An instance of distributed social computation: the multi-agent group membership problem

Lorenzo Coviello, *Student Member, IEEE,* and Massimo Franceschetti, *Member, IEEE,*

***Abstract*—We consider a scenario in which agents in a network are rewarded if they collectively solve a group membership task from only local distributed interactions. We propose a simple distributed algorithm with a number of desirable features aimed at modeling distributed social computation: the algorithm is self-stabilizing, decisions are made using only local information, the exchanged messages are minimal and can be represented by a single bit, agents pursue local stability, and have no memory of the past. Using this model, we characterize mathematically the trade-off between the quality of a solution and the time needed to reach it, proving that a *good* solution is always found quickly while improving it to the *optimum* might require excessive time.**

We also present laboratory experiments where a group of human subjects were rewarded if they solved the same group membership task. We observe a good fit between the humans' performance and our algorithm's predictions, showing that the global dynamics of agents with diverse strategies can be described by simple, synthetic agents with a uniform strategy of interaction. This suggests that simple models of distributed computation can be constructed to predict the aggregate performance of real populations of humans solving computational problems over networks, addressing a question recently posed by Kearns (2012).

***Index Terms*—Social computation, group membership, distributed algorithms.**

I. Introduction

We consider a distributed computation scenario in which there are agents of two types, *leaders* and *followers*. Each leader is equipped with the task to form a group of followers of a certain cardinality, by sending them requests. Followers can either accept or reject incoming leaders' requests. Each follower can be part of a single group at any time but can change group over time. Multiple followers can be part of a leader's group, but each leader can only recruit followers with whom it shares a communication link. These communication links are described by an arbitrary bipartite network, and we assume that each agent has knowledge of, and can interact with its neighbors over the network. In practice, the structure of the network can be dictated by physical or social constraints, see Figure 1. Leaders and followers share the *common goal* of reaching a state in which each leader formed a group of the right size, and we call *stable* such a state of "social welfare." We refer to this scenario as the *group membership problem.*

The contribution of the present work is twofold. First, we show that simple local rules of interaction lead to stable, or close to stable, group membership in reasonable time, where by "close to stable" we mean that the total number of additional followers required to satisfy all group size constraints is an arbitrary small fraction of the entire population. Then, we show that such rules can predict the performance of a group of human subjects solving the same group membership task in a laboratory setting, suggesting that simple, synthetic, multi-agent systems are useful to describe the computational performance of a complex, real, heterogeneous human population subject to information constraints.

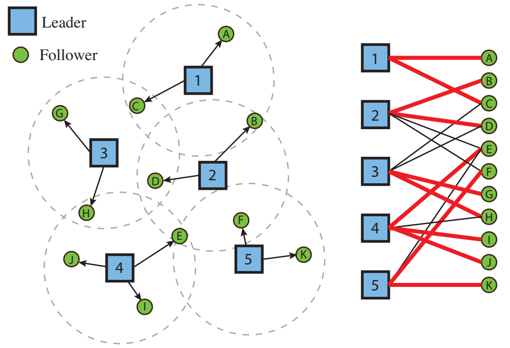

Fig. 1. **Example of a bipartite network between leaders and followers determined by physical constraints.** Left: each leader can recruit the followers in its visibility range (dotted circle), arrows represent group membership, the set of arrows defines a partition of the followers into groups. Right: the resulting bipartite network. An edge between leader ℓ and follower f exists if and only if f is in ℓ's visibility range. Matching edges are highlighted.

Regarding the first contribution, we propose a simple, distributed, memoryless algorithm in which leaders only pursue local stability, and we show that, in any network of size n, any constant approximation of a globally stable outcome (or of a suitably defined *best* outcome if a stable one does not exist) is reached in time polynomial in n with high probability. In other words, within an acceptable approximation, our algorithm is able to find a solution in feasible time on any instance of the problem. In contrast, we show that there exist networks requiring an exponential gap between the time needed to reach stability and that needed to reach approximate stability, that is, to find the *best* solution compared to a *good* solution.

Regarding the second contribution, we created an artificial environment in which human subjects have to solve a group membership task on virtual networks of leaders and followers. We conducted 36 experiments of group membership on a pool of 10 different networks with 16 nodes each. In each experiment, participants controlled the nodes of a virtual network and interacted with their neighbors via the point-and-click interface shown in Fig. 2. In order to elicit the common goal of reaching stability, they received a monetary reward if they reached a stable state within a maximum time of 5 minutes. We observe a good fit between experimental data and

L. Covello and M. Franceschetti are with the Department of Electrical and Computer Engineering, University of California San Diego, CA, 92093, USA e-mail: lcoviell@ucsd.edu. Work partially supported by ARO Award No. W911NF-11-1-0363.

Manuscript received XX; revised XX

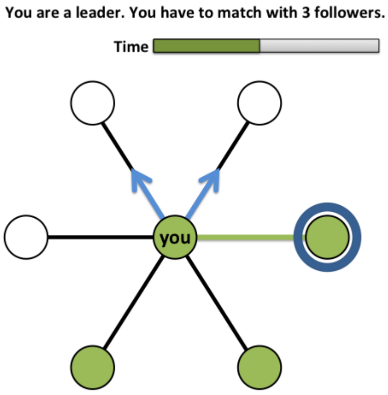

Fig. 2. **Computer interface of a participant playing the role of a leader.** The participant controls the node in the middle (with the label "you") and has to match with 3 followers (neighboring nodes). The green line indicates a matched pair with the follower on the right. Followers marked in green (on the bottom) are currently matched with other leaders. Blue arrows represent outgoing matching requests (to the followers on top). The bar on top shows the remaining time (out of the 5 minutes allowed).

the algorithm's predictions. On the one hand, the algorithm was able to predict which networks where the most difficult to solve by the human subjects. On the other hand, the human subjects always found good solutions quickly and spent most of the time attempting to improve them to the optimum. These results suggest that, at least in the specific context of the group membership problem considered here, simple local rules of interaction are able to simulate complex global dynamics, and therefore tools from traditional computation theory can be used to study distributed social computation.

We point out that the idea of using simple interactions to predict global outcomes resulting from possibly complex and diverse microscopic effects is not new. The theme is recurrent in statistical physics and cellular automata [1]–[4], but has yet to gain popularity in the context of social computation. Our reduction of social interaction to algorithmic modeling is somehow reminiscent of the work of Herbert Simon, who claimed that information processing is at the basis of human decision-making [5]. Finally, we refer to the influential work of Kearns and his collaborators [6]–[11], who pointed out the need to study the principal mechanisms of social computation and strategic interaction over networks. Our experimental approach, based on a highly constrained laboratory setting, where language and other natural forms of communication are eliminated in favor of enforcing simple actions, follows closely this line of work. In addition, we advocate for simple computational models of individual behavior for predictive and explanatory purposes.

II. Related work

The study of distributed multi-agent coordination has received a great deal of attention by the control and computer science communities, particularly in settings where the agents perform simple local updates, do not have complete knowledge of the entire population, and communication between them is limited. On the other hand, in computational social science similar coordination problems have been considered with the goal of providing a model of distributed dynamics of human networks.

Within the first group of studies, one of the main issues is whether distributed multi-agent coordination dynamics converge in finite time to a set of desirable configurations defined by notions of stability and optimality. For example, Roth and Vande Vate [12] considered two-sided marriage markets and showed that better- and best-response dynamics always reach stability in finite time. Bertsekas and Castanon [13]–[15] studied distributed dynamics for the assignment problem and proved their converge with probability one to optimal assignments. In our work, convergence to the set of optimal solutions always occurs in any instance of the problem. In particular, we define a potential function for our algorithm (that we call *deficit*) and a structural result guarantees that this decreases in a finite number of iterations (see Lemma 1).

Beside convergence in finite time, another important issue is convergence in *feasible* time. In this case, results are based on computer simulation [15], [16], or require specific modeling assumptions. For example, in the context of two-sided matching, best- or better-response dynamics are shown to converge to stability in polynomial time in the cases of global rankings [17], correlated markets [18] and geometric preferences [19], but might require exponential time in the general case (see for example [18]). In the context of distributed network coloring, Vattani et al. [20] proposed simple local dynamics that converge in polynomial time on any bipartite network. These works provide relevant insights about certain classes of problems, but their analysis is restricted by the specific assumptions they make to provide convergence guarantees.

An alternative approach, that allows to obtain results in more general scenarios, is to introduce a notion of *approximate* solution and to quantify the tradeoff between the quality of a solution and the time needed to reach it. The objective is to provide provable performance guarantees that hold for any instance of the problem, and to understand what types of configuration can be reached in practical time. In this context, Nedic and Ozdaglar [21] studied the distributed optimization of sums of convex functions by agents who optimize their local objectives and exchange information locally, focusing on the tradeoff between solution accuracy and time. In the same spirit, our work considers a setup in which a globally stable solution is approximated by agents concerned by their local stability. In the context of bargaining over social exchange networks, Kanoria et al. [22] showed that ε-approximate Nash bargaining solutions are reached by a simple distributed algorithm in time polynomial in ε^{-1} and in the network size. Coviello et al. [23] considered the problem of maximum matching and showed that a ε-approximate maximum matching is reached by simple local interaction in time increasing in ε^{-1} and polynomial in the network size. In the context of distributed consensus and averaging, Olshevsky and Tsitsiklis [24] showed that the number of iterations to convergence is polynomial in the number of agents and increases only logarithmically in the target accuracy ε^{-1} (where ε is the maximum allowed distance of an agent's opinion from consensus). Nax et al. [25] considered a formulation of the assignment problem in which limited information is available and can be exchanged between the agents, and proposed a simple distributed scheme for the agents' local updates. While the proposed algorithm is shown to always converges to optimal and stable allocations, the authors do not study its rate of convergence.

A different aspect of our work is the proposal of a simple distributed dynamical model to describe and predict the outcomes of groups of humans who have to coordinate over networks.

In computer science, human coordination has been studied under the premise that coordination constitutes the basis for *social computing* [26]. Following this approach, distributed collections of humans are tried to collectively solve traditional algorithmic tasks, such as coloring, consensus, and various forms of matching. For example, in the work of Kearns et al. [6], human subjects positioned at the vertices of a virtual network were shown to be able to collectively reach a coloring of the network, given only local information about their neighbors. Other works further investigated human coordination in the case of coloring [9], [27], [28], consensus [8], [9], matching [23], bargaining and trade [7], [22], [29], and network formation [10]. Quoting Kearns [11], the main findings of research on experimental social computation up to date are the ability of humans to solve a wide range of tasks in a distributed fashion, the effect of the network structure on performance, with opposite effects for different tasks [9], and the emergence of behavioral characteristics of individuals [8]. However, the effectiveness of mathematical models of social computation to predict performance still needs to be assessed.

In this work, we address the question posed in [11] regarding the possibility of using simple models of social computation to predict the performance of humans on specific computation tasks over networks. Within an extremely wide design space, we focus on the distributed task of group membership, and use computational complexity and equilibrium concepts as the rigorous language to express these predictions.

III. The group membership task

We consider a bipartite network $G = (L \cup F, E)$ whose nodes are the disjoint sets L of leaders and F of followers, and where there exists an edge $(f, \ell) \in E$ between follower f and leader ℓ if and only if f and ℓ can communicate between each other (see Figure 1). Let $N_\ell = \{f \in F : (f, \ell) \in E\}$ be the neighborhood of $\ell \in L$. For each $\ell \in L$, leader ℓ has to form a group of c_ℓ followers from N_ℓ, where $c_\ell \geq 1$.

Definition 1 (Matching): A subset $M \subseteq E$ is a matching of G if for each $f \in F$ there exists at most a single $\ell \in L$ such that $(\ell, f) \in M$.

The definition of matching permits multiple followers to be part of a leader's group. There is a one-to-one correspondence between matchings M of G and groups $\{T_\ell(M) : \ell \in L\}$, where $T_\ell(M)$ denotes the group of leader ℓ under the matching M. We have that $T_\ell(M) = \{f \in F : (\ell, f) \in M\} \subseteq N_\ell$ for every matching M. Agents in $L \cup F$ are rewarded if each leader ℓ controls a team of c_ℓ followers, therefore we consider the following notion of stability.

Definition 2 (Stable matching): Given constraints c_ℓ for each $\ell \in L$, a matching M of G is stable if and only if $|T_\ell(M)| = c_\ell$ for each $\ell \in L$.

Having a local view of the network, each leader ℓ can only assess if "local stability" holds (i.e., if it is matched with c_ℓ followers), in contrast with the notion of "global stability" defined above.

Given the constraints c_ℓ, a network G might not admit a stable matching. Nonetheless, given a matching of G, we are interested in assessing its *quality*. Our main result builds on the following definitions of *deficit* of a leader and a matching.

Definition 3 (Deficit): Let ℓ be a leader with constraint c_ℓ, and M be a matching of G. The deficit of ℓ under M is

$$d_\ell(M) = c_\ell - |T_\ell(M)|.$$

The deficit of M is

$$d(M) = \sum_{\ell \in L} d_\ell(M) = \sum_{\ell \in L} \left(c_\ell - |T_\ell(M)|\right).$$

In words, $d_\ell(M)$ is the number of additional followers leader ℓ needs to satisfy its size constraint. Similarly, $d(M)$ sums the numbers of additional followers all leaders need to satisfy their size constraints. Given a matching M, we say that a leader ℓ is *poor* if $d_\ell(M) > 0$ (that is, $|T_\ell(M)| < c_\ell$) and *stable* if $|T_\ell(M)| = c_\ell$ (we exclude $|T_\ell(M)| > c_\ell$ assuming that matching with additional followers is costly). Observe that only poor leaders contribute to $d(M)$, and that M is stable if and only if $d(M) = 0$. Given G, two matchings can be compared with respect to their deficit, and the best matching of G can be defined as one minimizing the deficit.

Definition 4 (Best matching): A matching M of G is a best matching if $d(M) \leq d(M')$ for every matching M' of G.

Observe that a stable matching is also a best matching, and that a best matching always exists for any network G and constraints c_ℓ. Moreover, if G admits a stable matching, $d(M)$ quantifies how much M differs from a stable matching of G. In general, if M^* is a best matching of G with $d(M^*) = d^*$, then, $d(M) - d^*$ tells how much M differs from a best matching of G. Given a matching M of G, the following definitions provide a measure of how well M approximates a best matching of G, or a stable matching (if one exists).

Definition 5 (Approximate best matching): Fix $\varepsilon \in [0, 1]$, and let m be the number of followers in G. Let M^* be a best matching of G. Then, a matching M is a $(1-\varepsilon)$-approximate best matching of G if $d(M) - d(M^*) < \varepsilon m$.

Definition 6 (Approximate stable matching): Let G admit a stable matching. Fix $\varepsilon \in [0, 1]$, and let m be the number of followers in G. Then, a matching M is a $(1-\varepsilon)$-approximate stable matching of G if $d(M) < \varepsilon m$.

IV. The group membership algorithm

For ease of presentation, we assume that agents are synchronized. However, our results continue to hold also in the case of asynchronous agents (see discussion in Section VII). We assume that time is divided into rounds and each round is composed of two stages. In the first stage, each leader acts according to the algorithm in Table 1, and in the second stage each follower acts according to the algorithm in Table 2.

First consider a leader ℓ, and let M be the matching at the beginning of a given round. If ℓ is poor (that is, $|T_\ell(M)| < c_\ell$) and $|T_\ell(M)| < |N_\ell|$ (that is, ℓ is not already matched with all followers in N_ℓ) then, with probability p (where $p \in (0, 1]$ is a fixed constant), ℓ attempts to match with an additional follower. We assume that leaders always prefer followers that are currently unmatched over matched ones. Note that a leader first checks if *local stability* holds (i.e., its group size is c_ℓ).

Consider now a follower f. During each round, if f has incoming requests then each is rejected independently of the others with probability $1 - q$ (where $q \in (0, 1]$ is a fixed constant). If all incoming requests are rejected, then f does not change group (if currently matched) or it remains unmatched (if currently unmatched). Otherwise, one among the active requests is chosen uniformly at random, f matches with the corresponding leader, and all the other requests are discarded. For ease of presentation, we assume that a follower is equally likely to accept a request when unmatched or matched, and that p and q are the same for all agents. Our results hold for more general choices of the parameters, that can vary between agents, as long as they remain bounded away from zero[1].

The proposed algorithm has the following desirable features aimed at modeling distributed social computation: agents have no memory of the past, decision are based only on local information, it is *self-stabilizing* (i.e. it stops when a stable matching is reached), the exchanged messages can be represented by a single bit[2], and each leader only pursues local stability.

The single invariant of the algorithm is that leaders prefer unmatched followers and pursue local stability. Followers, on the other hand, act in a randomized fashion and ensure exploration of the state space. Despite their simplicity, these simple rules allow to reach a good approximate solution and capture the collective behavior of real human networks, as we will see in Section V and Section VI, respectively. As a remark, we consider an algorithm in which agents have no memory of the past for ease of analysis. We believe that allowing agents' decisions to depend on the past actions (made by them and their neighbors) would not change our results.

Table 1 Algorithm for leader $\ell \in L$

```
if $|T_\ell(M)| < \min\{c_\ell, |N_\ell|\}$ then
    with probability $p$ do the following
    if $\exists$ unmatched $f \in N_\ell$ then
        choose an unmatched follower $f' \in N_\ell$ u.a.r.
    else
        choose a follower $f' \in N_\ell \backslash T_\ell(M)$ u.a.r.
    end if
    send a matching request to $f'$
end if
```

Table 2 Algorithm for follower $f \in F$

```
if $f$ has incoming requests then
    for each leader $\ell$ requesting $f$ do
        with probability $1 - q$ reject $\ell$'s request
    end for
    if there are active requests then
        select one u.a.r. and join the corresponding team
        reject all other requests
    end if
end if
```

V. Complexity results

For ease of presentation, we only consider networks admitting stable matchings and show that, given any network and any constant $\varepsilon \in (0, 1)$, a $(1-\varepsilon)$-approximate stable matching is reached in a number of rounds that is polynomial in the network size with high probability (Theorem 1). Then, we show through a counterexample that improving from approximate stability to stability might require time exponentially large in the network size (Theorem 2). Our results hold in general for reaching approximate best matchings.

Theorem 1: Let G be a network with m followers and which admits a stable matching. Let $\Delta = \max_{\ell \in L} |N_\ell|$ be the maximum degree of the leaders. Fix $0 < \varepsilon < 1$, and let $c \geq 1 + \frac{1}{m(1-\varepsilon)}$. Then, a $(1 - \varepsilon)$-approximate stable matching of G is reached within $c\lfloor 1/\varepsilon \rfloor (\Delta/pq)^{\lfloor 1/\varepsilon \rfloor} m$ rounds of the algorithm with probability at least $1 - e^{-cm\varepsilon^2/2}$.

As an example, if Δ is constant in the network size, then one can choose $\varepsilon = 1/\log m$, and Theorem 1 implies that a $(1 - 1/\log m)$-approximate stable matching is reached in at most $\mathcal{O}(m^2 \log m)$ rounds with probability that goes to one as $m \to \infty$.

To prove Theorem 1 (see Appendix A) we introduce the notion of *deficit-decreasing* path, that in our setup plays the role of the augmenting path in the context of one-to-one matching. Since we consider bipartite networks, a path alternates leaders and followers.

Definition 7 (Deficit-decreasing path): Given a matching M of G, a cycle-free path $P = \ell_0, f_1, \ell_1, \ldots, f_{k-1}, \ell_{k-1}, f_k$ (of odd length 2k-1) is a deficit-decreasing path relative to M if $(\ell_i, f_i) \in M$ for all $1 \leq i \leq k - 1$, ℓ_0 is a poor leader, and f_k is an unmatched follower.

In other words, a deficit-decreasing path starts at a poor leader with an edge not in M, ends at a follower that is not matched, and alternates edges in M and edges not in M. Observe that a new matching M' such that $d(M') = d(M) - 1$ is obtained by flipping each unmatched edge of a deficit-decreasing path P into a matched edge, and vice versa. This is shown in Figure 3. The proof of Theorem 1 is based on a technical lemma (see Appendix B) that extends a previous combinatorial result by Hopcroft and Karp [30, Theorem 1]. Given a matching M with $d(M) \geq \varepsilon m$, we guarantee the existence of a deficit-decreasing path of length at most $2\lfloor 1/\varepsilon \rfloor$. Such a "short" path allows us to bound the number of rounds needed for a one-unit reduction of the deficit. The symmetric difference of two sets A and B is $A \oplus B = (A \backslash B) \cup (B \backslash A)$.

[1] If each leader ℓ has a parameter p_ℓ and each follower f has a parameter q_f, our main result in Theorem 1 holds with $p = \min_\ell p_\ell$ and $q = \min_f q_f$ as long as these lower bounds are bounded away from zero.

[2] For example, when a follower accepts a matching request by leader ℓ, she might communicate it by sending a bit '1' to leader ℓ and a bit '0' to all other neighboring leaders. Our main result (Theorem 1) does not consider the total number of bits exchanged to reach a given approximation of the optimal solution.

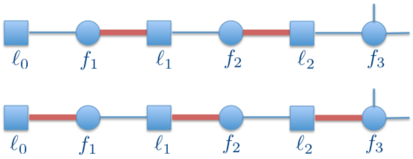

Fig. 3. **Deficit-decreasing path.** Top: a deficit-decreasing path of length 5: ℓ_0 is a poor leader, f_3 is an unmatched follower, and matching edges are highlighted. Bottom: the path is "solved" by turning each matched edge into an unmatched edge and vice versa; ℓ_0 obtains an additional follower (and its deficit decreases by one), ℓ_1 and ℓ_2 do not change their numbers of followers.

Two paths are *follower-disjoint* if they do not share any follower (even though they might share some leader).

Lemma 1: Let G admit a stable matching N. Let M be a matching of G with deficit $d(M) > 0$. Then, in $M \oplus N$ there are at least $d(M)$ follower-disjoint deficit-decreasing paths relative to M.

We make use of Lemma 1 through the following corollary, which holds as the lengths of a set of follower-disjoint paths sum to at most $2m$.

Corollary 1: Let G be a network with m followers, admitting a stable matching N. Let M be a matching of G with deficit $d(M) \geq \varepsilon m$, for some $\varepsilon > 0$. Then, in $M \oplus N$ there exists a deficit-decreasing path relative to M of length at most $2\lfloor 1/\varepsilon \rfloor - 1$.

As a remark, Corollary 1 and the observation that the deficit is non-increasing guarantee that our algorithm always convergences to the set of optimal solutions in finite time in any instance of the problem[3].

Theorem 1 gives a polynomial bound for reaching a $(1-\varepsilon)$-approximate stable matching for any constant $0 < \epsilon < 1$ and any network. However, a polynomial guarantee cannot be derived for the case of a stable matching (that is, for $\varepsilon = 1/m$). To show this, we define a sequence of networks in which the number of rounds required to converge from an approximate matching M with $d(M) = 1$ to the stable matching is exponentially large in the network's size with high probability from an overwhelming fraction of the approximate matchings M such that $d(M) = 1$.

For $n \geq 1$, let $G_n = (L_n \cup F_n, E_n)$ be the network with leaders $L_n = \{\ell_1, \dots, \ell_n\}$, followers $F_n = \{f_1, \dots, f_n\}$, edges $E_n = \{(\ell_i, f_j) : 1 \leq i \leq n, j \leq i\}$, and group size constraints $c_\ell = 1$ for all $\ell \in L_n$, see Figure 4. G_n has a unique stable matching given by $M_n^* = \{(\ell_i, f_i) : 1 \leq i \leq n\}$.

Theorem 2: For any matching M of G_n, let $\tau^*(M)$ denote the number of rounds to converge to the perfect matching when starting from M. Then, for any fixed constant $0 < \gamma < 1$, $\tau^*(M)$ is exponentially large in γn with high probability for a $1 - O(n2^{-(1-\gamma)n})$ fraction of all the matchings M such that $d(M) = 1$.

Here we only provide the idea of the proof, whose details are presented in Appendix C. To get an understanding of the algorithm's dynamics, consider the matching

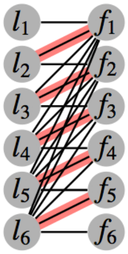

Fig. 4. **The network G_n for $n = 6$.** The matching M_n' is highlighted.

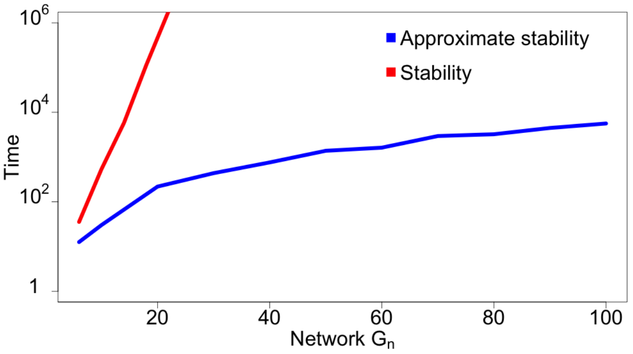

Fig. 5. **Algorithm's performance on the hard networks G_n.** The x-axis reports n, the y-axis reports the average time over 1000 simulations (in base-10 logarithm). The red line (top) shows the time to reach the stable matching; the blue line (bottom) shows the time to reach a $(1-\varepsilon)$-approximate matching for $\varepsilon = 0.1$.

$$M_n' = \{(\ell_i, f_{i-1}) : 2 \leq i \leq n\},$$

highlighted in Figure 4 for the case of $n = 6$. Observe that $d(M_n') = 1$ and ℓ_1 is poor. According to the algorithm, ℓ_1 tries to match with f_1. If f_1 accepts, then ℓ_2 becomes poor (and tries to match with f_1 or f_2). After each round, there exists a unique poor leader until the stable matching is reached. The stable matching is reached when ℓ_{n-1} (ℓ_5 in Figure 4) becomes poor and matches with f_{n-1} (f_5 in Figure 4), and finally ℓ_n matches with f_n. The stochastic process tracking the position of the poor leader is not a classical random walk and its transition probabilities at each time depend on the current matching. We show that convergence to stability requires a number of rounds that is exponential in n with high probability, and this holds for an overwhelming fraction of all matchings with $d(M) = 1$.

Fig. 5 shows the algorithm's average convergence time on the sequence of networks G_n (in logarithmic scale). The average number of rounds to reach a 0.9-approximate stable matching is upper bounded by a polynomial of small degree (bottom, blue line), consistently with Theorem 1, while convergence to the stable matching requires an average number of rounds that grows exponentially in n (top, red line), as predicted by Theorem 2. Fig. 6 shows the algorithm's performance in reaching successively finer approximations of the best matching on random networks $G(n, m, \rho)$. Here, $G(n, m, \rho)$ refers to a random bipartite network with n leaders and m followers, in which each edge exists independently of the others with probability ρ (we fixed $\rho = 0.04$), and with constraint $c_\ell = \min\{m/n, |N_\ell|\}$ for each leader ℓ. For each

[3] The deficit never increases over time as a leader never voluntarily disengages from a follower (see Table 1), and a follower disengages from a leader only when she accepts a new matching request (see Table 2). Convergence follows by considering the Markov chain whose state space is the set of all matchings. For each matching M, only matchings M' with $d(M') \leq d(M)$ can be reached from M, and Corollary 1 guarantees the existence of a finite sequence of transitions that lead from M to M' such that $d(M') < d(M)$ with finite probability.

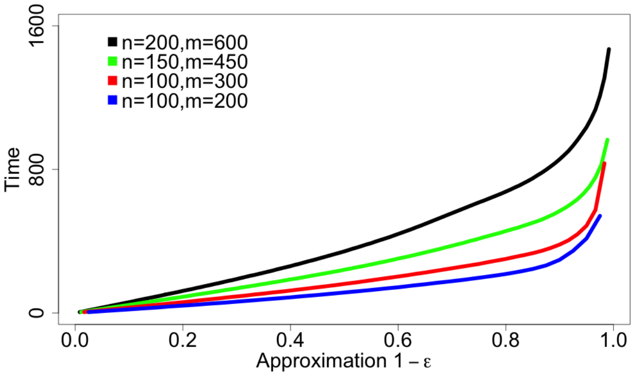

Fig. 6. **Algorithm's performance on random networks.** Algorithm's performance to reach a $(1-\varepsilon)$-approximate best matching on random bipartite networks $G(n, m, \rho)$, for $\rho = 0.04$ and different choices of n and m. For each choice of n and m, 100 random networks where generated, and each was simulated 100 times.

choice of n and m that we considered, 100 random $G(n, m, \rho)$ were generated, and the algorithm was run 100 times on each. We observe that, consistently with Theorem 1, $\tau(\varepsilon)$ increases both when ε decreases (i.e., a finer approximation is desired) and when the number m of followers increases. The plot shows that a good solution is reached quickly, while most of the time is spent in the attempt of improving it to the best solution.

VI. Experiments of human social computation

We conducted 36 experiments on a pool of 10 networks of 16 nodes each (each network was tested 3 or 4 times). Each of sixteen participants controls a node in the network via a computer interface which shows only its immediate neighbors (see Figure 2). During each experiment, a network is chosen and subjects are randomly assigned to nodes and informed whether they are playing the role of followers or leaders (in the latter case, the target number of followers is also specified). In order to elicit the common goal of reaching stability, each subject is paid a reward of $1 if stability is reached within the maximum time of 5 minutes. Subjects can only interact via the computer interface: leaders can send matching requests to followers and break them with clicks of the mouse (for each leader, the number of concurrent outgoing requests plus matched followers can be at most equal to its group target size); followers can accept or reject leaders' requests and break their own existing matched pairs, with clicks of the mouse.

The networks range from simple random topologies to topologies similar to the network G_n defined above (and are not shown due to space constraints). After the experiments, each network was assigned a networkID such that higher IDs correspond to higher average solving time (if an experiment is not solved within the 5 minutes maximum time, a time of 5 minutes is considered in the analyses). Figure 7 compares the performance of the human subjects (average number of seconds for each network, sorted by increasing solving time) and of the algorithm (average number of rounds over 10000 simulations on each network). Networks that required more time to be solved during the experiments also required more rounds of the algorithm (correlation 0.64 between number of seconds in the experiments and average number of rounds for the algorithm, p-value=0.04). Moreover, the networks with NetworkID from 8 to 10 (the most difficult to solve for the human subjects) are the topologies similar to the network G_n and were solved 6 times out 11 (all other experiments were solved). Our results do not seem to be determined by participants learning or getting tired (i.e., improving or worsening over time).[4]

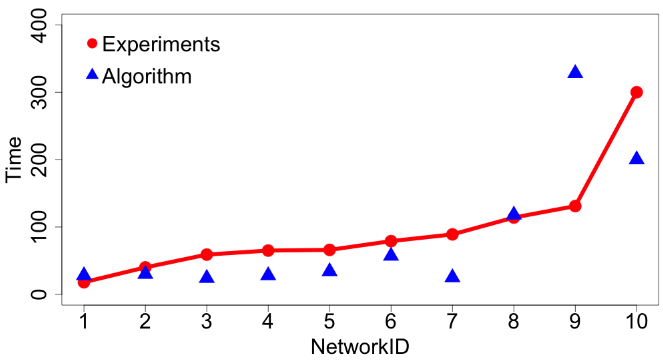

Fig. 7. **Algorithm performance versus human subjects performance.** The x-axis shows the NetworkID for the 10 different networks tested in the experiments, sorted by increasing average solving time in the experiments. The red line shows the average solving time (in seconds) for each network in the experiments (each tested three of four times); the blue triangles show the average number of rounds needed by the algorithm to solve the same networks (over 1000 simulations). The correlation between experiments' average time and algorithm's average number of rounds is 0.64 (p-value=0.04).

Figure 8 compares the time needed by the human subjects to reach a stable matching versus an approximate solution in which only a single leader needs an additional follower (that is, with deficit equal to 1, according to the definition given above). Experiments are sorted by increasing solving time and we observe that a good solution is reached quickly while most of the time is spent improving it to the optimum, in agreement with the algorithm predictions. The time to reach an approximate solution with deficit equal to 1 and a stable solution are correlated (0.55, p-value 0.0004), and on average reaching the approximate solution requires about 7% of the total time (least squares regression, 0.065 p-value 0.0005 without controlling for NetworkID, 0.073 p-value 0.005 controlling for NetworkID).

At the end of the experimental session, subjects were asked to complete an exit survey about their strategies. A wide range of strategies was reported. As for the leaders, participants reported to favor unmatched followers (7 surveys, notice how this criterion agrees with our algorithm), blink (that is, quickly sending and canceling requests) in order to capture the attention of a follower (3 surveys), only request followers who did not break the matching earlier (2 surveys), try to match with new followers if the game is not solved for a while (3 surveys), and many other criteria. As for the followers, participants reported to always accept new requests (5 surveys), match with the leaders who are more persistent (4 surveys), match with leaders that are blinking (2 surveys), and so on. Clearly, trying to take all reported strategies into account (that might not correspond to the real strategies employed) would result in

[4] Difference between the solution time of subsequent experiments not significant (least squares regression, average additional $1.44 sec.$ for each subsequent experiment, p-value 0.35), also controlling for NetworkID ($0.51 sec.$ less for each subsequent experiment on the same network, p-value 0.70).

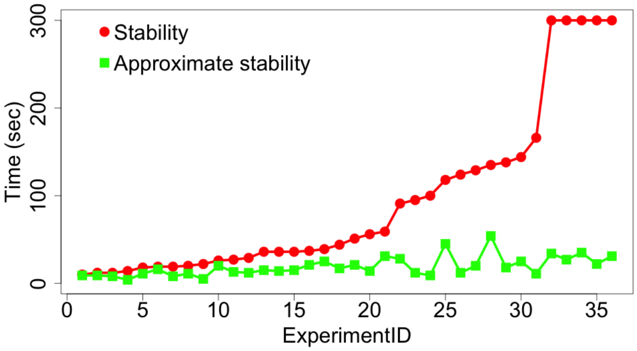

Fig. 8. **Human subjects performance: approximation versus stability.** The x-axis shows the experimentID (sorted by increasing solving time). The red line shows the time spend by the human subjects to solve each experiment (300 seconds if the experiment was not solved), the green line shows the corresponding time spent to reach an approximate matching in which only a single leader needs an additional follower.

a complex and mathematically intractable model, and prevent us from deriving the clean trade-off between time and quality of the solution as stated in Theorem 1. In practice, preferring unmatched followers appears to be a natural strategy, pursuing local stability is an inherent characteristic of human behavior – although subjects might not admit it explicitly – while randomization captures the diversity of the actions of the population as evidenced in the exit pools.

VII. Discussion

The algorithmic model we proposed presents a set of desirable features aimed at modeling distributed social computation, and is simple enough to be prone to rigorous mathematical analysis. Despite its simplicity, it is able to predict human performance and fits the experimental data, showing that the global dynamics of complex agents with possibly diverse strategies can be described by simple synthetic agents with uniform strategies. We advocate the usage of similarly simple algorithmic models to capture the essence of social interaction and to investigate a wider variety of social computation tasks.

In order to evaluate the proposed algorithmic model as a possible description of human behavior, we created an artificial environment in which human subjects solve the group membership task on virtual networks. In these experiments, participants have the possibility to send, accept or decline matching requests as long as a solution is not reached and they are given a monetary reward upon successful and timely completion of each task. Two important features of the experiments are absent from the algorithmic model – the monetary reward and the time threshold. The monetary reward provides an incentive to solve each experiment and, as such, it appears necessary in the experimental design. On the other hand, in the algorithmic model, agents act until a stable solution is reached. The five-minute time threshold on each experiment guarantees that the entire experimental session has a constrained duration even in the presence of hard-to-solve networks. For such networks, we can compare the time to reach approximate solutions and a lower bound for the time to reach a stable solution. Despite these differences and the fact that participants can follow arbitrary strategies, the proposed algorithmic model is able to qualitatively capture the dynamics of the human subjects.

For ease of presentation, we assumed that agents are synchronized. We can consider an asynchronous setting where each of the m leaders has a clock that activates at random times. When a leader's clock rings, the leader acts according to the algorithm in Table 1. Followers are activated by incoming requests from leaders and act according to the algorithm in Table 2. Our results would continue to hold substantially unchanged. For example, if we consider independent Poisson processes with inter-point times that are exponentially distributed with parameter $\lambda = 1$, a total of m events occur on average in a unit of time. This is comparable to the synchronous scenario in which all m leaders have the possibility to act during each time interval. The argument in the proof Theorem 1 would follow similarly by considering time intervals of fixed duration δ and replacing p by $p(1-e^{-\delta})$, where the term between parentheses is a lower bound for the probability that a given clock rings within δ. The upper bound for the time to reach a $(1-\varepsilon)$-approximate stable matching would present a multiplicative factor that depends on δ. The best choice of δ depends on ε and for constant ε the multiplicative factor is constant in m.

In practical scenarios, the constraints c_ℓ can be lower bounds rather than exact targets. In this case, we assume that leader ℓ sends matching requests as long as her team's size is smaller than c_ℓ or there are unmatched followers in her neighborhood, and our main result in Theorem 1 continues to hold. In addition, if $\sum_\ell c_\ell < n$, where n is the number of followers, some followers will be unmatched in any configuration, and particularly in the first stable matching that is reached (if a stable matching exists), unless the constraints c_ℓ are lower bounds.

In the present paper, a fixed network topology is assumed. Multi-agent systems are often characterized by time-changing topologies, which result from either agents mobility or unreliable communication and whose effect on the system dynamics depends on the particular scenario. For example, Sarwate and Dimakis [31] showed that evolving topologies might help the diffusion of information in the context of averaging, Xiao and Wang [32] showed that they can harm convergence to consensus unless certain connectivity conditions hold. If the network topology is allowed to vary arbitrarily, our main result (Theorem 1) gives a performance guarantee in terms of the time *since* the most recent change in the network topology. We observe that a change in network topology that affects only few edges might in general result in a significant change in the optimal solution, and we leave the rigorous analysis of time-varying topologies to future investigation.

Finally, in the present work, we defined the quality of a solution in terms of the additional number of followers needed by all leaders in order to satisfy all size constraints (and we called this quantity the *deficit* of a matching). We proved that our algorithm constitutes a Polynomial Time Approximation Scheme (PTAS) for the minimization of this quantity, that is, *any* constant approximation of the optimal solution is reached in polynomial time in *any* instance of the problem. Different quantities might be better suited to express the quality of an approximate solution in different applications. However, provable performance guarantees might in general be derived only for a subclass of the problem instances, and the analysis depends on the particular definition of approximation that is considered.

Appendix A
Proof of Theorem 1

Fix $0 < \varepsilon < 1$. Observe that $d(M(t))$ is non-increasing in t, as leaders do not voluntarily disengage from the followers in their groups (and therefore the deficit of a leader increases of one unit only if the deficit of another leader decreases by one unit). Moreover, since $c_\ell \geq 1$ for every leader ℓ, and G admits a stable matching, $d(M(t)) \leq m$ for every t.

For every $0 < x \leq 1$, let

$$\tau(x) = \min\left\{t \geq 0 : d(M(t)) < xm\right\}$$

be the first round at whose beginning the deficit is strictly smaller than xm. We want to find an upper bound for $\tau(\varepsilon)$.

Consider any round $t \geq 0$. Since $d(M(t)) \leq m$, there exists $0 < \varepsilon' \leq 1$ such that $d(M(t)) = \varepsilon' m$ (we assume $\varepsilon' > 0$, as the case of $\varepsilon' = 0$ is trivial). The following lemma bounds the number of rounds $\tau(\varepsilon') - t$ needed for a one-unit reduction of the deficit. Let $\Delta = \max_{\ell \in L} |N_\ell|$.

Lemma 2: Let $d(M(t)) = \varepsilon' m$ for some $0 < \varepsilon' \leq 1$. Then

$$\Pr\left(\tau(\varepsilon') - t \leq \lfloor 1/\varepsilon' \rfloor\right) \geq \left(\frac{pq}{\Delta}\right)^{\lfloor 1/\varepsilon' \rfloor}.$$

Proof: Let $h(t) \geq 1$ be the odd length of each shortest deficit-decreasing path relative to $M(t)$. By Corollary 1, $h(t) \leq 2\lfloor 1/\varepsilon' \rfloor - 1$. We distinguish the cases of $h(t) = 1$ and $h(t) \geq 3$. First consider $h(t) = 1$. With probability at least pq/Δ the deficit decreases by at least one unit during the next round of the algorithm. Too see this, consider a deficit-decreasing path ℓ, f. With probability at least p/Δ, ℓ tries to match with f and, conditional on this, f considers ℓ's proposal with probability q, resulting in the lower bound pq/Δ.

Now consider $h(t) \geq 3$, and let P be a shortest deficit-decreasing path of length $h(t)$ ending at an unmatched follower f. The length of P decreases by one in the next round with probability at least pq/Δ (f is unmatched after round t as P is a shortest deficit decreasing path and $h(t) > 1$).

By independence of successive rounds of the algorithm and the bound $h(t) \leq 2\lfloor 1/\varepsilon' \rfloor - 1$, with probability at least $(pq/\Delta)^{\lfloor 1/\varepsilon' \rfloor}$, a sequence of $\lfloor 1/\varepsilon' \rfloor - 1$ rounds reduces the length of P to 1 and then in one additional round P is "solved" and the deficit decreases by one unit. □

Consider consecutive phases of $\lfloor 1/\varepsilon \rfloor$ rounds each. For phases $i = 0, 1, 2, \ldots$, let X_i be iid Bernoulli random variables with $\Pr(X_i = 1) = (pq/\Delta)^{\lfloor 1/\varepsilon \rfloor}$. By Lemma 2, after T phases (i.e., at the beginning of round $t^* = T\lfloor 1/\varepsilon \rfloor$), the deficit of the matching is upper bounded by

$$d(M(t^*)) < \max\{\varepsilon m, m + 1 - \sum_{i=1}^{T} X_i\},$$

as the matching at the beginning of round 0 has deficit $d(M(0)) \leq m$. By independence of the phases, a Chernoff bound implies that for any $0 < \delta \leq 1$

$$\Pr\left(\sum_{i=1}^{T} X_i < (1-\delta) T (pq/\Delta)^{\lfloor 1/\varepsilon \rfloor}\right) < e^{-T(pq/\Delta)^{\lfloor 1/\varepsilon \rfloor}\delta^2/2}.$$

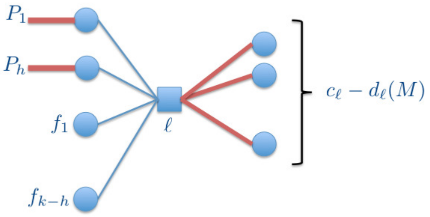

Fig. 9. The leader ℓ in the proof of Lemma B, matched edges are highlighted.

Setting $\delta = \varepsilon$ and $T = cm(\Delta/pq)^{\lfloor 1/\varepsilon \rfloor}$ (where c is a constant to be specified later), the deficit of the matching at the beginning of round $t^* = \lfloor 1/\varepsilon \rfloor cm(\Delta/pq)^{\lfloor 1/\varepsilon \rfloor}$ is

$$d(M(t^*)) < \max\{\varepsilon m, m + 1 - (1-\varepsilon)cm\}$$

with probability at least $1 - e^{-cm\varepsilon^2/2}$. To conclude the proof of the theorem we need that $\varepsilon m \geq m + 1 - (1-\varepsilon)cm$, which is true for any $c \geq 1 + \frac{1}{m(1-\varepsilon)}$.

Appendix B
Proof of Lemma 1

Given the matching M and the stable matching N, for brevity we write deficit-decreasing path instead of deficit-decreasing path in $M \oplus N$ relative to M.

The proof is divided in two parts. First, we show that for each leader ℓ with deficit $d_\ell(M) > 0$ there are at least $d_\ell(M)$ follower-disjoint deficit-decreasing paths starting at ℓ. Then, we argue that $d(M)$ follower-disjoint deficit-decreasing paths can be chosen, $d_\ell(M)$ of which start at each leader ℓ with deficit $d_\ell(M) > 0$, and the claim of the lemma follows.

Consider a leader ℓ with $d_\ell(M) > 0$. Assume by contradiction that there are strictly less then $d_\ell(M)$ follower-disjoint deficit-decreasing paths starting at ℓ (see Figure 9). Since ℓ has a team size constraint $c_\ell > 0$, there are exactly $c_\ell - d_\ell(M)$ followers that are matched to ℓ. Observe that no follower matched to ℓ can be the first follower of a deficit-decreasing path starting at ℓ (because the first edge must be in $N \backslash M$).

Since G admits a stable matching, the neighborhood N_ℓ of ℓ has size $|N_\ell| \geq c_\ell$. Therefore, there are $k \geq d_\ell(M)$ followers in N_ℓ that are not matched to ℓ. Assume that $h < d_\ell(M)$ of the followers in N_ℓ are the first followers of h follower-disjoint deficit-decreasing paths starting at ℓ ($P_1, \ldots, P_h$ in Figure 9). Denote the remaining $k - h > 0$ followers by $f_1, \ldots, f_{k-h}$, and assume by contradiction that none among them is the first follower of a deficit-decreasing path starting at ℓ (i.e., there are strictly less than $d_\ell(M)$ follower-disjoint deficit-decreasing paths starting at ℓ).

Observe that, in order to become stable, ℓ needs to match with at least one additional follower among $\{f_1, \ldots, f_{k-h}\}$. We show that, under the assumption above, a one-unit reduction in the deficit of ℓ would eventually result in a one-unit increase of the deficit of another leader, implying that G does not admit a stable matching, generating a contradiction.

Consider any follower $f' \in \{f_1, \ldots, f_{k-h}\}$, and observe that f' is matched in M since otherwise $\ell f'$ would be a deficit-decreasing path starting at ℓ. Let ℓ' be the leader such that $(\ell', f') \in M$, and observe that if ℓ' is matched to all followers in $N_{\ell'}$ then ℓ cannot match to f' without causing a one-unit increase of the deficit of ℓ'. Therefore assume that in $N_{\ell'}$ there is a follower f'' such that $(\ell'', f'') \in M$ for some leader $\ell'' \neq \ell'$ (f'' is matched in M since otherwise ℓ, f', ℓ', f'' is a deficit-decreasing path). In the following two cases ℓ cannot match to f' without eventually increasing the deficit of another leader: (i) $\ell'' = \ell$, and $\ell, f', \ell', f'', \ell$ is a cycle; (ii) $\ell'' \neq \ell$ and ℓ'' is matched to all followers in $N_{\ell''}$ other than f'.

Therefore assume that in $N_{\ell''}$ there is a follower f''' such that $(\ell''', f''') \in M$ for some leader $\ell''' \neq \ell''$ (f''' is matched in M). Again, ℓ cannot match to f' without eventually increasing the deficit of another leader if either $\ell''' = \ell$ or $\ell''' = \ell'$ (each similar to the case (i) above), or if ℓ''' is matched to all followers in $N_{\ell''}$ other than f', f'' (similar to the case (ii) above). By iteration, it follows that ℓ cannot match to any follower $f' \in \{f_1, \ldots, f_{k-h}\}$ without eventually increasing the deficit of another leader, contradicting with the existence of the stable matching N. Hence, there are at least $d_\ell(M)$ follower-disjoint deficit-decreasing paths starting at ℓ.

To complete the proof of the lemma, we show that we can choose $d(M)$ follower-disjoint deficit-decreasing paths, $d_\ell(M)$ of which start at each leader ℓ with $d_\ell(M) > 0$. We proceed by contradiction, and make the following assumption. For any set $\mathcal{P}$ of $d(M)$ deficit-decreasing paths, $d_\ell(M)$ of which start at each leader ℓ with $d_\ell(M) > 0$ (denote by $\mathcal{P}_\ell$ the elements of $\mathcal{P}$ starting at ℓ), there are leaders ℓ, ℓ' and paths $P \in \mathcal{P}_\ell$, $P' \in \mathcal{P}_{\ell'}$ that are not follower-disjoint. In order to reach the stable matching N starting from M, a set of $d(M)$ deficit-decreasing paths must be solved. However, if P is solved (by "flipping" matched edges into unmatched edges, and vice versa) then P' is not solved, and if P' is solved then P is not solved. It follows that N cannot be reached from M by solving the $d(M)$ deficit-decreasing paths in $\mathcal{P}$. By the assumption above, the last argument holds for any choice of $\mathcal{P}$, and this generates a contradiction on the reachability of N starting from M (observe that N can be reached from M in finite time, e.g. by a cat-and-mouse argument on the space of all the matchings of G). The lemma is proven.

Appendix C
Proof of Theorem 2

Let $\mathcal{M}_n$ be the set of all the matchings of G_n such that $d(M) = 1$. We proceed as follows. First, we show that each $M \in \mathcal{M}_n$ is uniquely identified by the set of the leaders that are not matched with "horizontal" edges (that is, leaders ℓ_i such that $(\ell_i, f_i) \notin M$). Second, we define trees T^*_m, $m \geq 1$ such that a random walk on T^*_m starting at any node different than the root hits the root after a number of steps that is exponentially large in m with high probability. Third, for each matching $M \in \mathcal{M}_n$ we define a quantity $h(M)$ that we call the *height* of M and we argue that, when initialized at M, the algorithm's dynamics is equivalent to a random walk on the tree $T^*_{h(M)}$ and reaching the stable matching of G_n corresponds to reaching the root of $T^*_{h(M)}$ (and therefore requires a number of rounds that is exponentially large in $h(M)$ with high probability). Finally, by a counting argument, we show that for any constant $0 < \gamma < 1$ a $1 - O(n2^{-(1-\gamma)n})$ fraction of all the matchings in $\mathcal{M}_n$ have height at least γn, completing the proof of the theorem.

A. Properties of the matchings in $\mathcal{M}_n$

Matchings in $\mathcal{M}_n$ enjoy the following structural properties.

Lemma 3: Let $M \in \mathcal{M}_n$. The following properties hold.

(1) There are a single poor leader $\ell_{i^*(M)}$ and a single unmatched follower $\ell_{j^*(M)}$ in M.
(2) $1 \leq i^*(M) \leq j^*(M) \leq n$.
(3) $(\ell_k, f_k) \in M$ for all $k < i^*(M)$ and all $k > j^*(M)$.
(4) Let $\mathcal{I}(M) = \{j_0, j_1, \ldots, j_K\}$ be the sorted set of indexes j such that $(\ell_j, f_j) \notin M$. Then
   (a) $j_1 = i^*(M)$ and $j_K = j^*(M)$.
   (b) $(\ell_{j_{k+1}}, f_{j_k}) \in M$ for all $k \in \{0, \ldots, K-1\}$.

Proof: Property (1). Since $d(M) = \sum_{\ell \in L} d_\ell(M) = 1$, there is a single poor leader $\ell_{i^*(M)}$ in M. Since $c_\ell = 1$ for all $\ell \in L$, each leader $\ell \neq \ell_{i^*(M)}$ is matched to a single follower. It follows that there is a unique unmatched follower $f_{j^*(M)}$.

Property (2). Suppose by contradiction that $i^*(M) > j^*(M)$. Since $N_{\ell_{j^*(M)}} = \{f_1, \ldots, f_{j^*(M)}\}$ and $f_{j^*(M)}$ is unmatched, leader $\ell_{j^*(M)}$ is matched to one of the followers in $\{f_1, \ldots, f_{j^*(M)-1}\}$. Hence, the $j^*(M) - 1$ leaders $\ell_1, \ldots, \ell_{j^*(M)-1}$ are matched to at most $j^*(M) - 2$ out of the $j^*(M) - 1$ followers $f_1, \ldots, f_{j^*(M)-1}$, and one of them is necessarily poor, contradicting Property (1).

Property (3). We proceed by induction. If $i^*(M) > 1$, then $(\ell_1, f_1) \in M$ since $N_{\ell_1} = \{f_1\}$ and ℓ_1 is matched with a follower. Assume that if $i^*(M) > j$ then $(\ell_k, f_k) \in M$ for all $k \leq j$. If $i^*(M) > j + 1$, then, by the inductive assumption, ℓ_{j+1} can only be matched to f_{j+1} since $N_{\ell_{j+1}} = \{f_1, \ldots, f_{j+1}\}$. This shows that $(\ell_k, f_k) \in M$ for all $k < i^*(M)$. $k > j^*(M)$ is shown similarly.

Property (4). If $K = 0$ then $M = \{(\ell_i, f_i) : i \neq i^*(M)\}$, $j^*(M) = i^*(M)$, and properties (4a) and (4b) trivially hold. Now consider $K \geq 1$. Let $\mathcal{I}(M) = \{j_0, j_1, \ldots, j_K\}$ be the sorted set of indexes j such that $(\ell_j, f_j) \notin M$. By property (3), we have that $j_0 = i^*(M)$ and $j_K = j^*(M)$, therefore property (4a) follows. Hence, $(\ell_{j_2}, f_{j_1}) \in M$ since $(\ell_k, f_k) \in M$ for all $k \in \{j_1 + 1, \ldots, j_2 - 1\}$ by definition of $\mathcal{I}(M)$, and $N_{\ell_{j_2}} = \{f_1, \ldots, f_{j_2}\}$. Property (4b) follows by induction. □

Lemma 3 states that non-horizontal matching edges do not intersect. In particular, given a matching $M \in \mathcal{M}_n$, the set $\mathcal{I}(M)$ represents the set of (the sorted indexes of) the leaders that are not matched with horizontal edges (see Figure 10 for an example), $\ell_{i^*(M)}$ for $i^*(M) = \min \mathcal{I}(M)$ is the unique unmatched leader, and $f_{j^*(M)}$ for $j^*(M) = \max \mathcal{I}(M)$ is the unique unmatched follower. Recall that $M^*_n = \{(\ell_k, f_k) : 1 \leq k \leq n\}$ is the unique stable matching of G_n, and let $\mathcal{I}(M^*_n) = \emptyset$. Lemma 3 implies that every matching $M \in \mathcal{M}_n \cup \{M^*_n\}$ is uniquely identified by the set $\mathcal{I}(M)$.

Lemma 4: The mapping $\mathcal{I}(\cdot)$ from $\mathcal{M}_n \cup \{M^*_n\}$ to $\mathcal{S} = \{A : A \subseteq \{1, \ldots, n\}\}$ defined by $M \mapsto \mathcal{I}(M)$ is a bijection.

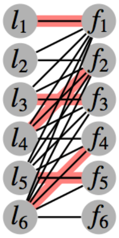

Fig. 10. An example of a matching M of G_6 with $d(M) = 1$. M is uniquely determined by the set $\mathcal{I}(M) = \{2, 4, 6\}$, encoding the following: ℓ_2 is not matched, ℓ_4 is matched with f_2, ℓ_6 is matched with f_4, f_6 is not matched. $P(M) = \ell_2, f_2, \ell_4, f_4, \ell_6, f_6$ is the unique deficit-decreasing path.

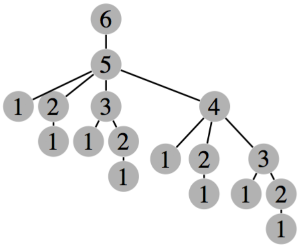

Fig. 11. The three T^*_m for $m = 5$.

Proof: $\mathcal{I}(\cdot)$ is injective as if $M, M' \in \mathcal{M}_n$ and $M \neq M'$ then $\mathcal{I}(M) \neq \mathcal{I}(M')$. $\mathcal{I}(\cdot)$ is surjective, as for $K \leq n - 1$ and $A = \{i_0, \ldots, i_K\} \in \mathcal{S}$ such that $1 \leq i_0 < \ldots < i_K \leq n$, the matching $M \in \mathcal{M}_n$ such that $\mathcal{I}(M) = A$ is $M = \{(\ell_{i_{k+1}}, f_{i_k}) : 0 \leq k \leq K - 1\} \cup \{(\ell_k, f_k) : k \notin A\}$.□

B. The tree T^*_m

Let T_1 be a labeled rooted tree with a singleton node with label 1. Inductively, for $i \leq 2$, let T_i be the labeled rooted tree whose root is labeled with i and its $i-1$ children are the roots of copies of $T_1, \ldots, T_{i-1}$. Define T^*_m to be the tree with a root with label $m+1$ whose only child is the root of a copy of T_m (see Figure 11). Let r^* be the root of T^*_m. We show that the hitting time of r^* for a random walk on T^*_m starting at any node $u \neq r^*$ is exponential in m with high probability. For a node $u \neq r^*$, we call the edge that connects u to its parent u's *exit* edge. For any subtree $T_i \subset T^*_m$, let Z_i be the random variable denoting the number of steps that it takes for a walk starting at the root of T_i to exit T_i (that is, to hit the parent of the root of T_i). The following bound holds for Z_i.

Lemma 5: There exist $\alpha, \gamma > 0$ such that, for all $i \geq 2$, $\Pr[Z_i \geq \gamma \cdot 2^{i/(\alpha \log^2 i)}] \geq 1 - \frac{1}{\log i}$.

Proof: We proceed by induction on i. Let $g(i) = \alpha \log^2 i$ and $f(i) = \gamma \cdot 2^{i/g(i)}$ for some $\alpha, \gamma > 0$. For any $\alpha > 0$ and $i \geq 2$, we can choose $\gamma > 0$ such that $f(i) \leq 1$; therefore, as $Z_i \geq 1$ with probability 1, the claim holds trivially for any $i \leq i^*$, where i^* is a suitably large constant. Now consider any $i \geq i^*$ and suppose the claim holds up to $i-1$. Every time the walk is on the root of T_i, it exits T_i with probability $1/i$ (since the root of T_i one parent and $i-1$ children). Therefore, letting E_t be the event that the first t times the walk is on the root of T_i it does *not* exit T_i, we have $\Pr[E_t] \geq 1 - t/i$. Let $t = i/(2 \log i)$, and let D_j, $1 \leq j \leq t$, be the event that, when it is on the root of T_i for the j-th time, the walk moves to the root of one of the subtrees $T_{i-g(i)}, \ldots, T_{i-1}$ *and* takes at least $f(i - g(i))$ steps to exit that subtree. For $1 \leq j \leq t$,

$$\begin{aligned}\Pr[D_j \mid E_t] &\geq g(i)/i \cdot \Pr[Z_{i-g(i)} \geq f(i - g(i))] \\ &\geq g(i)/i \cdot \left(1 - [\log(i - g(i))]^{-1}\right),\end{aligned}$$

by the induction hypothesis on $Z_{i-g(i)}$. Let χ_j be the indicator function of D_j for $1 \leq j \leq t$. The probability that at least two of the events D_j happen, given E_t, is lower bounded by:

$$\begin{aligned}&\Pr\left[\sum_{j=1}^{t} \chi_j \geq 2 \,\middle|\, E_t\right] \geq \Pr\left[\sum_{j=1}^{t/2} \chi_j \geq 1, \sum_{j=t/2+1}^{t} \chi_j \geq 1 \,\middle|\, E_t\right] \\ &= \Pr\left[\sum_{j=1}^{t/2} \chi_j \geq 1 \,\middle|\, E_t\right]^2 \geq \left(1 - \prod_{i=1}^{t/2}(1 - \Pr[D_j|E_t])\right)^2 \\ &\geq \left(1 - \left(1 - \frac{g(i)}{i}\left(1 - \frac{1}{\log(i - g(i))}\right)\right)^{t/2}\right)^2 \\ &\geq \left[1 - \exp\left[\frac{-\alpha \log i}{4}\left(1 - \frac{1}{\log(i - g(i))}\right)\right]\right]^2 \geq \left(1 - \frac{1}{i^{\alpha/8}}\right)^2\end{aligned}$$

where we applied union bound in the third line, and the last step holds for i sufficiently large so that $\log(i - g(i)) \geq 2$. Noting that

$$(1 - \frac{1}{i^{\alpha/8}})^2 \geq 1 - \frac{2}{i^{\alpha/8}},$$

we conclude that

$$\begin{aligned}&\Pr[Z_i \geq 2 \cdot f(i - g(i))] \geq \Pr\left[\sum_{j=1}^{t} \chi_j \geq 2\right] \\ &\geq \Pr\left[\sum_{j=1}^{t} \chi_j \geq 2 \,\middle|\, E_t\right] \Pr[E_t] \geq \left(1 - \frac{2}{i^{\alpha/8}}\right)\left(1 - \frac{t}{i}\right),\end{aligned}$$

which is greater than $1 - 1/\log i$ for α sufficiently large. The claim follows since $2 \cdot f(i - g(i)) \geq f(i)$. □

Corollary 2: The hitting time of r^* of a random walk starting at any node $u \neq r^*$ is $2^{\Omega(n/\log^2 n)}$ with high probability.

C. The dynamics of the algorithm starting from $M \in \mathcal{M}_n$

For ease of presentation, we set the parameters of the algorithms to $p = q = 1$ (our result holds in general).

Definition 8 (The height of a matching): Let $M \in \mathcal{M}_n$, $\mathcal{I}(M) = \{i_0, \ldots, i_K\}$. The height of M is $h(M) = 0$ if $K = 0$, and $h(M) = i_{K-1} \in \{1 \ldots, n-1\}$ if $K \geq 1$.

For $M \in \mathcal{M}_n$, $h(M) > 0$, $\mathcal{I}(M) = \{i_0, \ldots, h(M), i_K\}$. For $t \geq 0$, let $M(t)$ be the matching at the beginning of round t. For ease of notation let $\mathcal{I}(t) = \mathcal{I}(M(t))$. For $M \in \mathcal{M}_n$ let

$$\tau^*(M) = \min\left\{t : M(t) = M^*_n | M(0) = M\right\}$$

be the number of steps that the algorithm needs to reach the stable matching starting from M.

Note that, with $p = q = 1$, $t^*(M) = 1$ for every $M \in \mathcal{M}_n$ such that $h(M) = 0$ (that is, $|\mathcal{I}(M)| = 1$), since according to the algorithm leaders prefer unmatched followers. We are interested in relating $\tau^*(m)$ and $h(M)$ for every matching $M \in \mathcal{M}_n$ such that $h(M) > 0$ (that is, $|\mathcal{I}(M)| > 1$).

We study how the matching evolves over time through the Markov process $\{\mathcal{I}(t) : 0 \leq t \leq \tau^*(M)\}$. Since $\mathcal{I}(M^*_n) = \emptyset$, $\tau^*(M) = \min\{t : \mathcal{I}(t) = \emptyset\}$. The state space of the Markov process is given by the set $\mathcal{S}$ defined in Lemma 4. The transition probabilities are as follows.

Lemma 6: Conditional on $\mathcal{I}(t) = I \in \mathcal{S}$, $|I| > 1$, the transition probabilities at time t are given by

$$\Pr\Big(\mathcal{I}(t+1) = I' \big| \mathcal{I}(t) = I\Big) = 1/\min I.$$

if $I' \in \{I \cup \{k\} : k < \min I\} \cup \{I \backslash \{\min I\}\}$ and 0 otherwise. Moreover $\Pr(\mathcal{I}(t+1) = \emptyset | \mathcal{I}(t) = \emptyset) = 1$, and $\Pr(\mathcal{I}(t+1) = \emptyset | \mathcal{I}(t) = I) = 1$ for every I sich that $|I| = 1$.

The proof is omitted for brevity. For every $M \in \mathcal{M}_n$ such that $h(M) > 0$ and $\mathcal{I}(M) = \{i_0, \ldots, i_K\}$, define the matching $\mathcal{L}(M) = \{(\ell_j, f_j) : j \neq i_K\}$, and $\tau(M) = \min\{t : M(t) = \mathcal{L}(M)\}$. Note that $h(\mathcal{L}(M)) = 0$ and $\tau^*(M) > \tau(M)$.

For every matching M such that $|\mathcal{I}(M)| > 1$, let $\mathcal{R}(M)$ be the set of the matchings in $\mathcal{M}_n$ that can be reached from M (after one or multiple steps). By the transition probabilities of Lemma 6, $\mathcal{R}(M) = \{M' \in \mathcal{M}_n : I(M') = A \cup \{h(M), i_K\}, A \in \mathcal{A}\} \cup \{\mathcal{L}(M)\}$,where $\mathcal{A} = \{A \subseteq \{1, \ldots, h(M) - 1\}\}$. Observe that every $M' \in \mathcal{R}(M) \backslash \{\mathcal{L}(M)\}$ has height $h(M') = h(M)$. The following lemma characterizes the one-to-one correspondence between matchings in $\mathcal{R}(M)$ and nodes of the tree $T^*_{h(M)}$.

Lemma 7: Consider the mapping $\omega(\cdot)$ from $\mathcal{R}(M)$ to $T^*_{h(M)}$ defined as follows. Let $\omega(\mathcal{L}(M)) = r$, where r is the root of $T^*_{h(M)}$. For $M' \in \mathcal{R}(M) \backslash \{\mathcal{L}(M)\}$ and $\mathcal{I}(M') = I$, let $\omega(M')$ be the node of $T^*_{h(M)}$ with label $\min I$ and connected to the root with the path of nodes labeled by the sorted indexes in $I \backslash \{\min I\}$. Then $\omega(\cdot)$ is a bijection.

The proof follows from the construction of $T^*_{h(M)}$ and $\mathcal{I}(\cdot)$.

Lemma 8: The process $\{\mathcal{I}(t) : 0 \leq t \leq \tau(M) | M(0) = M\}$ is equivalent to a random walk on $T^*_{h(M)}$ starting at $\omega(M)$.

The proof is omitted for brevity (the transition probabilities between two matchings $M_1, M_2 \in \mathcal{R}(M)$ are nonzero if and only if the nodes $\omega(M_1)$ and $\omega(M_2)$ are adjacent in $T^*_{h(M)}$). To summarize, the number of steps to reach the stable matching of G_n starting from $M \in \mathcal{M}_n$ with $h(M) > 0$ is upper bounded by the time $\tau(M)$ to reach the matching $\mathcal{L}(M)$, and reaching $\mathcal{L}(M)$ is equivalent to reaching the root of $T^*_{h(M)}$ starting from the node $\omega(M)$. By Corollary 2, $\tau(M)$ is exponentially large in $h(M)$ with high probability.

D. The fraction of matchings $M \in \mathcal{M}_n$ such that $h(M) \geq \gamma n$

Let N be the number of matchings in $\mathcal{M}_n$. Fixed a constant $0 < \gamma < 1$, let $\mathcal{M}_\gamma = \{M \in \mathcal{M}_n : h(M) < \gamma n\}$ and let $N_\gamma = |\mathcal{M}_\gamma|$. For $j = 0, \ldots, n-1$, let $N(j)$ be the number of matchings $M \in \mathcal{M}_n$ such that $h(M) = j$. It follows that $N = \sum_{j=0}^{n-1} N(j)$, $N_\gamma \leq \sum_{j=0}^{\lceil \gamma n \rceil - 1} N(j)$.

Lemma 9: $N(0) = n$ and $N(j) = (n-j)2^{j-1}$ for all $j = 1, \ldots, n-1$.

Proof: $N(0) = n$ since there are n matchings M with $h(M) = 0$ ($\{(\ell_j, f_j) : j \neq k\}$ for $1 \leq k \leq n$). Fix $j \in \{1, \ldots, n-1\}$. By Lemma 4, any $M \in \mathcal{M}_n$ with $h(M) = j$ is uniquely identified by $\mathcal{I}(M) = \{i_0, \ldots, i_{K-1}, i_K\}$ for some $1 \leq K \leq n-1$ and $i_{K-1} = j$. Since $\mathcal{I}(\cdot)$ is a bijection, to determine $N(j)$ we need to count all subsets of $\{1 \ldots, n\}$ of the form $\{i_0, \ldots, j, i_K\}$, thus $N(j) = (n-j)2^{j-1}$. $\square$

For any constant $0 < \gamma < 1$, the fraction of $M \in \mathcal{M}_n$ such that $h(M) < \gamma n$ goes to zero exponentially fast in n

Lemma 10: Fix $0 < \gamma < 1$. Then, $N_\gamma / N = O(n 2^{-(1-\gamma)n})$.

Proof: We first compute N.

$$N = \sum_{i=0}^{n-1} N(i) = n + \sum_{i=1}^{n-1} (n-i) 2^{i-1} = n + n \sum_{i=0}^{n-2} 2^i - \sum_{i=1}^{n-1} i 2^{i-1}.$$

The second sum can be shown (e.g. by induction) to be equal to $(n-1) + (n-2)(2^{n-1} - 1)$. Therefore,

$$N = n + n(2^{n-1} - 1) - (n-1) - (n-2)(2^{n-1} - 1) = \Omega(2^n).$$

Similarly, letting $k = \lceil \gamma n \rceil$ we have that,

$$\begin{aligned} N_\gamma &\leq \sum_{i=0}^{k-1} N(i) = n + n \sum_{i=0}^{k-2} 2^i - \sum_{i=1}^{k-1} i 2^{i-1} \\ &= n + n(2^{k-1} - 1) - (k-1) - (k-2)(2^{k-1} - 1) \\ &= 2^{k-1}(n - k - 2) - 1 = O(n 2^{\lceil \gamma n \rceil}). \end{aligned}$$

Therefore, the fraction of matchings in $\mathcal{M}_n$ with height $h(M) < \gamma n$ is $N_\gamma / N = O(n 2^{-(1-\gamma)n})$. $\square$

References

[1] M. Cook, "Universality in elementary cellular automata," *Complex Systems*, vol. 15, no. 1, pp. 1–40, 2004.

[2] T. M. Liggett, *Interacting particle systems*. Springer, 2005.

[3] A. Bovier, *Statistical mechanics of disordered systems: a mathematical perspective*. Cambridge University Press, 2006, vol. 18.

[4] J. Von Neumann, A. W. Burks *et al.*, "Theory of self-reproducing automata," *IEEE Transactions on Neural Networks*, vol. 5, no. 1, pp. 3–14, 1966.

[5] H. A. Simon, *Administrative behavior*. Cambridge University Press, 1976.

[6] M. Kearns, S. Suri, and N. Montfort, "An experimental study of the coloring problem on human subject networks," *Science*, vol. 313, no. 5788, pp. 824–827, 2006.

[7] J. S. Judd and M. Kearns, "Behavioral experiments in networked trade," in *Proceedings of the 9th ACM conference on Electronic commerce*. ACM, 2008, pp. 150–159.

[8] M. Kearns, S. Judd, J. Tan, and J. Wortman, "Behavioral experiments on biased voting in networks," *Proceedings of the National Academy of Sciences*, vol. 106, no. 5, pp. 1347–1352, 2009.

[9] S. Judd, M. Kearns, and Y. Vorobeychik, "Behavioral dynamics and influence in networked coloring and consensus," *Proceedings of the National Academy of Sciences*, vol. 107, no. 34, p. 14978, 2010.

[10] M. Kearns, S. Judd, and Y. Vorobeychik, "Behavioral experiments on a network formation game," in *Proceedings of the 13th ACM Conference on Electronic Commerce*. ACM, 2012, pp. 690–704.

[11] M. Kearns, "Experiments in social computation," *Communications of the ACM*, vol. 55, no. 10, pp. 56–67, 2012.

[12] A. E. Roth and J. H. V. Vate, "Random paths to stability in two-sided matching," *Econometrica: Journal of the Econometric Society*, pp. 1475–1480, 1990.

[13] D. P. Bertsekas, "A distributed asynchronous relaxation algorithm for the assignment problem," in *Decision and Control, 1985 24th IEEE Conference on*, vol. 24. IEEE, 1985, pp. 1703–1704.

[14] ——, "The auction algorithm: A distributed relaxation method for the assignment problem," *Annals of operations research*, vol. 14, no. 1, pp. 105–123, 1988.

[15] D. P. Bertsekas and D. A. Castanon, "A forward/reverse auction algorithm for asymmetric assignment problems," *Computational Optimization and Applications*, vol. 1, no. 3, pp. 277–297, 1992.

[16] V. Bala and S. Goyal, "A noncooperative model of network formation," *Econometrica*, vol. 68, no. 5, pp. 1181–1229, 2000.

[17] D. J. Abraham, A. Levavi, D. F. Manlove, and G. OMalley, "The stable roommates problem with globally-ranked pairs," in *Internet and Network Economics*. Springer, 2007, pp. 431–444.

[18] H. Ackermann, P. W. Goldberg, V. S. Mirrokni, H. Röglin, and B. Vöcking, "Uncoordinated two-sided matching markets," *SIAM Journal on Computing*, vol. 40, no. 1, pp. 92–106, 2011.

[19] E. M. Arkin, S. W. Bae, A. Efrat, K. Okamoto, J. S. Mitchell, and V. Polishchuk, "Geometric stable roommates," *Information Processing Letters*, vol. 109, no. 4, pp. 219–224, 2009.

[20] A. Israeli, M. D. McCubbins, R. Paturi, and A. Vattani, “Low memory distributed protocols for 2-coloring,” in *Stabilization, Safety, and Security of Distributed Systems*. Springer, 2010, pp. 303–318.

[21] A. Nedic and A. Ozdaglar, “Distributed subgradient methods for multi-agent optimization,” *Automatic Control, IEEE Transactions on*, vol. 54, no. 1, pp. 48–61, 2009.

[22] Y. Kanoria, M. Bayati, C. Borgs, J. Chayes, and A. Montanari, “Fast convergence of natural bargaining dynamics in exchange networks,” in *Proceedings of the Twenty-Second Annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM, 2011, pp. 1518–1537.

[23] L. Coviello, M. Franceschetti, M. D. McCubbins, R. Paturi, and A. Vattani, “Human matching behavior in social networks: an algorithmic perspective,” *PloS one*, vol. 7, no. 8, p. e41900, 2012.

[24] A. Olshevsky and J. N. Tsitsiklis, “Convergence speed in distributed consensus and averaging,” *SIAM Journal on Control and Optimization*, vol. 48, no. 1, pp. 33–55, 2009.

[25] H. H. Nax, B. S. Pradelski, and H. P. Young, “Decentralized dynamics to optimal and stable states in the assignment game,” in *Decision and Control (CDC), 2013 IEEE 52nd Annual Conference on*. IEEE, 2013, pp. 2391–2397.

[26] J. Surowiecki, *The wisdom of crowds*. Random House Digital, 2005.

[27] D. P. Enemark, M. D. McCubbins, R. Paturi, and N. Weller, “Does more connectivity help groups to solve social problems,” in *Proceedings of the 12th ACM conference on Electronic commerce*, 2011, pp. 21–26.

[28] M. D. McCubbins, R. Paturi, and N. Weller, “Connected coordination network structure and group coordination,” *American Politics Research*, vol. 37, no. 5, pp. 899–920, 2009.

[29] T. Chakraborty, S. Judd, M. Kearns, and J. Tan, “A behavioral study of bargaining in social networks,” in *Proceedings of the 11th ACM conference on Electronic commerce*. ACM, 2010, pp. 243–252.

[30] J. E. Hopcroft and R. M. Karp, “An nˆ5/2 algorithm for maximum matchings in bipartite graphs,” *SIAM Journal on computing*, vol. 2, no. 4, pp. 225–231, 1973.

[31] A. D. Sarwate and A. G. Dimakis, “The impact of mobility on gossip algorithms,” *Information Theory, IEEE Transactions on*, vol. 58, no. 3, pp. 1731–1742, 2012.

[32] F. Xiao and L. Wang, “Asynchronous consensus in continuous-time multi-agent systems with switching topology and time-varying delays,” *Automatic Control, IEEE Transactions on*, vol. 53, no. 8, pp. 1804–1816, 2008.

Lorenzo Coviello (S’11) received a Laurea degree (magna cum laude) in Telecommunication Engineering from the University of Padova, Italy, in 2008, and a MS and Ph.D. in Electrical and Computer Engineering from the University of California San Diego in 2013 and 2015, respectively, under the advice of Prof. Massimo Franceschetti. His research is focused on social network analysis, computational social science, and algorithms.

Massimo Franceschetti (M’98–SM’11) received the Laurea degree (magna cum laude) in computer engineering from the University of Naples, Naples, Italy, in 1997, and the M.S. and Ph.D. degrees in electrical engineering from the California Institute of Technology, Pasadena, CA, in 1999, and 2003, respectively. He is professor of Electrical and Computer Engineering, University of California San Diego. Before joining UC San Diego, was a postdoctoral scholar at the UC Berkeley. He has held visiting positions at the Vrije Universiteit Amsterdam, the Ecole Polytechnique Federale de Lausanne, and the University of Trento. His research interests are in communication systems theory and include random networks, wave propagation in random media, and control over networks. Dr. Franceschetti was an Associate Editor for Communication Networks of the IEEE Transaction on Information Theory (2009 – 2012) and has served as Guest Editor for two issues of the IEEE Journal on Selected Areas in Communication. He is currently serving as Associate Editor of the IEEE Transactions on Networked Control Systems and the IEEE Transactions on Network Science and Engineering. He was awarded the C. H. Wilts Prize in 2003 for best doctoral thesis in electrical engineering at Caltech; the S.A. Schelkunoff Award in 2005 for best paper in the IEEE Transaction on Antennas and Propagation; a National Science Foundation (NSF) CAREER award in 2006, an ONR Young Investigator Award in 2007; the IEEE Communications Society Best Tutorial Paper Award in 2010; and the IEEE Control theory society Ruberti young researcher award in 2012.