AAAI Technical Report SS-12-03
Game Theory for Security, Sustainability and Health

Incentive Based Cooperation in Multi-Agent Auctions

Charles Pippin
Georgia Tech Research Institute
Georgia Institute of Technology
Atlanta, Georgia 30332

Henrik Christensen
Center for Robotics and Intelligent Machines
Georgia Institute of Technology
Atlanta, Georgia 30332

Abstract

Market or auction based algorithms offer effective methods for de-centralized task assignment in multi-agent teams. Typically there is an implicit assumption that agents are willing to cooperate and can be trusted to perform assigned tasks. Reciprocal collaboration may not always be a valid assumption. In cases where auctions are used for task allocation, without explicit revenue exchange, incentives are needed to enforce cooperation. An approach to incentive based trust is presented, which enables detection of team members that are not contributing and for dynamic formation of teams.

1 Introduction

As the use of robotic platforms increases, it will be important for robots with different sensors and capabilities to form dynamic teams and cooperate on tasks. Some example scenarios include dynamic allocation of robot teams to disaster locations, search and rescue operations, and target detection. The assignment of robots to tasks is known as the multi-robot task allocation (MRTA) problem (Gerkey and Mataric 2003). In a specific type of the MRTA problem, there are multiple robots and multiple sequential tasks, which are locations to be visited, with the goal being to assign a robot to each of the locations while minimizing the overall team cost. Gerkey and Mataric showed that the MRTA problem can be reduced to the well known optimal assignment problem from operations research (Kuhn 1955), which can be solved using linear programming methods.

However, centralized approaches to the task allocation problem can be a source for communications and processing bottlenecks in the system and a single point of failure (Ekici, Keskinocak, and Koenig 2009). Also, in dynamic environments it may not be practical to keep central nodes up to date with the current state of the environment and of other agents. Furthermore, centralized approaches, while able to find optimal solutions, may not scale as easily as a distributed system and are less practical when changes in a dynamic environment require frequent re-planning. Conversely, distributed approaches involving teams of robots operate using local state information. They can work on tasks in parallel, perform distributed sensing and operate in multiple locations at once. Furthermore, a team of robots adds redundancy to the system. Unfortunately, a tradeoff is that these teams must communicate and work together and uncertainty can exist regarding robots' intentions towards each other. For instance, a team member may have trouble cooperating due to communication errors, or because they are busy performing other tasks, or even because of conflicting goals (Arkin 1998).

Auction methods are a class of decentralized algorithms that solve this problem by splitting computation across multiple nodes and iteratively performing task assignments (Bertsekas 1990). The basic auction approaches to the task allocation problem assume that team members can be trusted to cooperate and have the goal of the team in mind (to reduce the overall cost) (Koenig, Keskinocak, and Tovey 2010). These algorithms serve as a mechanism for distributed task allocation and generally do not need to consider incentives because of the cooperation assumption. As such, these methods do not explicitly account for trust between team members, but assume that *a)* team members will participate in bidding on tasks that are presented to them and *b)* team members will attempt to perform tasks that are assigned to them. However, there are situations in which teams may be formed dynamically, without explicit cooperation built into the system. While the team may have the same common goal, the individual players may have different levels of interest in the cooperation. That is, some of the team members may place a higher utility on successful completion of tasks, while others are obligated to participate, but wish to conserve resources. In these situations, it is assumed that the non-cooperative agents will not attempt to sabotage operations, but are self-interested and may not fully cooperate either. Agents should prefer to participate in teams because this will allow them to assign tasks to others that might complete them more efficiently. However, this means that they will be required to assist others in return.

This paper will present a game theoretic approach for providing incentives to cooperation in multi-agent auctions that do not explicitly exchange revenue. This is accomplished using an observation based trust model for evaluation and partner selection. Each interaction between agents is modeled as a 2-player prisoner's dilemma (PD) game. This approach can be used to select team members for auctions by cooperating only with those agents that have cooperated effectively in previous interactions. The rest of this paper is organized as follows. In Section 2, we present the background and related work. In Section 3, we discuss the use of trust applied to an auction setting and consider incentives, using the PD game and social norm strategy from game theory. In Section 4 we perform experiments using this approach and in Section 5, we summarize our findings.

2 Background

In collaborative multi-agent environments, robots are often explicitly designed to work together in teams. However, there may be situations in which self-motivated agents may benefit by cooperating with other agents when they do not share common goals, team structure or have cooperation built-in. An overview of incentives for cooperation in these types of systems is provided by (Kraus 1996). Such incentives include the use of contracting through monetary schemes and the exchange of credits between systems and their owners, as well as through bartering. Bartering depends on agents needing assistance from each other and may not work well when one agent can provide help and does not need any help itself, or in situations when agents may not be available in the future. Auction based approaches express tasks and costs in terms of a common utility. The use of incentives in contracting can inform the use of incentives for cooperation in auctions. For instance, (Kraus 1996) shows that the use of monitoring a task's completion can improve an agent's utility when it is risk averse.

The use of monitoring and incentives can also be used in a decision-theoretic approach to mitigate risk in agent interactions (Burnett, Norman, and Sycara 2011). Models of trust can be maintained about potential team members based on repeated interactions and these models can be used to calculate expected utility decision trees for cooperation with other agents. Ahn, DeAngelis, and Barber further investigate reputation with the concept of multi-dimensional trust (Ahn, DeAngelis, and Barber 2007). Trust can be described by different characteristics, such as quality, reliability and availability. They show that modeling trust with multiple dimensions can lead to greater agent rewards. Game theory approaches are used to perform dynamic team formation in network routing problems in (Blanc, Liu, and Vahdat 2005), (Jaramillo and Srikant 2010), (Srivastava et al. 2005) and (Baras and Jiang 2005). Cohen describes the use of incentives using the Tit-for-Tat strategy for improving robustness in the peer to peer file sharing network BitTorrent (Cohen 2003).

Kandori presents the social norm strategy as an approach to the random matching game for situations when agents may not interact with the same partner repeatedly, but perform interactions within a society. Kandori shows that with the addition of a reputation mechanism, community enforcement of social norms provides sufficient incentives for cooperation (Kandori 1992). Blanc et al (Blanc, Liu, and Vahdat 2005) applied Kandori's social norm to the peer-to-peer routing task. This paper combines incentives from the game-theory literature, particularly Kandori's social norm strategy, with auction based algorithms for providing incentives to cooperation in multi-agent auctions. The concepts of trust and reputation are also used to model direct and indirect observations that are used to create a distributed reputation authority for use by the social norm strategy.

2.1 Motivation

In traditional multi-agent systems approaches, each team member explicitly operates as part of a team and has the team's goals either explicitly or implicitly encoded. Future robotic teams may have different internal goals as well as configurations, quality levels, costs, operational capabilities, owners, and concept of operations. These robots will require mechanisms to dynamically form themselves into teams. Such teams may need to learn which team members are trustworthy and dynamically adjust their team composition accordingly.

There are many different methods for performing distributed cooperation, including centralized optimization algorithms and game theoretic techniques. However, auction based algorithms generally have low communication requirements (agents coordinate tasks through bid messages), and therefore are well suited to environments with communication constraints. Multi-robot auctions can perform computations in parallel and the methods take advantage of the local information known to each agent (Dias and Stentz 2000). For instance, an unmanned aerial vehicle (UAV) would not need to communicate a low fuel state to the entire team for allocating tasks, but could implicitly include this knowledge in their own task selection through cost-based bidding. Finally, these approaches are also amenable to standardization and cooperation across teams, as heterogeneous teams that are dynamically formed need only implement the auction messages in order to cooperate.

Examples of robots explicitly formed into cooperative teams using auction approaches are seen in the multi-robot mapping (Zlot et al. 2002), coordinated box-pushing (Gerkey and Mataric 2002), and Mars rovers (Schneider et al. 2005) domains; as well as in simulated UUVs (Sariel, Balch, and Stack 2006), and UAVs (Ryan et al. 2007). In many of these market-based approaches, self-interested robots operate in a virtual economy and exchange goods (information, task performance, etc.) for virtual revenue, which is not necessarily exchanged. While each agent seeks to improve their virtual profit, the entire team benefits from the cooperation. However, there are market-based schemes that use the actual exchange of virtual currency to provide incentives to cooperation. Currency exchange mechanisms require agents to share a common valuation, to keep accounting of interactions and to have a secure mechanism for performing the currency transfer. We assume that in some situations, there may not be a suitable mechanism in place for currency exchange. Therefore, in this work, while costs and rewards use the same basis for calculation, no revenue is actually exchanged between agents. As such, auctions are presented here as a mechanism for partner selection, based on agents' submitted estimates for performing a task. This work seeks to explore methods in which these approaches can additionally include incentives for cooperation.

3 Analysis

3.1 Dimensions of Trust

Trust and reputation (shared trust) mechanisms can be applied to auction algorithms for determining dynamic team formation. This work investigates the use of observation based trust and game theory mechanisms for determining when to remove a non-cooperative team member from an auction team by ignoring its auction requests. If a robot is no longer on a team, it loses opportunities for others to assist it with tasks when those tasks could be done more efficiently as part of a team than alone. In the auction context, robots that do not bid on each other's tasks can be viewed as non-cooperative and removed from a team. From a robot's viewpoint, it is better to have team members that cooperate and participate in the auction algorithm as this leads to more efficient outcomes. From a global viewpoint, it is desirable to have an efficient team that is composed of cooperative members; each non-cooperative member decreases the overall team performance. Therefore, it is desirable to perform dynamic team formation by allowing team members to perform auctions only with other cooperative team members.

In a dynamically formed auction team, agents may encounter other agents for which they have no prior experience. The use of a trust model would allow for an agent to reason about other agents' trustworthiness using observation histories and reputation information. In these settings, there are multiple dimensions that could be used to define trust, such as auction participation and task completion. This work will consider participation in auctions to illustrate the use of incentives for cooperation. However, additional trust dimensions could also be applied to this framework.

3.2 Trust Model

This work incorporates the use of the trust mechanism from (Teacy et al. 2006) for incorporating direct trust and reputation into a probabilistic formulation. This mechanism provides not only a trust belief about an agent, but also a confidence. The approach uses the beta probability distribution function and can incorporate positive (α) and negative (β) histories to calculate the belief and confidence. Each agent maintains a set of α and β vectors that represent the histories of interactions with each team member. Regarding auction participation, when a agent within range is sent an auction announcement and they do not respond with a bid, this is counted as a β observation while a bid response is counted as an α observation. An agent is initially trusted until sufficient β observations cause the trust value to be low, with high confidence.

We also use this mechanism to incorporate the reputation information (indirect observations) from other trusted team members using the same approach. However, the shared reputation information must be combined with the locally observed trust vectors. In our auction framework, each agent regularly posts their trust model's α and β vectors to all other team members that are within range. In addition, agents only incorporate those updates from other currently trusted team members. These shared, indirect observation vectors are easily integrated into the local vectors and the scalar trust and confidence values are recalculated.

Each time that an agent receives an auction message from another agent, they can evaluate the trust model to determine whether to participate. If the calculated trust value is less than the trust threshold, ϕ, and with confidence greater than γ, it is not trusted. However, a succession of positive observations (direct or indirect) can move an untrusted agent back to being trusted again. Furthermore, this approach is tolerant of noise as it can take multiple observations to move the value above or below the trust threshold.

3.3 Basic Auction Approach

In the basic multi-agent auction algorithm, the problem is to assign tasks to agents. In this paper, the tasks are to visit a target location and perform an observation. In the auction framework, each robot is a *bidder* and the items to be auctioned are the 'visits'. Each of the agents in the system also participates as an *auctioneer* and periodically auctions new task requests (it is assumed that the task requests are provided to the agent by an external process, such as a human operator or other event). This approach can easily be used on teams with different robot characteristics, as long as costs can be expressed in a common basis, such as time; each robot knows their own location and cost function and submits cost based bids to the auctioneer.

The approach followed by the *auctioneer* is shown in Procedure 1. The *auctioneer* first handles any auctions that have already been announced and are ready to close. This step is shown in detail in Procedure 2. In lines 1-3, the *auctioneer* selects the maximum bid from all bids received by the agents within communications range (including their own) as the winner of that auction and performs the task assignment by announcing the winning bidder. In lines 5 and 7, the *auctioneer* updates the trust model (described in Section 3.2) for each possible *bidder* that was sent the auction announcement. The trust model is referenced by the *bidder* in Procedure 3, when an auction announcement is received. If the originator of the auction announcement is not trusted, using the trust model, then the auction announcement is ignored, effectively isolating the untrusted agent from the benefits of cooperation.

In this paper, each target to be visited has a reward that is linearly decreasing with time (for example, consider a hurricane survivor scenario or forest fire scenario in which time to discovery is critical). The agents each maintain a current task list and locally compute their bid to complete the proposed task. For each auction announcement received, the agent calculates their bid as shown in Procedure 4. The surplus gain in unit time ($sgut$) is calculated as the change in surplus for inserting the task into the current task list. In this case, the bid consists of the surplus gain per unit time for them to perform the task, in addition to all of their other tasks, where surplus is defined as the total reward collected minus the total travel cost, as described by (Ekici, Keskinocak, and Koenig 2009). Each robot also incurs a small bidding cost with each bid. This represents the amount of computation and communication resources that need to be consumed to calculate and send the bid. The incremental travel cost is known as the cheapest insertion heuristic: for each pair of tasks in the current task list, the agent compares the additional Euclidian distance based cost for inserting the new task, and selects the insertion that maximizes its surplus gain, which forms the agent's bid. When the winning *bidder* is assigned a new task, the task is inserted into the agent's task list, again using the insertion heuristic.

Procedure 1 $Auctioneer :: PerformAuctions$

Input: The set of open auctions, A_{open}.
Input: The set of new task requests, $TaskRequests_{new}$.

for all $a : A_{open}$ **do**
  $HandleAuctionBids(a)$
end for

for all $a : TaskRequests_{new}$ **do**
  $Recipients_a \leftarrow AnnounceAuction(a)$
end for

$ReauctionRemaining(n, tasklist)$

Procedure 2 $Auctioneer :: HandleAuctionBids$

Input: An auction, a.
Input: The set of posted bids, B_a.
Input: The set of announcement recipients, $Recipients_a$.

$winner \leftarrow Max(B_a)$
$AnnounceWinner(winner, a)$
for all $a : Recipients_a$ **do**
  if $a \in B_a$ **then**
    $UpdateParticipation(TRUST_o, 1)$
  else
    $UpdateParticipation(TRUST_o, 0)$
  end if
end for

Procedure 3 $Bidder :: HandleAnnouncements(A)$

Input: An set of announced auction tasks, A.
Input: The auction originator trust model, $TRUST_o$.

for all $a : A$ **do**
  if $CanTrust(TRUST_o)$ **then**
    $bid \leftarrow CalculateBid(a)$
    if $bid > 0$ **then**
      $PostBid(bid)$
    end if
  end if
end for

3.4 Social Norm Strategy

At this point, basic concepts from game theory (Osborne 2003) can be introduced to show how incentives can be used to induce cooperation on auction teams. Consider the well known two player game from the game theory literature, the

Procedure 4 $Bidder :: CalculateBid(a)$

Input: An auction task, a.
Output: The agent's bid.

$[wri, dri] \leftarrow CalculateSurplus(tasklist)$
$[wri', dri'] \leftarrow CalculateInsertion(tasklist, a)$
$wri' \leftarrow wri' - BidCost$
$sgut \leftarrow (wri' - wri)/(dri' - dri)$
return $bid \leftarrow sgut$

Prisoner's Dilemma (PD), shown in Table 1. The payoff table reflects values of T for *temptation* to "defect", R representing the *reward* or for cooperation, P for *punishment* related to joint defections and S for *sucker* related to unilateral cooperation. The payoffs satisfy the following condition:

$$T > R > P > S \tag{1}$$

In a single round of play the rational player in PD should choose to defect. However, in repeated games, players will meet each other multiple times and can consider the history of their opponent's actions in determining an action. If there is a threat of punishment, then cooperation can be induced in repeated play. There are several strategies that can be used to induce cooperation in repeated play, such as Tit-for-Tat, which is discussed further below.

Table 1: Payoff Matrix for the Prisoner's Dilemma

	C	D
C	R, R	S, T
D	T, S	P, P

Cooperation on multi-agent teams can also be modeled using the PD game. In each round of an auction, players are matched by the rules of the auction and can choose to participate (cooperate) or not participate (defect). Here, it is assumed that players will be repeatedly matched against each other. The global team score will be better if all agents fully participate in auctions, not just when it suits their interests. For instance, it is possible for agents to take advantage of the auction setting to allow others to perform their tasks while not performing others' tasks in return. The disincentive to cooperate could be attributed to selfishness of uncooperative agents, agents that are overloaded with tasks have have nothing to offer, or agents that are incapable of effective participation. Each interaction in the auction setting can be treated as a two-player game.

The game is modeled as a prisoner's dilemma, where each interaction represents two separate auctions, one initiated from each player, shown in Table 1. The players cooperate by bidding on each other's auctions and defect by not submitting a bid or a bid that is valid. This game can be treated as a random matching game, because it is assumed that if the game is played for a long enough time horizon, each player will eventually have an opportunity to bid on the other's auctions. The payoffs in this game are as follows:

- $R = b - c$: Benefit (time discounted reward) when another agent completes a task minus the cost for performing a task for that agent.
- $T = b$: Benefit (time discounted reward) when another agent completes a task.
- $S = -c$: Cost for unilaterally performing a task on behalf of another agent.
- $P = 0$: There is no additional gain if neither player cooperates.

The Tit-for-Tat strategy can be useful for inducing cooperation, but it is sensitive to noise and does not allow for the agent that was defected against to quickly recover from defect losses. This strategy is also dependent on repeated interactions as part of the random matching assumption. However, there are situations in which agents interact but change partners frequently and may not have a chance to apply timely punishment after an interaction. A strategy that uses a community model for conveying trust is the social norm strategy as given by (Kandori 1992). The strategy requires that each agent is associated with a reputation label which is visible to all other agents in the community. The social norm strategy relies on a (generally centralized) reputation authority that observes pairwise interactions between players and assigns each player's label as either *Cooperator* or *Defector*. The social norm strategy also allows for the defected-against agent to recoup loses. Cooperation is sustained because the strategy allows other agents in the community to apply sanctions when a defection occurs. When 2 agents meet the social norm strategy dictates the following approach:

- If both agents are labeled *Cooperator*, they both cooperate.
- If both agents are labeled *Defector*, they both defect.
- If one agent is labeled *Defector*, then the *Defector* player should cooperate while the *Cooperator* player defects. This allows for the *Cooperator* player to recoup reward. The *Defector* player effectively "repents" through unilateral cooperation.
- Any deviation from the above strategy marks the deviator as a *Defector* for τ rounds.
- After τ rounds of following the above strategy, a *Defector* player is forgiven and becomes labeled *Cooperator* again.

3.5 Incentives for Cooperation

The social norm strategy for the PD game was shown by Kandori (Kandori 1992) to be a subgame-perfect equilibrium, if the agents use an appropriate discount factor, δ, and set the punishment period, τ, effectively. The discount factor reflects the willingness of the player in a repeated game to continue playing the game. A value of $\delta = 1$ reflects that the players are infinitely patient and expect the game to continue forever, while a $\delta \to 0$ means that agents prefer more immediate gains.

Table 2: Reputation Authority Probability Model

		$Cooperator$	$Defector$
RA label	$\widehat{Cooperator}$	x	0.10
	$\widehat{Defector}$	$(1 - x)$	0.90

Reputation Authority For the decentralized case, this work uses the distributed reputation mechanism, as described in Section 3.2, as the reputation authority that provides the labels for each of the players. Note that the distributed reputation authority relies on the combined direct and indirect observations in calculating an agent's label. This allows for a "sticky" reputation which is less sensitive to noise in the observations. While the social norm approach is still sensitive to noise (agents that do not bid can be counted as deviating from the strategy), the social norm approach allows for the defector agent to recover.

Voided Contract As mentioned above, the social norm strategy allows for the defected-against agent to recoup losses when a *defector* agent follows the strategy and cooperates while a *cooperator* player defects. However, we provide an extension to the strategy for use in auctions by performing additional punishment toward the deviator: any tasks in the *cooperator* agent's task list that originated with the *defector* agent are dropped. In doing so, the defected against agent effectively considers the cooperation contract 'voided' and is under no obligation to complete those tasks. This provides additional incentive for cooperation as the dropped tasks will not be completed and those rewards will therefore not be returned to the *defector* agent (however, the *defector* agent could elect to reclaim and execute the dropped tasks at presumably higher cost).

Probabilistic Forgiveness In practice, a reputation authority will likely contain a small amount of error in the classifications that it provides. If an estimate of the error probabilities for the distributed reputation authority is known in advance, then it is possible to calculate the probability of incorrect classifications using Bayes' rule. For instance, consider the example probability model for a reputation authority as shown in Table 2, and let $x = 0.80$. This model reflects the probabilities that 80% of the time, a *Cooperator* agent will be correctly labeled as $\widehat{Cooperator}$ by the reputation authority and that 90% of the time, a *Defector* agent will be correctly labeled as $\widehat{Defector}$.

The noise in the model could be due to multiple causes, including communication error, noise in the observation, and error in classification. In the case that a *Cooperator* agent is incorrectly labeled $\widehat{Defector}$, the incentives for cooperation can breakdown. However, given a model of the reputation authority, it is possible to calculate the probability that an agent is actually a *Cooperator*, given that the authority labeled it $\widehat{Defector}$, as shown in Equation 2. For instance, in this example, there is still a 34% probability that the agent is actually a *Cooperator*. In order to tolerate noise in the system, we can periodically reset the labels of some *Defector* agents, before the end of the τ punishment period, by sampling from this probability distribution. This allows for true *Cooperator* agents to return to cooperative behavior as they will see that others are again cooperating with them.

$$
\begin{aligned}
&P(Cooperator|\widehat{Defector}) \qquad (2)\\
&= \frac{P(\widehat{Defector}|Cooperator)P(Cooperator)}{P(\widehat{Defector})}\\
&= \frac{(0.20)(0.70)}{(0.20)(0.70)+(0.90)(0.30)} = 0.34
\end{aligned}
$$

4 Experimental Results

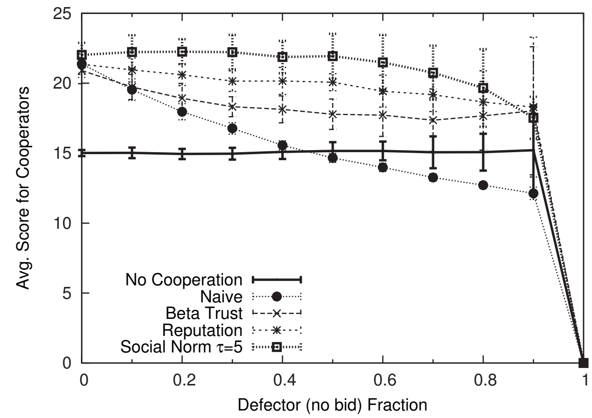
A set of experiments were performed in simulation to test the trust strategies in a multi-agent auction environment. In these experiments, the robots are represented by unmanned aerial vehicles (UAVs). Each UAV has 50 tasks that arrive at regular intervals and are sequentially auctioned by that UAV's auctioneer. As part of the auction process, they also bid on their own tasks. The UAVs in the simulation have a limited communications range and can therefore only perform auctions or exchange reputation information with a subset of the other team members at a given time.

In addition, each UAV periodically re-auctions the last n tasks to other agents in range. This allows tasks to be more optimally assigned by giving other agents a chance to bid on them if they were not in range during the initial auction. Rewards are given for task completion to the UAV that originated the task, and rewards decrease linearly with time until they reach 0. Each agent submits bids that represent the surplus gain per unit time for performing the additional task. Once a UAV finishes all tasks in their list, they no longer accumulate costs in the simulation. The initial locations of the UAVs and the tasks are randomly chosen for each iteration. For each set of experiments, results were averaged over 100 runs using 10 simulated UAVs.

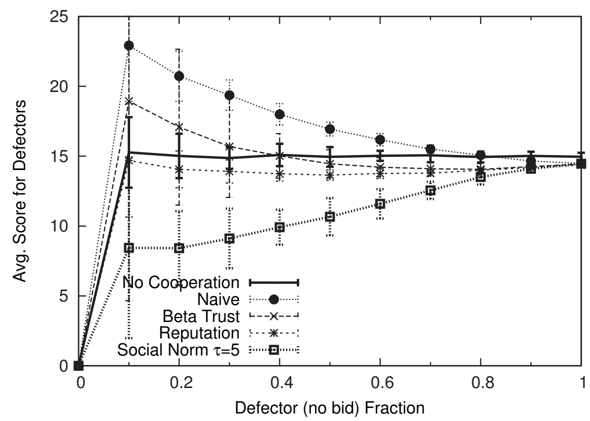
Detecting and Punishing Defectors In this set of experiments, a fraction of the agents on the team defect by not participating in auctions (not bidding on others' tasks). Each *Defector* agent only participates in auctions 10% of the time. As a result, naive agents (using no trust mechanism) end up doing additional work for the *defector* agents and receive nothing in return. The task for the *cooperator* agents is to detect those team members that regularly fail to participate in auctions and to isolate them from future cooperation by not bidding on the *defectors'* tasks.

The agents that use the social norm (SN) strategy can quickly punish and isolate the defectors from the team by no longer bidding on their auctions. The results of this experiment, shown in Figure 1(a), reflect that the agents running the SN strategy receive better scores than those using beta trust and reputation methods alone, even as the fraction of defectors increases. Finally, the beta trust, reputation and SN methods all perform better than the naive strategy which trusts all team members unconditionally.

For this same experiment, the average score for all of the *defectors* is shown in 1(b), for each of the strategies employed by the cooperative agents. Clearly, the *defectors* do well when the *cooperators* run the naive strategy. However, the *cooperators* running the SN strategy provide strong incentives for the *defectors* to cooperate (when the *cooperators* run the SN strategy, the *defectors* receive much lower scores than the *cooperators*).



(a) Average Score for *Cooperators*



(b) Average Score for *Defectors*

Figure 1: Agents that defect by not participating can be detected and isolated using observation based trust mechanisms. The *defector* fraction is plotted against the average unit score of the (a) *cooperator* agents and (b) *defector* agents for each trust strategy run by the *cooperators*. The error bars reflect one standard deviation.

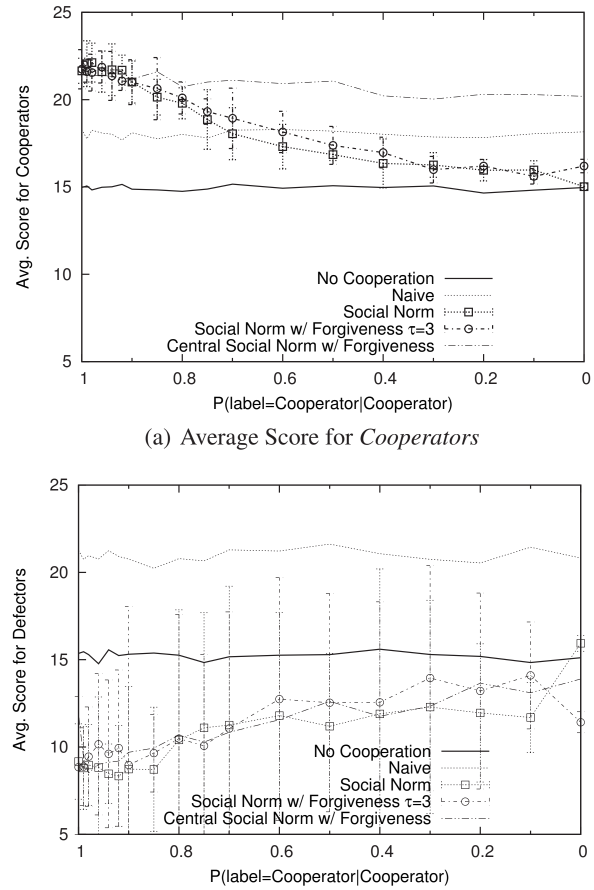
Noisy Reputation Authority In some cases, the SN strategy can cause *cooperator* agents to be punished unfairly. This can happen, as mentioned above, when the bid participation trust dimension is used and some agents do not submit bids because they cannot perform the task. In other cases, there may be noise in the reputation authority mechanism that marks some agents as defectors when in fact they cooperated or vice-versa.

In the following experiment, a noisy, decentralized reputation authority is compared against an accurate centralized reputation authority. The probability of incorrect label assignments by the reputation authority is show by the model in Table 2. With small probability, a *Defector* agent will be incorrectly classified as a *Cooperator*, but most of the time will be correctly labeled. The experiment decreases the probability x that a *Cooperator* agent will be correctly labeled.

The SN strategy with a centralized reputation authority provides the most favorable incentives for cooperation, resulting in the highest scores for *cooperators* and very low scores for *defectors*. However, in practice a central authority may not always be available and it may be necessary to rely on the decentralized authority. With the decentralized authority, when *Cooperator* agents are incorrectly labeled as *Defector*, this can lead to a breakdown of cooperation. However, the SN strategy allows for forgiveness through different settings for the punishment period, τ. In addition, we allow for probabilistic forgiveness, to account for incorrect labeling as described in Section 3.5. Here, 20% of the agents are actual *defectors*. The results, as shown in the scores for *cooperators*, Figure 2(a), and *Defectors*, Figure 2(b), indicate that the SN methods provide sufficient incentives for cooperation, even as the probability for an agent being incorrectly labeled is increased.

For both SN strategies, the scores for cooperation exceed the scores for defection, when the accuracy for correctly labeling *cooperator* agents is above about 75%. In this case, the use of these strategies removes any incentive to not cooperate. Additionally, as the accuracy for correctly labeling *cooperator* agents decreases below about 75%, the naive strategy results in better scores for *cooperators* than the SN strategies. This result is due to the unfair punishment of other *cooperators* because of the noise in labeling. As such, when noise levels in the decentralized reputation authority reach this threshold, it becomes worthwhile to improve the labeling accuracy or rely on a centralized reputation mechanism. In addition, because of the modeled noise in the system the use of probabilistic forgiveness with accuracy below 75% causes the defector agents to be forgiven more often. This results in the high standard deviation values shown for the defector scores in Figure 2(b).

This experiment suggests that the use of forgiveness can be used to sustain cooperation for predictable levels of noise in the system. After this point, the cooperators can achieve better scores by switching to the naive strategy, even if this means that they will be occasionally exploited.



(a) Average Score for *Cooperators*

(b) Average Score for *Defectors*

Figure 2: The SN strategy provides strong incentives for cooperation, even as the reputation authority mislabels *Cooperator* team members as *Defector*. (a) The average score for following the SN strategies exceeds the average score for defection (b), removing incentives for defection.

5 Summary and Future Work

Traditional auction algorithms for performing the robot task assignment problem assume that robots are equally incentivized to participate in auctions. However, there are situations in which agents may assign tasks to others on the team, without taking on a fair number of additional tasks in return. This paper presents an approach for using observation based trust and a shared reputation mechanism in determining which agents to include in multi-agent auctions. The experimental results show that by incorporating the use of trust strategies into the basic auction mechanism, agents can perform better than agents that trust unconditionally. Furthermore, the introduction of punishment through isolation from future auctions and through dropping already assigned tasks provides incentives for cooperation in multi-agent auctions that weren't present in traditional approaches.

Future work will consider additional trust dimensions relevant to auctions, including task completion and correctness. This is related to the problem of determining how to recognize when tasks that were assigned to another agent were completed and how to determine if they were completed to satisfaction.

6 Acknowledgements

The authors wish to thank Alan Wagner, Lora Weiss and the anonymous reviewers for their helpful comments on this work. This work was funded internally by the Georgia Tech Research Institute.

References

Ahn, J.; DeAngelis, D.; and Barber, S. 2007. Attitude driven team formation using multi-dimensional trust. In *Intelligent Agent Technology, 2007. IAT '07. IEEE/WIC/ACM International Conference on*, 229 –235.

Arkin, R. C. 1998. *Behavior-Based Robotics*. Cambridge, Mass., MIT Press. chapter 9.

Baras, J. S., and Jiang, T. 2005. Cooperation, trust and games in wireless networks. In Abed, E. H., ed., *Advances in Control, Communication Networks, and Transportation Systems*, Systems & Control: Foundations & Applications. Birkhauser Boston. 183–202.

Bertsekas, D. P. 1990. The auction algorithm for assignment and other network flow problems: A tutorial. *Interfaces* 20(4):pp. 133–149.

Blanc, A.; Liu, Y.-K.; and Vahdat, A. 2005. Designing incentives for peer-to-peer routing. In *INFOCOM'05*, 374–385.

Burnett, C.; Norman, T. J.; and Sycara, K. 2011. Trust decision-making in multi-agent systems. In *IJCAI 2011: AAAI Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence*.

Cohen, B. 2003. Incentives build robustness in bittorrent. *In Proceedings of the 1st Workshop on Economics of Peer-to-Peer Systems*.

Dias, M., and Stentz, A. 2000. A free market architecture for distributed control of a multirobot system. In *6th International Conference on Intelligent Autonomous Systems (IAS-6)*, 115–122.

Ekici, A.; Keskinocak, P.; and Koenig, S. 2009. Multi-robot routing with linear decreasing rewards over time. In *Proceedings of the 2009 IEEE international conference on Robotics and Automation*, ICRA'09, 3944–3949. Piscataway, NJ, USA: IEEE Press.

Gerkey, B. P., and Mataric, M. J. 2002. Sold!: Auction methods for multirobot coordination. *Robotics and Automation, IEEE Transactions on*.

Gerkey, B. P., and Mataric, M. J. 2003. Multi-robot task allocation: Analyzing the complexity and optimality of key architectures.

Jaramillo, J. J., and Srikant, R. 2010. A game theory based reputation mechanism to incentivize cooperation in wireless ad hoc networks. *Ad Hoc Netw.* 8:416–429.

Kandori, M. 1992. Social norms and community enforcement. *Review of Economic Studies* 59(1):63–80.

Koenig, S.; Keskinocak, P.; and Tovey, C. 2010. Progress on agent coordination with cooperative auctions. In *AAAI Conference on Artificial Intelligence*.

Kraus, S. 1996. An overview of incentive contracting. *Artif. Intell.* 83:297–346.

Kuhn, H. W. 1955. The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly* 2(1-2):83–97.

Osborne, M. J. 2003. *An Introduction to Game Theory*. Oxford University Press, USA.

Ryan, A.; Tisdale, J.; Godwin, M.; Coatta, D.; Nguyen, D.; Spry, S.; Sengupta, R.; and Hedrick, J. 2007. Decentralized control of unmanned aerial vehicle collaborative sensing missions. *American Control Conference* 4672–4677.

Sariel, S.; Balch, T.; and Stack, J. 2006. Empirical evaluation of auction-based coordination of auvs in a realistic simulated mine countermeasure task. *Proceedings of the International Symposium on Distributed Autonomous Robotics Systems* 197–206.

Schneider, J.; Apfelbaum, D.; Bagnell, D.; and Simmons, R. 2005. Learning opportunity costs in multi-robot market based planners. In *In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*.

Srivastava, V.; Neel, J.; Mackenzie, A.; Menon, R.; Dasilva, L.; Hicks, J.; Reed, J.; and Gilles, R. 2005. Using game theory to analyze wireless ad hoc networks. *Communications Surveys Tutorials, IEEE* 7(4):46 – 56.

Teacy, W. T. L.; Patel, J.; Jennings, N. R.; and Luck, M. 2006. Travos: Trust and reputation in the context of inaccurate information sources. *Journal of Autonomous Agents and Multi-Agent Systems* 12:2006.

Zlot, R. M.; Stentz, A. T.; Dias, M. B.; and Thayer, S. 2002. Multi-robot exploration controlled by a market economy. In *IEEE International Conference on Robotics and Automation*, volume 3, 3016 – 3023.