaf; value of leaf is output of protocol

Labels of edges traversed in this manner on input (x, y) is the **transcript** of Π on input (x, y). The transcript is the only information shared by Alice and Bob.

Cost and Complexity

Definition

The **communication cost** of a protocol Π, denoted $\mathrm{CC}(\Pi)$ is the depth of the associated tree, i.e., the length of the longest simple path from root to leaf.

Definition

The **communication complexity** of a function $f : X \times Y \to \{0, 1\}$, denoted $D(f)$, is the minimum cost among all protocols which compute f.

Randomized Protocols

- fix a probability distribution μ on set of all protocols
- Alice and Bob jointly pick Π from this distribution
- execute Π on their input
- requirement: for all $(x, y) \in X \times Y$

$$\mathbf{P}_\mu(\Pi(x, y) = f(x, y)) \geq 1 - \delta$$

for (fixed) $\delta < 1/2$.

Definition

The **communication cost** of a randomized protocol μ is

$$\mathrm{CC}(\mu) = \max\left\{\mathrm{CC}(\Pi) \,|\, \Pi \in \operatorname{supp}\mu\right\}.$$

The **randomized communication complexity** of f is

$$R_\delta(f) = \min\left\{\mathrm{CC}(\mu) \,|\, \mu \text{ computes } f\right\}$$

Disjointness Function

Example

- Alice and Bob hold $A, B \subset [n]$ respectively
- define
$$\mathrm{DISJ}(A,B) = \begin{cases} 1 & A \cap B = \varnothing \\ 0 & A \cap B \neq \varnothing. \end{cases}$$
- equivalently, x, y are characteristic functions of A and B,
$$\mathrm{DISJ}(x,y) = \neg \bigvee_{i=1}^{n} (x_i \wedge y_i)$$
- What is $R_\delta(\mathrm{DISJ})$?

Disjointness Lower Bound

Theorem (Razborov, 1992)

For any $\delta < 1/2$, $R_\delta(\mathrm{DISJ}) = \Omega(n)$. This bound holds even if we assume that the input sets are assumed to be either disjoint or uniquely intersecting.

- used as the "canonical" hard communication problem
- we will use DISJ to show that the stable marriage problem requires a lot of communication

Contents

Communication Complexity and Stable Marriages

First studied by Segal in 2007:

Theorem (Segal, 2007)

Any deterministic or nondeterministic protocol Π which for any preference structure P computes a stable matching requires $\Omega(n^2)$ total communication between the players.

- proof idea: "fooling set" method
- construct a large family of preferences with unique stable matchings
- find a large set $S \subset X \times Y$ of inputs such that for

  $$(x, y) \neq (x', y') \in S$$

  one of the following holds
  1. $f(x, y) \neq f(x', y')$
  2. $f(x', y) \neq f(x, y)$
  3. $f(x, y') \neq f(x, y)$.

Question

Does this lower bound still hold for randomized protocols?

Communication of Approximate Stable Matchings

Introduced by Chou & Lu 2010:
What if the output matching is allowed to have few blocking pairs?

- measure instability by fraction of "unstable partners," i.e., fraction of players involved in blocking pairs
- distributed input: each player knows only their own preferences
- centralized (deterministic) computation: players communicate only with a central server
- "sketch model"

Communication of Approximate Stable Matchings

Theorem (Chou & Lu 2010)

Any algorithm which finds an ε-stable matching for all preference structures requires $\varepsilon^2 n^2 \log(n\varepsilon)$ bits of communication.

Proof idea: show that if stable players send less than ε fraction of input, then there exists inputs with same message transcript but different stable matchings.

Question

Is it possible to obtain a similar lower bound for distributed computation?

Contents

Main Result for Stable Matchings

Theorem (OR, 2014)

Any protocol (deterministic, nondeterministic, or randomized) that for any set of preferences finds a stable matching requires $\Omega(n^2)$ bits of communication between the men and women.

- generalizes Segal's result to randomized protocols
- only measures communication between men and women

Proof idea: embed large instance of DISJ into preferences

General case strategy

- break men and women up into two groups: passionate and pragmatic
- passionate players either love or despise one another
- pragmatic players prefer passionate players to pragmatists
- embed DISJ into the passionate players preferences in such a way that there is "mutual affection" if and only if embedded sets intersect
- true love (i.e. set intersection) will make itself known in the stable matching

The Case $n = 2$

- embed DISJ of size $n/2 = 1$ (i.e., "nand" function)
- men and women hold $y, x \in \{0, 1\}$ respectively
- men's preferences for $y = 0$ and $y = 1$ are

$$\begin{array}{lll} m_1: & 2 & 1 \\ m_2: & 1 & 2 \end{array} \quad \text{and} \quad \begin{array}{lll} m_1: & 1 & 2 \\ m_2: & 1 & 2 \end{array}$$

  respectively
- women's preferences are analogous

Lemma

For the preferences described above, the unique stable matchings are given by

$$\begin{cases} \{(m_1, w_2), (m_2, w_1)\} & \text{if } \neg(x \wedge y) = 1 \\ \{(m_1, w_1), (m_2, w_2)\} & \text{if } \neg(x \wedge y) = 0 \end{cases}$$

General case $n = 2k$

- men and women hold

$$y, x \in \{0,1\}^{k^2}$$

  indexed by $(i,j) \in [k]^2$
- men's preferences determined by parity
- even men $m = m_{2i}$ have preferences

$$m_{2i}: \quad 1 \quad 3 \quad \cdots \quad 2k-1 \quad 2 \quad 4 \quad \cdots \quad 2k$$

- odd men m_{2i-1} have preferences determined by y:
  1. odd women w_{2j-1} with $y_{i,j} = 1$
  2. even women
  3. odd women w_{2j-1} with $y_{i,j} = 0$
- women's preferences analogous

General case $n = 2k$

Lemma

For the preferences described in the previous slide, a stable matching contains an odd couple (i.e., (m_{2i-1}, w_{2j-1})) if and only if $\mathrm{DISJ}(x, y) = 0$.

Proof idea:

- if $\mathrm{DISJ}(x, y) = 1$ then any odd couple would block a preferred mixed couple
- if $\mathrm{DISJ}(x, y) = 0$ with $x_{ij} = y_{ij} = 1$ and $(m_{2i-1}, w_{2j-1}) \notin M$, then m_{2i-1} or w_{2j-1} must have an odd partner.

Corollary

Any protocol for stable marriage with $2k$ players and B bits of communication between men and women can be used to solve any instance DISJ with inputs of size k^2 using B bits of communication.

Proof.

Use preferences described above. Simulate SM protocol. Look for an odd couple. □

Contents

Notions of Approximate Stable Matchings

1. count number of blocking pairs d^{bp} (normalized by n^2)
2. count number of unstable couples d^{uc} (normalized by n)
3. define **divorce distance** between any pair of matchings

   $$d(M, M') = n - \left|M \cap M'\right|$$

   take **distance to stability**

   $$d(M) = \min\left\{d(M, M') \,\middle|\, M' \text{ is stable}\right\}$$

   normalized by n.

3 is particularly appealing because $d(\cdot, \cdot)$ is a metric on the symmetric group

Definition

A matching M is ε-**unstable** or $(1-\varepsilon)$-**stable** with respect to preferences P if

$$d(M) \leq \varepsilon n.$$

Computing Distance to Stability

- set of stable matchings may be exponentially large
- but it is possible to compute d in polynomial time
- set of stable matchings is distributed lattice
- can efficiently find compact representation from preferences (see Gusfield & Irving, 1989 for details)
- can compute d via linear programming or “maximum closure” algorithm

Contents

Approximation Lower Bounds

Theorem (OR, 2014)

Suppose a (randomized) protocol Π produces a $(1 - \varepsilon)$ stable matching for any instance of the stable matching problem with probability at least $2/3$ and $0 < \varepsilon < \frac{1}{2}$. Then Π requires $\Omega(n^2)$ communication between the men and women.

- much less restrictive computational model than Chou & Lu...
- ...but finer notion of approximation

Proof idea: embed instance of disjointness of size $\Omega(n^2)$ into preferences.

Disjointness Embedding Strategy

- similar ideas as exact computation strategy
- embed DISJ into subset of players' preferences such that set intersection forces a particular couple to appear in stable matching
- choose remaining preferences to detect/amplify the existence of a couple from set intersection
- set intersection forces other couples to work around fixed couple
- true love can move mountains

Disjointness Embedding for ASM

- assume $n = 2k$, $0 < \delta < 1$ a parameter
- break men and women into three groups:
  - X_ℓ (low) with $|X_\ell| = k$
  - X_m (mid) with $|X_m| = (1-\delta)k$
  - X_h (high) with $|X_h| = \delta k$

  (similar for men)
- embed instance of DISJ of size $\delta^2 k^2$ into preferences of high players, $x, y \in \{0,1\}^{\delta^2 k^2}$
- low and mid preferences fixed

Disjointness Embedding for ASM

- low preferences: $w_j \in X_\ell$ $(j > k)$

$$w_i : \quad 1 \quad 2 \quad \cdots \quad n$$

- mid preferences: $w_j \in X_m$ $(\delta k < j \leq k)$

$$w_i : \quad k+1 \quad k+2 \quad \cdots \quad 2k \quad 1 \quad 2 \quad \cdots \quad k$$

- high preferences: $w_j \in X_m$ $(j \leq \delta k)$
  1. high men m_i, $i \leq \delta k$ such that $x_{ij} = 1$ (in order)
  2. low men m_i, $i > k$ (in order)
  3. mid men m_i, $\delta k < i \leq k$ (in order)
  4. high men m_i, $i \leq \delta k$ such that $x_{ij} = 0$ (in order)

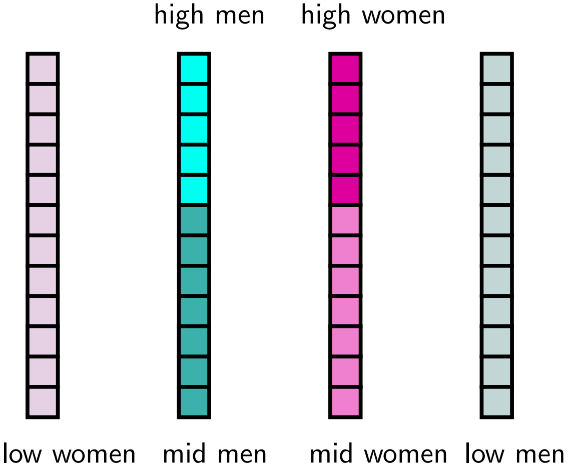
Lower Bound Lemmas

Lemma

Suppose M_1 is a stable matching for a disjoint instance of DISJ. Then

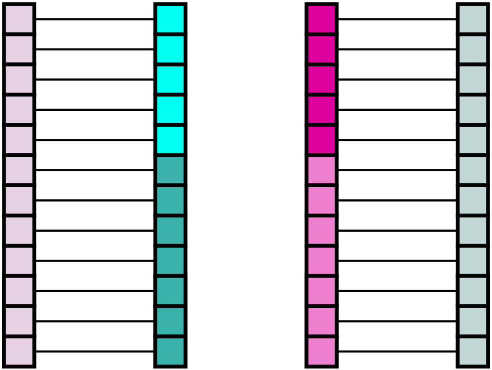
$$M_1 = \{(m_1, w_{k+1}), \ldots, (m_k, w_{2k})\} \cup \{(m_{k+1}, w_1), \ldots, (m_{2k}, w_k)\}.$$

Lower Bound Lemmas



Lower Bound Lemmas

stable matching structure for disjoint instances



Lower Bound Lemmas

Lemma

If M_0 is a stable matching for a uniquely intersecting instance of DISJ, then

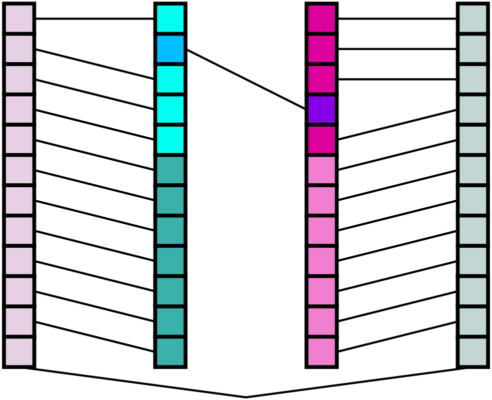
$$d(M_0, M_1) \geq (1-\delta)k$$

Proof idea:

- if $x_{ij} = y_{ij} = 1$, then $(m_i, w_j) \in M_0$
- remaining matches are as with M_1, except men m_s with $i < s \leq k$ and women w_t with $j < t \leq k$ shifted by one
- no mid men or women have the same partner in M_0 and M_1 (see figure)
- thus at least $(1-\delta)$-fraction of players have different partners

Lower Bound Lemmas

stable matching structure for intersecting instances



Lower Bound Proof

Combining previous lemmas gives proof of Theorem 2:

- suppose Π finds $(1-\varepsilon)$-stable matching with probability $2/3$ using B bits of communication where $\varepsilon < (1-\delta)/2$
- use preference structure above to embed DISJ instance
- let M be matching output by Π
- then

$$d(M, M_0) \leq \varepsilon n < (1-\delta)/2 \iff \mathrm{DISJ}(x, y) = 0$$

- thus a B-bit ASM protocol $\implies$ B-bit DISJ protocol of size $\delta^2 k^2 = \Theta(n^2)$.

Testing ASM is Hard

Corollary

Any protocol which given a matching M and preferences P determines if M is $(1-\varepsilon)$-stable with respect to P requires $\Omega(n^2)$ communication.

Proof.

- take $M = M_0$ and preferences from embedded disjointness
- M is $(1-\varepsilon)$-stable $\iff \mathrm{DISJ}(x, y) = 0$

$\square$

Remark

This is not true for d^{bp}-stability: sample a constant number of pairs to see if they are blocking.

Contents

Future Work

- communication lower bound for coarser notion of approximate, e.g., d^{bp}
- “round complexity” for fully distributed computation
- round efficient algorithms for distributed computation

Thank You!