A Stochastic Clustering Auction (SCA) for Centralized and Distributed Task Allocation in Multi-agent Teams

Kai Zhang, Emmanuel G. Collins Jr., Dongqing Shi, Xiuwen Liu, and Oscar Chuy Jr.

Abstract. This paper considers the problem of optimal task allocation for heterogeneous teams, e.g., teams of heterogeneous robots or human-robot teams. It is well known that this problem is NP hard and hence computationally feasible approaches must develop an approximate solution. This paper proposes a solution via a Stochastic Clustering Auction (SCA) that uses a Markov chain search process along with simulated annealing. The original developments are for a centralized auction, which may be feasible at the beginning of a mission. The problem of developing a distributed auction is also considered. It can be shown that if the distributed auction is such that the auctioneer allocates tasks to optimize the \textit{regional cost}, then the distributed auction will always decrease the global cost or have it remain constant, which provides the theoretical basis for distributed SCA. Both centralized SCA and distributed SCA are demonstrated via simulations. In addition, simulation results show that by appropriate choice of the parameter in SCA representing the rate of “temperature” decrease, the number of iterations (i.e., auction rounds) in SCA can be dramatically reduced while still achieving reasonable performance. It is also shown via simulation that in relatively few iterations (8 to 35), SCA can improve the performance of sequential or parallel auctions, which are relatively computationally inexpensive, by 6\%-12\%. Hence, it is complimentary to these existing auction approaches.
1 Introduction

Teams of heterogeneous agents, for example, teams of heterogeneous robots or human-robot teams are expected to be employed in a variety of applications [1, 2, 3]. The problem of coordinating these agents is known as the problem of task allocation. Current approaches to task allocation can be placed into five categories: 1) fully centralized approaches [4], 2) centralized auctions [5], 3) distributed auctions [1, 3, 6], 4) completely distributed approaches [7, 8], and 5) hybrid approaches of distributed auctions and emergent coordination [2, 3].

This paper contributes to methodologies for centralized and distributed auctions. Although the combinatorial auction provides an optimal centralized auctioning approach, it is well known to be NP complete [9] and hence becomes practically infeasible as the number of tasks and agents increase. This has led to the search for polynomial time algorithms that yield task allocations that approximate the optimal task allocation. Sandholm has suggested that probabilistic or stochastic algorithms may result in better approximation algorithms [9]. This paper presents such a stochastic algorithm for optimal task allocation; it is based on a recent algorithm that was successfully used to obtain nearly optimal clustering of planar shapes [10].

This research views optimal task allocation as optimal task clustering, i.e., determining the cluster of tasks to be assigned to each agent so that a global or regional cost function is optimized. The algorithm proposed here, called Stochastic Clustering Auction (SCA), uses a Markov chain search process along with simulated annealing to search the space of task clusters. The idea of task clustering is not new to the task allocation literature [3]. However, this research differs from previous clustering algorithms in that the clustering is directly based on optimizing a global or regional cost function by a stochastic approach. It will be shown via simulation that for centralized auctioning SCA can result in a nearly optimal task allocation while requiring reasonably small computational times.

As previously mentioned, centralized auctions are primarily useful for the beginning of a mission. However, due to communication and computational requirements, they may be impractical during the mission. Hence, a second goal of this paper is to use the SCA for distributed auctioning. It can be shown that if the distributed auction is based on what has been called opportunistic centralization [3], such that the auctioneer allocates task to optimize the regional cost, then the distributed auction will always decrease the global cost. This motivates the development of a distributed auction that uses SCA to optimize the regional costs.

The paper is organized as follows. Section 2 formulates the basic optimization problem for task allocation, provides a description of the Stochastic Clustering Auction algorithm, and discusses how the algorithm may be used for both centralized and distributed auctions. Section 3 presents simulation results for both centralized and distributed auctions. Finally, Section 4 presents conclusions.
2 Stochastic Clustering Auction for Centralized and Distributed Task Allocation

This section first presents the basic problem statement. It then describes the Stochastic Clustering Auction (SCA). After providing the fundamental theory of the regional cost, it is proposed that SCA be used to optimize the regional cost in a distributed auction.

2.1 Problem Statement

Let $\mathcal{H}$ denote a set of $k$ heterogeneous agents, and $\mathcal{T}$ denote a set of $n$ tasks, i.e. $\mathcal{H} = \{h_1, h_2, \ldots, h_k\}$ and $\mathcal{T} = \{t_1, t_2, \ldots, t_n\}$. Also, let $A$ denote the allocation, $\mathcal{A} = \{a_1, a_2, \ldots, a_k\}$, where $\bigcup_{i=1}^{k} a_i = \mathcal{T}$, $a_i \subseteq \mathcal{T}$ and the cluster $a_i$ is assigned to agent $h_i$. The cost associated with $A$ is given by either

$$C(\mathcal{A}) = \sum_{i=1}^{k} c(a_i) \quad (1)$$

or

$$C(\mathcal{A}) = \max_{i} c(a_i), \quad (2)$$

where $c(a_i)$ is the minimum cost of agent $i$ completing the set of tasks $a_i$. The problem is to solve the optimization problem $\min \ C(\mathcal{A})$. In practice the cost function in (1) might be used to represent the total distance traveled or the total energy expended by the agents, while the cost function in (2) might be used to represent the maximum time taken to accomplish the tasks.

2.2 Stochastic Clustering Auction (SCA)

SCA attempts to minimize the cost $C(\mathcal{A})$ using a Markov chain search process in the space of possible allocations. The basic algorithm was originally developed in [10]. The algorithm is in the class called Markov Chain Monte Carlo [11]. The essential mechanism of SCA is to start with an allocation $\mathcal{A}$ for $k$ clusters and to probabilistically reduce or hillclimb $C(\mathcal{A})$ by rearranging the tasks $\mathcal{T}$ among the clusters. The actual algorithm is described below.

2.2.1 Stochastic Clustering Auction

1. Partition the task set $\mathcal{T}$ into $k$ clusters to form an initial allocation $\mathcal{A} = \{a_1, a_2, \ldots, a_k\}$, where each cluster $a_i$ is an unordered subset of $\mathcal{T}$. 
2. Each agent \( h_i \in \mathcal{A} \) \((i = 1, 2, \ldots, k)\) uses a “constrained Prim’s Algorithm”\(^1\) (a greedy algorithm) to efficiently approximate the cost \( c_i(a_i) \) and submits its cost to the auctioneer. In this bid valuation stage, each cluster \( a_i \) becomes an ordered subset of \( T \). The auctioneer computes the global cost \( C(\mathcal{A}) \) using (1) or (2) and sets a high temperature \( T \).

3. The auctioneer rearranges the clusters using either a single move or a dual move.

   a. **Single Move:** Randomly select a task \( t_i \in a_t \) from agent \( h_t \). Assume that \( t_i \) is reassigned to agent \( h_j \), resulting in the new allocation \( \mathcal{A}'^{(j,i)} \) that has two modified clusters \( a_t^{(-i)} \) and \( a_j^{(i+1)} \). Assume that agent \( h_t \) computes \( c(a_t^{(-i)}) \) and for \( j = 1, 2, \ldots, k \) agent \( h_j \) computes \( c(a_j^{(i+1)}) \), which the auctioneer uses to compute the corresponding cost \( C(\mathcal{A}'^{(j,i)}) \) (based on (1) or (2)). The likelihood of the acceptance of the transfer of task \( t_i \) from agent \( h_t \) to agent \( h_j \) is given by

   \[
   P_S(i, j, k, T) = \frac{\exp(-C(\mathcal{A}'^{(j,i)})/T)}{\sum_{j=1}^{k} \exp(-C(\mathcal{A}'^{(j,i)})/T)}.
   \]  

   b. **Dual Move (Task Swap):** Randomly select two tasks in \( a_t(n) \) and \( a_m \), task \( t_i \) from agent \( h_t \) and task \( t_j \) from agent \( h_m \), and swap them, resulting in the new allocation \( \mathcal{A}'^{(m,i)} \) that has two modified clusters \( a_t^{(-i)} \) and \( a_m^{(i+1)} \). Assume that agent \( h_t \) computes \( c(a_t^{(-i)}) \) and agent \( h_m \) computes \( c(a_m^{(i+1)}) \), which the auctioneer uses to compute the corresponding cost \( C(\mathcal{A}'^{(m,i)}) \) (based on (1) or (2)). Then, the likelihood of swapping the two tasks is given by

   \[
   P_D(i, j, k, T) = \frac{\exp(-C(2)/T)}{\sum_{p=1}^{k} \exp(-C(2)/T)}.
   \]  

   where \( C(1) = C(\mathcal{A}) \), the cost before swapping, and \( C(2) = C(\mathcal{A}'^{(m,i)}) \), the cost after swapping.

4. If \( P_S(i, j, k, T) \) or \( P_D(i, j, k, T) \) falls into acceptance likelihood, the auctioneer accepts the proposal so that \( \mathcal{A}' \) is updated and the cost \( C(\mathcal{A}') \) is put on log. If \( P_S(i, j, k, T) \) or \( P_D(i, j, k, T) \) falls into rejection likelihood, the auctioneer declines the proposal and the auctioneer reserves the current configuration and goes back to Step 3.

5. If the auction evolution termination criteria is satisfied, i.e., \( T < T_{cut} \), where \( T_{cut} \) is some threshold temperature, then the auction is terminated and the final

---

1. This algorithm fixes the initial vertex with a single edge in Prim’s Algorithm [12], and hence, unlike Prim’s algorithm, is not guaranteed to be optimal.
2. Each cluster is treated as an unordered subset and is ordered in a later bid valuation stage.
3. This cost is computed using the constrained Prim’s algorithm during bid valuation stages.
A SCA for Centralized and Distributed Task Allocation in Multi-agent Teams

The allocation is \( \mathcal{A}^* = \mathcal{A}^{(j,l)}_i \) or \( \mathcal{A}^* = \mathcal{A}^{(l,m)}_{i,j} \) with final cost \( C(\mathcal{A})^* \). If the criteria is not satisfied, reduce \( T \), using \( T \leftarrow T/\beta \) where \( \beta > 1 \) and go to Step 3.

The above algorithm description only allows single moves and dual moves (task swapping). In the implementation of SCA used in this study, the algorithm alternated between single and dual moves. Simulation results (omitted for brevity) showed that when SCA alternates between single and dual moves it is more efficient than using exclusively single moves or dual moves.

In order to search for the global optimal, a simulated annealing method has been adopted. Similar to the seminal paper in [13], SCA starts with a high value of \( T \) and gradually reduces it in order to to make small variations in the task allocation while searching for the optimal allocation in \( \mathcal{T} \). Although simulated annealing and random search help avoid local minimum, SCA does not guarantee convergence to a global optimum. It can be shown that if the temperature \( T \) is gradually decreased, then the algorithm will converge to a global optimum as described in [11], but it is difficult to quantify the rate of this decrease. Hence, the primary practical value of using simulate annealing is to yield nearly optimal solutions in reasonably fast execution times rather than guarantee asymptotic convergence results.

### 2.3 Distributed Task Allocation

If all the mission tasks are given in \( \mathcal{T} \), then the SCA algorithm described above is a centralized auction. A centralized auction may make sense at the beginning of a mission, but as discussed in Section 1, due to limited communication and the computational cost of a centralized auction, it may not be feasible during the mission. Hence, once the mission begins, it is assumed that clustering must be performed in a distributed fashion in which each agent sequentially in a given (possibly random) order becomes the auctioneer. This approach of initializing a mission with a centralized auction and subsequently using a distributed auction is advocated in [6].

Assume that agent \( h_i \) is the auctioneer for the set of heterogeneous agents \( \mathcal{H}_i \) defined by \( \mathcal{H}_i = \{ h_p : p \in \mathcal{L} \} \) where \( i \in \mathcal{L} \subset \{ 1, 2, \ldots, k \} \). It follows that \( \mathcal{H}_i \subset \mathcal{H} \), and hence \( h_i \) is the auctioneer for a distributed auction. If the initial global allocation is given by \( \mathcal{A}^{(0)} = \{ a_1^{(0)}, a_2^{(0)}, \ldots, a_l^{(0)} \} \), then the set of tasks in the distributed auction is \( \mathcal{T}_D = \{ t \in a_p^{(0)} : p \in \mathcal{L} \} \subseteq \mathcal{T} \). An allocation associated with the distributed auction is given by \( \mathcal{A}_D = \{ a_p : p \in \mathcal{L} \} \) where the set of tasks in the allocation \( \mathcal{A}_D \) is \( \mathcal{T}_D \), i.e., \( \mathcal{T}_D = \{ t \in a_p : p \in \mathcal{L} \} \).

**Definition 1.** The **regional cost** associated with the agents \( \mathcal{H}_D \) and the tasks \( \mathcal{T}_D \) is given by

\[
C(\mathcal{A}_D) = \sum_{p \in \mathcal{L}} c_p(a_p) \tag{5}
\]

when the global cost \( C(\mathcal{A}) \) is defined by (1), and

\[
C(\mathcal{A}_D) = \max_{p \in \mathcal{L}} c_p(a_p) \tag{6}
\]

when the global cost \( C(\mathcal{A}) \) is defined by (2).
It can be proved that if the distributed auction is based on optimizing the *regional cost*, the global cost will either decrease or remain the same, which motivates basing distributed SCA on the optimization of regional costs. (The formal theorem statement and proof of this result are omitted for brevity.) The actual distributed auction using SCA can then proceed in a manner similar to the distributed auction described in [6]. In particular, each agent, sequentially or in random order, calls and clears one auction. Rounds are held repeatedly until a stable solution is reached. The auctioning process can recommence when a new task is obtained or when there is a substantial change in the existing costs.

3 Simulation Results

This section provides simulation results for SCA using the multi-agent routing problem, which is a standard test domain for agent coordination using auctions [3, 5]. The tasks in the multi-agent routing problem considered here are to visit targets and complete an assignment at that target with exactly one agent assigned per target. The goal is to minimize the total time to overall task completion. In the simulations agent heterogeneity was taken into account by assuming that the agents moved at differing velocities and also differed in their completion times at each target. Sequential (single-item) Auctions (SA) and Parallel Auction (PA) populated in centralized task allocation are compared [1, 2, 3, 5, 6]. In SA unsigned tasks are auctioned off one in every round, and they are assigned to the winner promptly until all the targets are owned by robots [3]. In PA, every robot bids on each target simultaneously until all the targets are bided off to robots [3].

Centralized auctioning is first considered. Then, the use of SCA in a distributed auction is demonstrated. For each simulation unless otherwise specified the following SCA parameters were used: initial temperature, \( T = 100 \); cooling schedule rate, \( \beta = 1.001 \); and termination temperature, \( T_{\text{cut}} = 20 \). Random scenarios involving 2 agents and from 5 tasks to 50 tasks in increments of 5 were considered. For each number of tasks 20 random scenarios were generated based on randomly distributed task sites on a 1000 m \( \times \) 1000 m area, random completion duration times for each agent at each task site in the interval \([0, 2000]\) s and random constant speeds for each agent in the interval \([0, 2]\) m/s. The initial robot positions are at \([100 0]\) and \([1000 0]\). The cost function is a minmax cost function (2) corresponding to the mission completion time.

SCA is evaluated for two cooling schedule ratios \( \beta \). As \( \beta \) increases, the annealing process cools down more rapidly, leading to faster algorithm completion time but less optimal results. “CI” denotes the mean cost improvement of SCA for the 20 scenarios over the minimum of the SA and PA costs. “iterations”\(^4\) is the average number of auction evaluations. Note that due to the randomness of the distributions the mean costs do not monotonically increase as the number of tasks increase.

Experiments (omitted for brevity) show that SCA with random initialization renders the lowest team cost of the auction methods evaluated; as expected, this is at

\(^4\) The “iterations” represent the bid evaluation cycle in Steps (3) and (4) of Section 2.2.1.
the expense of a longer runtime. However, the parameter $\beta$ enables SCA to trade-off optimality and computational performance. The performance of SCA with the parameter $\beta = 1.01$ enables it to render better performance than either SA or PA in relatively few (29 to 145) iterations.

Figure 1 shows the average CI when the SCA is initialized with the better of the SA or PA solutions (in terms of global cost). It is seen in Figure 1(a) that
initializations with SA or PA can reduce the iterations of SCA from the interval [290 1612] to [80 421], which results in a fourfold decrease in the communication load requirements. Furthermore, when the annealing suite is adjusted to speed up the convergence ($\beta = 1.01$) in Figure 1(b), an improvement of 6%-12% can be achieved on team performance in 8-35 iterations. Additional improvement can be achieved if the communication ability permits.

3.1 Distributed Auctions

To illustrate the efficacy of distributed auctioning using SCA, a symmetrical benchmark is considered involving 4 homogeneous agents and 32 initial task sites arranged with communications available between neighboring agents so that the globally optimal solution is for each homogeneous agent to have an identical pattern of site visitation. SCA is used in a centralized auction to compute the optimal task allocation $\{a_1, a_2, a_3, a_4\}$ in 1 iteration and a run time of 4.1 s. Subsequently, an additional task $t_d$ is added during the mission (to the upper right corner of Figure 2(a)).
Figure 2 shows the ability of the distributed auction to yield optimal results. The 4 distributed auctions were performed sequentially and had the following computational performance: 1) 3 iterations with a 17.4 s runtime, 2) 3 iterations with a 21.3 s runtime, 3) 4 iterations with a 21.3 s runtime, and 4) 4 iterations with a 21.3 s runtime. These results indicate that if the number of changes in the tasks or mission is small, distributed auctioning using SCA may not require many iterations (i.e., auctions) to converge to an optimal or near-optimal solution.

### 4 Conclusion and Future Research

In this paper we developed Stochastic Clustering Auction (SCA), a new auction algorithm based on the concept of stochastic clustering. SCA can be used for both centralized and decentralized auctions. When used for distributed auctions, it enables the auctioneer to allocate tasks to optimize the regional cost.

For centralized auctioning a suite of random scenarios was used to compare SCA with the sequential (single-item) auction and the parallel auction. SCA can improve upon these methods at the expense of greater communication and computational requirements. A positive aspect of SCA is that the choice of the cooling parameter $\beta$ enables a tradeoff between optimality and the communication and computational requirements. In practice SCA can be used to improve the task allocation obtained by these less optimal but computationally simpler auction methods when the agents have access to sufficient communication. Hence, SCA provides a mechanism for continuous auction improvement.

Future research will investigate mechanisms to speed up SCA and reduce the communication requirements for both central and distributed auctioning by only allowing certain tasks to be auctioned, e.g., those that have a high cost for the agent that has currently been assigned that task. SCA will also be extended to handle complex tasks [6], i.e., tasks that have interdependence. In addition, SCA will consider human risk by considering a constrained optimization problem (e.g., minimize total completion time subject to a constraint on human risk) and will consider human fatigue by allowing human agents to specify how many tasks and what kind of tasks they are willing to be assigned in the auctioning process. Experimental validation of these results is also underway.

**Acknowledgements.** This work was sponsored, in part, by the US Army Research Laboratory under Cooperative Agreement W911NF-06-2-0041. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the ARL or the US Government. The US Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

The first author thanks Yiming Wu for helpful discussions.

---

5 It should be noted that the distributed auction will converge to the optimal solution no matter what sequence of auctioneers is chosen as long as it is a repeating sequence. However, alternative sequences may take more time or iterations to converge.
References