The Effect of Mediated Partnerships in Two-Sided Economic Search

Philip Hendrix and David Sarne

School of Engineering and Applied Sciences
Harvard University
Cambridge, MA 02138 USA
{phendrix, sarned}@eecs.hrvard.edu

Abstract. In this paper we investigate the effect of mediated partnerships over agents' equilibrium strategies in two-sided economic search. A *mediated partnership* is formed when an agent acts as a mediator, establishing a partnership between a pair of agents it encountered along its search, thereby reducing the other agents' amount of search. Surprisingly, this reduction in market friction induced by mediated partnerships does not always improve market efficiency. Use of mediated partnerships changes the equilibrium strategies used by agents in two-sided search models and introduces substantial computational complexity. This computational complexity is overcome with an innovative algorithm that facilitates equilibrium calculation.

1 Introduction

Two-sided economic search concerns distributed problems in which self-interested agents search for appropriate partners to form mutually acceptable pairwise partnerships [10, 4, 1, 6, 3]. The concept of economic search differs from search in AI. AI search typically involves an agent finding a sequence of actions that take it from an initial state to a goal state, while economic search refers to the identification of the best agent for a partnership. Since the agents are self-interested, the analysis of two-sided economic search models is inherently equilibrium-based.

The goal of each agent in two-sided economic search is to find a partner that is optimally beneficial. Each agent is associated with a specific type that captures the benefit of partnering with it[1]. During each stage of search, agents randomly interact pairwise and learn each others' type. A partnership between two interacting agents will be formed if they both commit to it. The process of initiating and maintaining an interaction with another agent is associated with a cost (i.e., search cost) incurred by both agents. Therefore, a key challenge for each agent in such an environment is to identify the set of agent types with whom it is willing to partner.

This paper extends the traditional two-sided economic search model to mediated partnerships. A *mediated partnership* is formed when an agent acts as a mediator, using its memory of past interactions with two other agents to form a partnership between them. Specifically, we base our model on the traditional two-sided search model where utilities are non-transferable, an agent's utility is fully correlated with

[1] This concept of "type" is different than the concept of "type" used in many AI domains where an agent's type is derived from utility gained through different opportunities.

its partner's type, agents incur a search cost for each round of search, and agents' true types are revealed once paired [10, 6].

Multi-Agent System (MAS) environments in which agents must interact with each other to evaluate potential partners and form viable partnerships are of interest because of their potential economic and strategic importance. An example of such a system is the dual backup application [13] in which agents representing different servers seek to form partnerships for purposes of mutual offsite backup. Unlike tradition two-sided search models, when considering the process in a MAS, one must take into consideration the unique capabilities of autonomous agents for enhancing the search process. One important capability of this kind is an agent's ability to maintain a full recollection of past interactions with other agents and utilize this ability to act as a mediator for forming partnerships as described in this paper.

The contributions of the paper are threefold. First, it introduces and models a new distributed two-sided search enhanced with mediated partnerships. Second, the introduction of mediated partnership capabilities to two-sided economic search potentially creates an equilibrium which differs from that obtained in the traditional two-sided search model, and consequently can change individual and collective agent performance. The paper shows that the new model prevents a direct calculation of this new equilibrium. A new algorithm is introduced to find the new equilibrium. Finally, it is shown that although the memory and match-making extensions generally reduce market frictions and increase the overall efficiency of forming partnerships, in some rare cases overall system performance is decreased. This latter result is specifically important for market makers and multi-agent designers.

In Section 2 we formally introduce the two-sided search model with memory and mediated partnering. Section 3 gives an equilibrium analysis and discusses its cluster based partnering structure. We then describe a set of equations in Section 4 for the two-sided search model with mediated partnering for scenarios where agents form exactly into two-clusters. This analysis enables us to introduce examples in Section 5 showing the effect that memory can have on the traditional two-sided search model. Section 6 outlines an algorithmic based approach for determining agents' equilibrium strategies followed by experimental results that demonstrate its phenomenal efficiency. We conclude with reviewing related work (Section 7) and a discussion (Section 8).

2 The Model

We base our model on the traditional two-sided search model [10, 6]. In its most basic form the two-sided search model considers an environment populated with an infinite number of self-interested fully rational agents[2]. Each agent A_i has a type, defined over the continuum $[\underline{t}, \overline{t}]$, that captures special properties that characterize it. This type determines the utility that any other agent A_j gains if partnered with agent A_i. For simplicity, we assume that the expected utility gained by partnering

[2] The infinite number of agents assumption is common in two-sided search models (see [5, 17]). In many domains (e.g., eCommerce) this is derived from high entrance and exit rates, thus the probability of running into the same agent in a random pairing is negligible.

with an agent of type t ($t \in [\underline{t}, \overline{t}]$) is equal to t (i.e., $U(t) = t$). An agent can be of only one type, although many agents may be of the same type.

At the beginning of search, agents have no prior information concerning the type of specific agents in its environment. Agents randomly initiate pairwise interactions (i.e., search) with other agents to learn their type. The model assumes non-transferable utilities, thus no bargaining takes place as part of the interaction. Upon interaction, both agents reveal their type and the set of agent types with which it is willing to form a partnership. This set of acceptable types is called the *acceptance set*. A partnership is formed only if both agents are willing to commit to it (i.e., if each agent has the other agent's type in its acceptance set). Otherwise both agents resume their search in the same manner.

The process of initiating and maintaining an interaction is associated with a cost (i.e., search cost) incurred by both agents. This cost is equivalent to a reduction c in each agent's overall utility. Therefore, an agent chooses its acceptance set based on the expected utility gained from partnering and the cost of continued search[3].

While agents have no prior information concerning the types of a specific agent at the beginning of search, they are assumed to be acquainted with the general distribution of types in their environment, described by the probability distribution function $f(x)$ and cumulative distribution function $F(x)$. The nature of the two-sided search suggests that the agents are satisfied with a single partner, thus once a partnership is formed the two agents forming it terminate their search process and leave the environment. The two-sided search model assumes that the pair is replaced with two identical agents (i.e, having the same types) that start the search from scratch[4].

Our model extends the interaction protocol of two-sided search by utilizing information gleaned from previous interactions with potential partners. If a direct partnership is not formed, then instead of simply resuming search, as in the traditional two-sided search model, each agent records in memory the type and acceptance set of the other. Moreover, each agent seeks to act as a mediator (match maker) for the other by searching its memory, seeking a suitable partner for the other from its records of other agents already encountered. If such a mediator agent finds a partner for its newly rejected potential partner among the agents it has previously encountered, then a partnership is arranged for the newly rejected agent and the remembered agent, if neither has already entered a partnership. We call this process *mediated partnering*.

The model assumes that the information the agents exchange in their interactions is true, that it is not subject to change, and that the penalty for rejecting an agent of a type in the acceptance set is significantly greater than any utility that can be achieved. Thus, mediated partnering always takes place unless one of the agents has already partnered. We further assume that a mediated partnering is costless because it does not require any additional exploration or data exchange, except for checking that both agents are still in the market.

---

[3] In the absence of search costs, the unique competitive equilibrium has assortative joining, i.e., joined partners are identical (in type) [2].

[4] This assumption is commonly used in two-sided search literature in order to maintain the fixed distribution function $f(x)$ [3, 1].

3 Analysis

The analysis of the classical two-sided search model suggests a "complete segregation" solution in equilibrium [4, 6, 10]. Agents form clusters, based on their type, in which every agent in a cluster is always willing to partner with any other agent in the cluster. Consequently, the agents' acceptance (partnering) strategy is reservation-value based. The reservation value of an agent corresponds to the lowest type that it is willing to accept. This reservation value is different from the reservation price commonly found in e-commerce models associated with a buyer or a seller that is not involved in a search and denotes the buyer's or seller's true evaluation for the opportunity. The agent accepts all agents with types greater than or equal to its reservation value. Furthermore, the agents' search strategy is stationary (i.e. an agent will not change its reservation value).

In the traditional two-sided search model agents' reservation values in equilibrium can be found using backward induction. Consider the agent $A_{\bar{t}}$ of the highest type, $\bar{t}$, that needs to set its optimal reservation value. Obviously this agent's optimal reservation value will not be affected by the reservation value used by other agents, since all other agents will always accept type $\bar{t}$ (having no better type to strive for). Therefore, the expected utility of agent $A_{\bar{t}}$, denoted $V_{A_{\bar{t}}}(x)$, as a function of its reservation value x can be expressed as

$$V_{A_{\bar{t}}}(x) = \underset{\substack{\text{Cost of meeting}\\ \text{another agent}}}{-c} + \underbrace{\int_{y=x}^{\bar{t}} y f(y) dy}_{\substack{\text{Expected value of}\\ \text{partnering with}\\ \text{paired agent}}} + \underbrace{F(x) V_{A_{\bar{t}}(x)}}_{\substack{\text{Expected value of}\\ \text{continued search}\\ \text{(no partnering)}}} \qquad (1)$$

We use $x^*_{\bar{t}}$ to denote the reservation value that maximizes $V_{A_{\bar{t}}}(x)$. Now consider an agent of type $\bar{t}-\epsilon$. If this agent is accepted by agents of type $\bar{t}$, then it necessarily uses the same Equation 1 to set its own reservation value (substituting $V_{\bar{t}}(x)$ by $V_{\overline{t-\epsilon}}(x)$). Thus, the optimal reservation value for this agent is the same as for the highest type agent (i.e., $x^*_{\bar{t}} = x^*_{\overline{t-\epsilon}}$). The same logic holds for all other agents of types belonging to the interval $[x^*_{\bar{t}}, \bar{t}]$. Since none of the agents of types $[x^*_{\bar{t}}, \bar{t}]$ accepts any other agents of types in the interval $[\underline{t}, x^*_{\bar{t}})$ then the process can be replicated, resulting with additional clusters. The complete segregation scenario is illustrated in Figure 1.

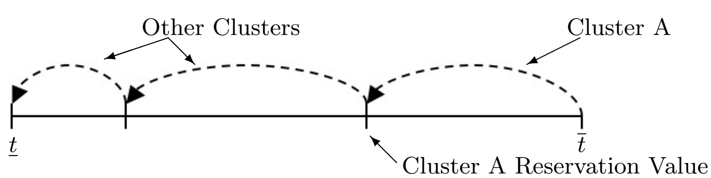


Fig. 1. Illustration of three clusters and the reservation value for the top cluster A.

The integration of mediated partnerships with the traditional two-sided search model introduces several new dynamics that must be added to the above analysis. The information the agents supply to each other is binding, therefore their acceptance strategy is reservation-value based. We begin by formulating the expected utility $V_{(\bar{t},M)}(x)$ of an agent of type $\bar{t}$ having a memory M when using a reservation value x, which can be formulated as

$$V_{(\bar{t},M)}(x) = \underbrace{-c}_{\text{Cost of search}} + \underbrace{\int_{y=x}^{\bar{t}} y f(y)dy}_{\substack{\text{Expected value of}\\ \text{partnering with}\\ \text{paired agent}}} + \int_{z=0}^{x} \left(\underbrace{\int_{y=x}^{\bar{t}} y G_{(M,z)}(y)dy}_{\substack{\text{Expected value of}\\ \text{being partnered by}\\ \text{mediated partnering}}} + \underbrace{\left(1 - \int_{y=x}^{\bar{t}} G_{(M,z)}(y)dy\right) V_{(\bar{t},M\cup z)}(x)}_{\substack{\text{Expected value of}\\ \text{continued search}\\ \text{with updated memory}}} \right) f(z)dz \tag{2}$$

where $G_{M,z}(y)$ is the probability of forming a mediated partnership with an agent of type y, initiated by the currently met agent or a perviously met agent. Notice that Equation 2 is an extensive modification of Equation 1 used for the traditional two-sided search.

Using similar logic as above, we obtain that all agents of types $[x^*_{\bar{t}}, \bar{t}]$ use the same optimal reservation value $x^*_{\bar{t}}$. For all agents of these types, the probability $G_{M,z}(y)$ receives a similar value. Thus clusters are formed in the same way as before, and each agent will only partner with other agents having a type in its own cluster.

The probability $G_{M,z}(y)$ is affected by the division of other agents into clusters. Agents of types that belong to small clusters are associated with longer searches and consequently are more likely to have a relevant agent in their memory upon being met. Therefore, unlike in the traditional model, each cluster in the new model depends on the number and the size of the other clusters. Hence, a backward induction solution as the one often suggested for the traditional two-sided search [10] is not applicable in our case.

4 Two Cluster Solution

Solving a set of expected utility Equations like the one given in equation 2 is impractical for the general case. In the general case, the probability of being partnered through mediated partnerships changes from one search round to another (based on the probability that other agents are still in the market). The extraction of the probability $G_{M,z}(y)$ requires understanding all the possible states the agent can be in and the transitions between them. In a finite state machine for an agent A_i, a state encapsulates information about which agents A_i holds in memory, and to which clusters they belong. The number of states require for an agent operating in an environment with N clusters is $2^{N-1}+1$, representing the combinations of having memory for the $N-1$ other clusters, and the end state where the agent has partnered and left the game[5]. The probability of transitioning from one state to another changes as the length of time the agents are kept in memory increases, because agents are likely to be partnered with other agents over time.

Fortunately, solving the problem analytically for environments where only two clusters are formed in equilibrium is possible. We will illustrate the two-cluster solution to show the dynamics of the system. In the two-cluster model the agent can have a memory of one agent in the other cluster. The agent can never have more than one agent in its memory, because having two un-partnered agents belonging

[5] Due to the unique cluster-base structure of the equilibrium, the agent will never have two agents from the same cluster in its memory, since these can always be matched.

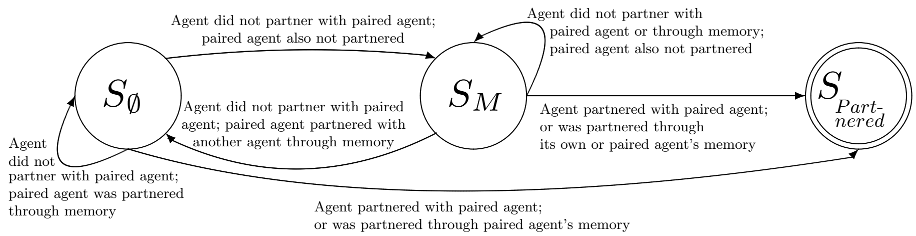


Fig. 2. Finite state machine for the two-cluster case of the two-sided search model with reputation.

to the same cluster would result in a partnering. Additionally, an agent will be in another agent's memory for at most one round of the search process. This derives from the fact that unless this agent is matched directly, it will be matched with whomever the agent storing it in memory is paired with the following search round (unless the storing agent is matched with its own kind, and leaves the market).

The finite state machine in Figure 2 explains the transitions of an agent in the two cluster solution using mediated partnering. In the two-cluster model, the finite state machine for an agent has three states: $S_\emptyset$ for the state where there is no agent in memory, S_M for the state where there is an agent of the other cluster in memory, and $S_{Partnered}$ for the state where the agent has joined with another agent of its own cluster and left the market.

Throughout the paper we will use the notation P_c to represent the probability of an agent randomly pairing with an agent in its own cluster, $\overline{P_c}$ to represent the probability of an agent randomly pairing with an agent of the other cluster, $P_{S_\emptyset}$ to represent the probability of a randomly encountered agent of the same cluster being in state $S_\emptyset$, and P_{S_M} to represent the probability of a randomly encountered agent of the same cluster being in state S_M. Similarly, we use $\overline{P_{S_\emptyset}}$ and $\overline{P_{S_M}}$ to represent the probability of a randomly encountered agent of the other cluster to be in states $S_\emptyset$ and S_M, respectively. Additionally we will use the term A_s to refer to a specific agent and A_p to refer to A_s's paired agent.

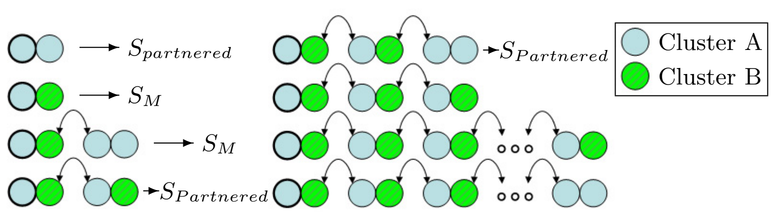


Fig. 3. Examples of how chains can form in a two cluster environment from the perspective of an agent in Cluster A with no memory. This agent with no memory is outlined with a bold circle. Arrows between agents signify a memory link. Arrows to states show that the agent in bold will transfer to that state with probability 1.

The pairing of agents with memory can lead to having chains of pairs of agents in which none of the pairs can directly join, however each agent in the sequence (except for the first and the last) has a memory link to another unjoined agent with whom its paired agent can join (see illustration in figure 3). We use L_s to denote the number of memory links in a sequence originating from the agent A_s. Similarly, we use L_p to denote the number of memory links in a sequence originating from the paired agent A_p. The probability for having exactly k links in a sequence originating from an agent that has memory can be calculated using

$$P(L_s = k) = \begin{cases} \overline{P_c} & \text{if } k = 0 \\ P_c(P_{S_\emptyset} + P_{S_M} P(L_s = 0)) & \text{if } k = 1 \\ P_c P_{S_M} P(L_s = k-1) & \text{if } k > 1 \end{cases} \tag{3}$$

$$P(L_p = k) = \begin{cases} P_c & \text{if } k = 0 \\ \overline{P_c}(\overline{P_{S_\emptyset}} + \overline{P_{S_M}} P(L_p = 0)) & \text{if } k = 1 \\ \overline{P_c P_{S_M}} P(L_p = k-1) & \text{if } k > 1 \end{cases} \tag{4}$$

The probability of an agent being partnered, or staying in the market follows from the Padovan [18] sequence defined as

$$Padovan(i) = \begin{cases} 1 & \text{if } i \in \{0,1,2\} \\ Padovan(i-2) + Padovan(i-3) & \text{if } i \in Z,\, i > 2 \end{cases} \tag{5}$$

In addition, we use $P_{partner}(L_p, L_s)$ to denote the probability of being partnered when having L_p and L_s memory links sequences originating from the agent and from his partner, respectively. The value of $P_{partner}(L_p, L_s)$ can be calculated using

$$P_{partner}(L_p, L_s) = \begin{cases} 0 & \text{if } L_p = 0, L_s = 0 \\ 1 & \text{if } L_p = 0, L_s = 1 \\ 1 & \text{if } L_p = 1, L_s = 0 \\ 0.5 & \text{if } L_p = 1, L_s = 1 \\ 1 - \frac{Padovan(L_s-2)}{Padovan(1+L_s)} = \frac{Padovan(L_s-1)}{Padovan(1+L_s)} & \text{if } L_p = 0, L_s > 1 \\ 1 - \frac{Padovan(L_p-2)}{Padovan(1+L_p)} = \frac{Padovan(L_p-1)}{Padovan(1+L_p)} & \text{if } L_p > 1, L_s = 0 \\ 1 - \frac{Padovan(L_p-2)Padovan(L_s-2)}{Padovan(1+L_p+L_s)} & \text{if } L_p > 1, L_s > 1 \end{cases} \tag{6}$$

Agent A_s will transition from state $S_\emptyset$ back to state $S_\emptyset$ if it cannot partner with agent A_p or an agent from A_p's memory, and the agent in A_p's memory is able to perform a mediated partnership to partner A_p. This probability is defined as

$$P_{S_\emptyset \to S_\emptyset} = (1 - P_c)\overline{P_{S_M}} \sum_{i=0}^{\infty} P(L_p = i)(1 - P_{partner}(i, 0))P_{partner}(0, i) \tag{7}$$

Agent A_s will transition from $S_\emptyset$ to the memory state S_M if it cannot partner with A_p, and neither A_s or A_p can be partnered through a mediated partnership. This transition probability is defined as

$$P_{S_\emptyset \to S_M} = (1 - P_c)\left(\overline{P_{S_\emptyset}} + \overline{P_{S_M}} \sum_{i=0}^{\infty} P(L_p = i)(1 - P_{partner}(i, 0))(1 - P_{partner}(0, i))\right) \tag{8}$$

Agent A_s will transition from $S_\emptyset$ to $S_{partnered}$ if it can partner with A_p, or if it can be partnered with an agent from A_p's memory. This transition probability is defined as

$$P_{S_\emptyset \to S_{partnered}} = P_c + (1 - P_c)\overline{P_{S_M}} \sum_{i=0}^{\infty} P(L_p = i)P_{partner}(i, 0) \tag{9}$$

Agent A_s will transtion from S_M to $S_\emptyset$ if it cannot partner with A_p or an agent from A_p's memory, but either A_s or the agent in A_p's memory is able to perform a mediated partnering and join A_p. This probability is defined as

$$P_{S_M \to S_\emptyset} = (1 - P_c)\,(A + B) \tag{10}$$

where

$$A = \overline{P_{S_\emptyset}} \sum_{i=0}^{\infty} P(L_s = i)(1 - P_{partner}(0, i))P_{partner}(i, 0)$$
$$B = \overline{P_{S_M}} \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} P(L_s = i)P(L_p = j)(1 - P_{partner}(j, i))P_{partner}(i, j))$$

Agent A_s will transition from S_M back to S_M if it cannot partner with A_p, and neither A_s or A_p can be partnered through a mediated partnership. This transition probability is defined as

$$P_{S_M \to S_M} = (1 - P_c)(C + D) \tag{11}$$

where

$$C = \overline{P_{S_\emptyset}} \sum_{i=0}^{\infty} P(L_s = i)(1 - P_{partner}(0, i))(1 - P_{partner}(i, 0))$$
$$D = \overline{P_{S_M}} \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} P(L_s = i)P(L_p = j)(1 - P_{partner}(L_p = j, L_s = i))(1 - P_{partner}(i, j))$$

Finally, agent A_s will transition from S_M to $S_{partnered}$ if it can partner with A_p, or if it can be partnered through a mediated partnership initiated by either A_p or the agent in A_s's memory. This transition probability is defined as

$$P_{S_M \to S_{partnered}} = P_c + (1 - P_c)(E + F) \tag{12}$$

where

$$E = \overline{P_{S_\emptyset}} \sum_{i=0}^{\infty} P(L_s = i)P_{partner}(0, i)$$
$$F = \overline{P_{S_M}} \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} P(L_s = i)P(L_p = j)P_{partner}(j, i)$$

The probabilities of an agent being in a specific state can be calculated using the above transition probabilities as follows:

$$P_{S_\emptyset} = P_{S_\emptyset} P_{S_\emptyset \to S_\emptyset} + P_{S_M} P_{S_M \to S_\emptyset}$$
$$P_{S_M} = P_{S_\emptyset} P_{S_\emptyset \to S_M} + P_{S_M} P_{S_M \to S_M}$$

Consequently, the expected utilities for an agent in the different states it may be in are defined as

$$V_{S_\emptyset} = -c + P_{S_\emptyset \to S_M} V_{S_M} + P_{S_\emptyset \to S_{partnered}} V_{S_{partnered}}$$
$$V_{S_M} = -c + P_{S_M \to S_\emptyset} V_{S_\emptyset} + P_{S_M \to S_{partnered}} V_{S_{partnered}}$$
$$V_{S_{partnered}} = E[cluster]$$

The above set of equations is solvable for any interval of agent types and any search cost. The expected performance of the agent is $V_{\emptyset}$, as this is its state upon entering the market. An equilibrium division into two-clusters is a clustering that results with (a) the expected utility for agents of type $\bar{t}$ (and all other agents in their cluster) using a reservation value x is equal to x; and (b) the expected utility of all the agents in the second cluster is less than $\underline{t}$.

5 The Mediated Partnering Effect

The integration of mediated partnering with the two-sided search model is meant to help market designers further reduce market friction and increase average overall agent utility. Market friction is the reason agents do not have complete information. In the two-sided search model, market friction is the search cost. If search cost is eliminated completely, then the unique competitive equilibrium has assortative joining, meaning that agents only select partners of their own type [2]. In the latter case, the overall social utility is maximized. The overall social utility can always be decomposed to the aggregate utility each individual agent gains through partnership and the accumulated cost along the agents' search. The first component is fixed, since the agents partner amongst themselves. The second component, is a parameter of the search cost and the division of agents into clusters (which affects the extent of agent search and thus the accumulated search cost). Similarly, when the search cost is significantly large, each agent is willing to accept all other agents, thus an immediate partnership is formed upon executing the first search round. By introducing the mediated partnering mechanism, we seemingly accelerate the agents' search process without adding any additional cost. Instead of having the ability to form partnerships based on single random interactions, each agent is now subject to additional mediated, targeted opportunities along its search. This reduces market frictions and according to economic theory should improve agents' performance (since less effort needs to be invested in costly search).

Intuitively, the method appears to improve performance since it reduces market frictions, thus market makers and agent designers may attempt to use it unquestionably. Nevertheless, as we demonstrate in the following paragraph, reducing market inefficiencies does not necessarily improve overall market performance. Using the analysis given in the former section we produce three contrasting examples to illustrate the effect that mediated partnering has over the system's overall performance. The first example uses a uniform distribution with agents having 128 distinct types ranging from 0 (inclusive) to 128 (exclusive), and with a cost of 32. The average performance (which is an indicator for the overall market performance) in this case when using mediated partnering is 20. This is an improvement in comparison to the average performance achieved in the traditional two-sided search model, with an average performance of 18.6. This effect of improvement is the most common result found when applying mediated partnering in a two-sided search model.

The second example is obtained using a search cost of 0.25 and a probability distribution with 0.75 of the agents having types uniformly distributed between 64 and 128, and the remaining 0.25 of the agents having types uniformly distributed between 0 and 64. The overall performance is 83.5 in the classical model, and reduced to 79.7 with the use of the new technique. Here, the reduction of market friction benefits some agents, but also hinders other agents. The agents that are hindered are those agents that initially were at the base of a cluster, but with the addition of mediated partnering fall into a lower cluster. These agents make up a small proportion of the distribution, but are significantly and negatively impacted by the use of mediated partnerships in such a manner that their reduced performance reduces the overall performance. Last, we consider the trivial case where the search cost c is zero (where all agents are accepting only agents of their own type) and the case where the search cost is prohibitively large (where only one cluster is formed). Here, the mediated partnering does not affect the system (nor any individual agent's) performance at all. The implications of the above are discussed towards the end of the paper.

6 Algorithmic Based Approach

Determining the number of clusters and cluster boundaries analytically for the general case is not feasible for two-sided search with mediated partnering. Fortunately, in most environments, the agent types are discrete rather than continuous (e.g., the number of available Gigabytes a server can offer to its partner in the dual backup application). Therefore, theoretically we could have tried all possible clustering combinations, and determined the equilibrium sets. Whether a set of clusters is in equilibrium will be based on the following rule: the expected utility of the agents in a cluster $(i...j)$ (i.e., the agents that use type i as their reservation value) should not exceed i and should not be smaller than $i-1$ (otherwise, each single agent of types $(i...j)$ would have an incentive to deviate from accepting all agents of type i and greater to a strategy of accepting all agents of types $i-1$ and greater or $i+1$ and greater, respectively). The expected utility of agents in the different clusters, in this case, can be extracted using simulation of the two-sided mediated partnership-based search in a specific environment given.

The main disadvantage of the above method, is the exponential number of simulation runs. We use $R(N)$ to denote the number of possible clusters division when having N unique agent types. $R(N)$ can be calculated as follows

$$R(N) = 1 + \sum_{i=1}^{N-1} R(i)$$

This is simply by analyzing the first (highest cluster) - it can either cover all types (i.e., 1) or can cover $i \in \{i = 1, ..., N-1\}$ subsequent types, in which case this should be multiplied by the number of clusters that can be formed with the remaining types $(R(N-i))$. Now we prove that $R(N) = 2^{N-1}$

Proof by induction. Base case: when $N = 1$, $R(1) = 1$. Inductive case: assume for $N = k$, $R(k) = 2^{k-1}$. We want to show that $R(k+1) = 2^k$.

$$R(k+1) = 1 + \sum_{i=1}^{k} R(i) = 1 + \sum_{i=1}^{k-1} R(i) + R(k) = 2R(k) = 2^k$$

By showing the base case and the inductive case we have shown that $R(N) = 2^{N-1}$ Q.E.D.

We have developed a more sophisticated algorithmic-based solution for the purpose of extracting the equilibrium set of clusters using simulation (see pseudocode below). According to our algorithm, all agents are first (step 1) initialized to accept only their own type (i.e., each type forms its own cluster). This is done using the method *generateClusters(clusters, lower, upper)*, which sets the reservation value of each cluster in the interval [lower, upper] to itself. The method *runSimulation* simulates an environment clustered according to the division given in the variable *clusters* and returns the expected utility of the agents in each cluster. Then, for the cluster with the highest type that is not satisfied[6] (step 4), decreases its lower bound to accept the next lowest type (step 5), and initializes all agents below this type to accept only their own type (using *generateClusters* in step 6). The process repeats until all clusters are satisfied or there exists a single cluster that covers the entire range of types.

Algorithm 1: CLUSTER(A, T, c) — Calculate optimal equilibrium clustering.

Input: The set of all agents A, a vector of all distinct types T, the search cost c.
Output: A Vector of clusters ordered from the cluster with the highest reservation value to the lowest.

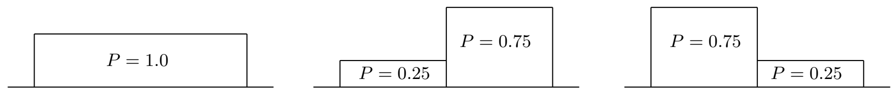
(1) generateClusters(clusters, T[0], T[T.length-1]);
(2) runSimulation(A, clusters, c);
(3) **while** (! satisfied(clusters))
(4)   Cluster s = satisfied(clusters);
(5)   s.lowest−−;
(6)   generateClusters(clusters, T[0], s.lowest-1);
(7)   runSimulation(A, clusters, c)
(8) **return** *clusters*

Theorem 1. *The clustering algorithm will always terminate in finite time and always result with an equilibrium (if at least one exists).*

Proof:
First we will show that given a newly updated reservation value r for a cluster, the expected performance of agents in clusters defined with reservation values greater than r is an upper bound to the equilibrium performance.

Let us denote agents with types greater than or equal to r as $A_{t \geq r}$, and agents with types less than r as $A_{t<r}$. After r is updated, all clusters with types below r are set so that their reservation value is equal to the maximum acceptance value, meaning agents of only one type are in each cluster. The probability of an agent $A_{t \geq r}$ meeting an agent $A_{t<r}$ with memory is $\int_{i=t}^{i<r} P(i) P_{M,i^*}$, where i^* is the reservation value of an agent with type i and P_{M,i^*} is the probability that an agent in a cluster with a reservation value of i^* will have memory. If any cluster with reservation value i, $i < r$, is expanded to accept agents in $[i-1, i]$ then the probability of an agent with type i having memory is reduced such that $P_{M,i-1} < P_{M,i}$. This is because the probability of an agent with type i directly pairing with an agent of its own cluster is increased by $P(i-1)$, thereby increasing the probability that an agent with type i will meet an agent in its own cluster, leave the market, and be replaced with a new agent with no memory. Therefore, the probability of an agent $A_{t \geq r}$ meeting an agent $A_{t<r}$ with a memory with whom it can partner is reduced.

[6] A cluster is *satisfied* if the expected value of the agents in the cluster falls below the reservation value but above the next lowest type of the reservation value.



Num Iterations - Unif Distribution						
Cost \ Number of Types	16	32	48	64	80	96
0.25	0	0	0	32	40	370
0.5	0	0	24	42	366	72
1	0	16	32	48	196	603
2	8	21	38	84	272	242
4	10	25	42	85	158	274
8	12	47	49	114	128	166
16	13	48	45	61	108	123
32	14	30	46	62	78	94

Num Iterations - 25-75 Skew						
Cost \ Number of Types	16	32	48	64	80	96
0.25	0	0	12	16	46	100
0.5	0	8	28	24	57	3248
1	0	18	58	52	188	349
2	4	24	38	62	352	98
4	10	28	53	194	92	104
8	12	31	44	69	105	238
16	13	29	60	61	77	93
32	14	30	46	62	78	94

Num Iterations - 75-25 Skew						
Cost \ Number of Types	16	32	48	64	80	96
0.25	0	0	12	16	46	669
0.5	0	8	28	352	367	1106
1	0	18	65	271	340	456
2	4	66	91	173	332	339
4	19	26	81	229	239	398
8	18	32	93	125	199	322
16	15	34	49	78	95	141
32	14	30	46	62	78	94

Table 1. The three tables depict the number of iterations for a given number of distinct types and a cost for the uniform distribution. Sketches of each distribution have been placed above their corresponding table.

# of Types	Uniform	25-75 Skew	75-25 Skew	Theoretical
16	0-14	0-14	0-19	3.28×10^{4}
32	0-48	0-31	0-66	2.15×10^{9}
48	0-49	12-60	12-93	1.41×10^{14}
64	32-114	16-194	16-352	9.22×10^{18}
80	40-272	46-352	46-367	6.04×10^{23}
96	72-370	93-3248	94-1106	3.96×10^{28}

Table 2. Comparison of the number of iterations to reach an equilibrium in three different two-sided search environments to the theoretical number of possible clusterings.

Therefore, if a cluster is not satisfied (the expected value of agents in the cluster falls below the next lowest type of the reservation value) the clustering of agents in $A_{t<r}$ in an attempt to increase the average agent performance in this cluster will not succeed. Similarly, combinations of clusters with reservation values greater than r cannot be modified and continue to be satisfied since these combinations were eliminated in former algorithm rounds. This leaves only the reservation value to be reduced to satisfy the cluster. Q.E.D.

We executed the algorithm varying the number of agent types and the distribution of types in the environment, measuring the number of simulations required before reaching a clustering in equilibrium. The results are shown in Table 1. Here, the search cost was varied between 0.25 and 32, and the number of distinct types varied between 16 and 96. The smallest type used was 0, and the largest 128.

We used three arbitrary distributions for illustration purposes, and found no significant differences as the number of iterations, search cost and number of distinct types were varied.

The number of iterations required for all scenarios tested was well below the theoretical 2^{N-1} maximum combinations, where N is the number of distinct types. As reflected in Table 2, the number of simulation runs required for extracting the equilibrium using the proposed algorithm is significantly smaller in magnitude.

7 Related Work

The two-sided search process is practically pairwise partnership formation application and can be related to the broader coalition formation domain [16, 15, 20, 21]. Nevertheless, coalition formation literature commonly assumes that agents can scan as many agents as needed (with no associated cost) or can make use of a central matcher or middle agents [7].

The analysis of two-sided search models is mostly found in economic search theory. These models can be distinguished according to several assumptions made. The first, is the payoff for each agent associated with a given partnership. While some of these models assume that the utility is a function of the other agent's type exclusively [10, 4], others assume a function defined over both types [1]. The second is the way according to which the search friction (cost) is modeled. This can be either the discounting of future flow of gains [4] or additive explicit search costs [10, 6, 1]. Lastly, the models are distinguished by the nature of the utility earned by each of the agents (transferable [1] and non-transferable [4, 6]). Our model was developed for autonomous software agents, operating in dynamic fast-paced environments. Therefore it assumes non-transferable utilities, explicit search costs and a payoff that depends on the other agent's type exclusively. For this model, it has been shown that in equilibrium, the agents perform a complete segregation[7] [10].

None of the above two-sided search models have made use of information agents collect throughout their search. The concept of matchmakers does arise occasionally in this literature, however always in the form of centralized rather than a distributed mechanism [3]. Mechanisms that made use of formerly collected "reputation" information can be found in MAS literature.

Reputation literature in AI focuses on reputation models in which agents classify the potential worth of agents through many rounds of direct and indirect interaction [19, 9]. Agents in these models usually form pairs, complete a task, and then at a later time period have the option of partnering with the same agent again. An agent generally judges the expected utility of joining with another agent using statistical models and information gathered from many agents [9, 22, 11, 19]. However, none of these works suggested a two-sided search model in a costly environment or presented results based on equilibrium agent strategies. Our model's use of mediated partnerships is similar to a form of reputation known as witness reputation. Witness reputation is information reported by an agent about its direct interactions with another agent [12]. In general, reputation information is noisy (potentially deceitful) and is generated from a history of interactions. The form of witness reputation we used assumes that agents gather exact information from only one interaction and that agents do not deceive when revealing their type or attempting to form mediated partnerships.

Very little work has been done in the AI reputation literature with measuring the cost of reputation data or finding beneficial agents [8]. Instead, most work only associates a reward for completing tasks in marketplace environments [19, 14, 22] or social networks [11]. Our work does not associate a cost directly with reputation data, but rather the cost of search. Other research [14] looks into the truthfulness of reputation data and provides models for judging the validity of reputation information.

8 Discussion, Conclusions and Future Work

The introduction of mediated partnering to two-sided search is an important extension whenever applying two-sided search in MAS. In contrast to the traditional two-sided search model, where formerly collected data has no value, mediated partnering allows a non-direct partnering based on agents' former experiences in the

[7] Complete segregation can also be found in many other model variants [1, 3, 6].

market. This approach fits well with autonomous agents that can easily support extensive memory storage and immediate costless interaction with formerly encountered agents. The proposed extension to the agents' search has not appeared in either economic search theory or MAS research, despite the fact that similar mechanisms in which agents use formerly collected data to improve system performance have been suggested (e.g., reputation systems and witness reputation).

As illustrate throughout our analysis, the integration of mediated partnering to two-sided search is not straightforward, and should be done with great caution. Despite the many advantages of witness reputation, in some settings it worsens overall performance (measured as aggregated social welfare). This is both surprising and important. It is surprising because we would have expected that reducing friction would improve efficiency and important because system designers should be careful not to automatically allow this option, but rather first calculate how reducing market friction can affect the performance of the system.

One important issue that ought to be discussed is the agents' incentive to use the proposed model. Here we distinguish between two aspects of the mediated partnering. The first is the agent's willingness to reveal its acceptance rule to the other agents it meets along its search. The agent will always benefit from revealing its acceptance rule since it enables the agent to be exposed to additional targeted partnering opportunities with no cost. The second is the incentive of agents to initiate the mediated partnering. Indeed, the agent is indifference between initiating the mediated partnering and simply resuming its search (since there is no direct benefit from doing this). However, an incentive can be easily created externally by enforcing the mediated partnering protocol by the market maker / agent designer, paying some of the market surplus to the agent in order to initiate mediated partnering or simply fining those agents that deviate from this protocol.

While the new model is commonly favorable, it adds a significant computational complexity to the equilibrium analysis. This prevents an analytic solution and requires approximation through appropriate algorithms and heuristics that embed simulation. The algorithm we present for this purpose in Section 6 suggests exceptional performance improvement within any range of agent types.

Finally, the attempt to integrate "search theory" techniques with day-to-day applications brings up the question of applicability. Justification and legitimacy considerations for this integration were discussed in the literature we referred to throughout the paper. This paper is not focused on re-arguing applicability, but rather on the improvement of the core two-sided search model.

This research is a first and important step towards developing far more advanced information sharing mechanisms for enhancing two-sided search applications in MAS environments. Extensions are numerous: from gossip (sharing all information available to the agent) to information bargaining (trading with the collected information with other agents).

9 Acknowledgments

We thank Barbara J. Grosz for her helpful comments and discussions. The work reported in this paper was supported in party by NSF grant CNS-0453923.

References

1. A. E. Atakan. Assortative matching with explicit search costs. *Econometrica*, 74(3):667–680, 05 2006.
2. G. Becker. A theory of marriage. *Journal of Political Economy*, 81:813–846, 1973.
3. F. Bloch and H. Ryder. Two-sided search, marriages, and matchmakers. *International Economic Review*, 41:93–115, 2000.
4. K. Burdett and M. G. Coles. Marriage and class. *The Quarterly Journal of Economics*, 112(1):141–68, February 1997.
5. K. Burdett and R. Wright. Two-sided search with nontransferable utility. *Review of Economic Dynamics*, 1(1):220–245, January 1998.
6. H. Chade. Two-sided search and perfect segregation with fixed search costs. *Mathematical Social Sciences*, 42(1):31–51, July 2001.
7. K. Decker, K. Sycara, and M. Williamson. Middle-agents for the internet. In *Proc. of IJCAI*, 1997.
8. P. Hendrix and B. Grosz. Reputation in the venture games. *National Conference on Artificial Intelligence (AAAI)*, July 2007.
9. T. D. Huynh, N. Jennings, and N. Shadbolt. Fire: An integrated trust and reputation model for open multi-agent systems. *Proceeding of 16th European Conference on Artificial Intelligence*, pages 18–22, 2004.
10. P. Morgan. A model of search, coordination, and market segmentation. mimeo, SUNY Buffalo, 1995.
11. M. Pujol, R. Sanguesa, and J. Delgado. Extracting reputation in multi agent systems by means of social network topology. *Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Ssytems: Part 1*, pages 467–474, 2002.
12. J. Sabater and C. Sierra. Reputation and social network analysis in multi-agent systems. *Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems: Part 1*, pages 475 – 482, 2002.
13. D. Sarne and S. Kraus. Time-variant distributed agent matching applications. In *Proceedings of AAMAS-2004*, pages 168–175, NY, 2004.
14. S. Sen, A. Biswas, and S. Debnath. Believing others: Pros and cons. *Proceedings of the Fourth Internation Conference on MultiAgent Systems (ICMAS-2000)*, page 279, 2000.
15. S. Sen and P. Dutta. Searching for optimal coalition structures. In *Proceedings of the Fourth International Conference on Multiagent Systems*, pages 286–292, July 2000.
16. O. Shehory and S. Kraus. Methods for task allocation via agent coalition formation. *Artificial Intelligence Journal*, 101(1-2):165–200, 1998.
17. R. Shimer and L. Smith. Assortative matching and search. *Econometrica*, 68(2):343–370, 2000.
18. I. Stewart. Tales of a neglected number. *Scientific American*, 274(6):102–103, June 1996.
19. W. Teacy, J. Patel., N. Jennings, and M. Luck. Coping with inaccurate reputation sources: Experimental analysis of a probabilistic trust model. *Proceedings of Fourth International Joint Conference of Autonomous Agents and Multiagent Systems*, pages 997–1004, 2005.
20. N. Tsvetovat, K. Sycara, Y. Chen, and J. Ying. Customer coalitions in electronic markets. In *Proceedings of AMEC 2000*, pages 121–138, Barcelona, 2000.
21. J. Yamamoto and K. Sycara. A stable and efficient buyer coalition formation scheme for e-marketplaces. In *Proceedings of the Fifth International Conference on Autonomous Agents*, pages 576–583, Montreal, Canada, 2001.
22. G. Zacharia and Maes P. Trust management through reputation mechanisms. *Applied Artificial Intelligence*, 14:881–907, 2000.