Coalition, Cryptography, and Stability: Mechanisms for Coalition Formation in Task Oriented Domains

Gilad Zlotkin
Center of Coordination Science, Sloan School of Management, MIT
Cambridge, MA 02139 USA
Jeffrey S. Rosenschein
Computer Science Department, Hebrew University
Givat Ram, Jerusalem, Israel

Abstract

Negotiation among multiple agents remains an important topic of research in Distributed Artificial Intelligence (DAI). Most previous work on this subject, however, has focused on bilateral negotiation, deals that are reached between two agents. There has also been research on n-agent agreement which has considered "consensus mechanisms" (such as voting), that allow the full group to coordinate itself. These group decision-making techniques, however, assume that the entire group will (or has to) coordinate its actions. Sub-groups cannot make sub-agreements that exclude other members of the group.

In some domains, however, it may be possible for beneficial agreements to be reached among sub-groups of agents, who might be individually motivated to work together to the exclusion of others outside the group. This paper considers this more general case of n-agent coalition formation. We present a simple coalition formation mechanism that uses cryptographic techniques for subadditive Task Oriented Domains. The mechanism is efficient, symmetric, and individual rational. When the domain is also concave, the mechanism also satisfies coalition rationality.

Introduction

In multi-agent domains, agents can often benefit by coordinating their actions with one another; in some domains, this coordination is actually required. In two-agent encounters, the situation is relatively simple: either the agents reach an agreement (i.e., coordinate their actions), or they do not. With more than two agents, however, the situation becomes more complicated, since agreement may be reached by sub-groups.

The process of agent coordination, and of reaching agreement, has been the focus of much research in Distributed Artificial Intelligence (DAI). The general term used for this process is "negotiation" (usually in the 2-agent case) [0; 0; 0; 0; 0; 0], and "reaching consensus" (in the n-agent case) [0; 0; 0]. Both approaches, though dealing with different numbers of agents, share one underlying assumption: the agreement, if it is reached, will include all relevant members of the encounter. Thus, even in the n-agent case where a voting procedure might enable consensus to be reached, the entire group will be bound by the group decision. Sub-groups cannot make sub-agreements that exclude other members of the group. Interesting variations on these approaches, which nonetheless remain bilateral in essence, are the Contract Net [0], which allows bilateral agreement in n-agent environments, and bilateral negotiation among two sub-groups discussed in [0].

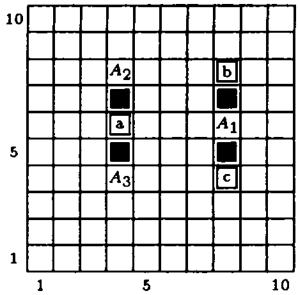
In some domains, however, it may be possible for beneficial agreements to be reached among sub-groups of agents, who might be individually motivated to work together to the exclusion of others outside the group. Voting procedures are not applicable here, because the full coalition may not be able to satisfy all its members, who are free to create more satisfying sub-coalitions. This paper considers this more general case of n-agent coalition formation (recent pieces of work on similar topics are [0; 0]). Building on the work of Zlotkin and Rosenschein [0], which dealt only with bilateral negotiation mechanisms, we here analyze the kinds of n-agent coordination mechanisms that can be used in specific classes of domains.

Coalitions

An Example—The Tileworld

Consider the following simple example in a multi-agent version of the Tileworld [0] (see Figure 1 on the left). A single hole in the grid is represented by a framed letter (such as $\boxed{\text{a}}$). Each agent's position is marked by its name (such as A_1). Tiles are represented by black squares (■) inside the grid squares.

Agents can move from one grid square to another horizontally or vertically (unless the square is occupied by a hole—multiple agents can be in the same grid square at the same time). When a tile is pushed into any grid square that is part of a hole, the square is filled and becomes navigable as if it were a regular grid square. The domain is static except for changes brought about by the agents.



Coalitions	u_1	u_2	u_3
{1},{2},{3}	0	0	0
{1,2},{3}	6	6	0
{1,3},{2}	6	0	6
{2,3},{1}	0	0	0
{1,2,3}	4	$6\frac{1}{2}$	$6\frac{1}{2}$

Figure 1: Three-Agent Encounter in the Tileworld and Possible Coalitions

Agent 1's goal is to fill hole $\boxed{a}$, while agents 2 and 3 need to fill holes $\boxed{b}$ and $\boxed{c}$ respectively. To fill its hole, each agent needs to do 7 steps. Agents 2 and 3 each need to travel 6 steps to reach the (8,6) grid position (the initial position of agent 1), and then A_2 pushes a block into the north hole while A_3 pushes a block into the south hole. Agent 1's best plan is to travel 6 steps to grid position (4,4) (A_3's initial position) or to grid position (4,8) (A_2's initial position), and then take a seventh step to push a block into its hole.

Agents can cooperate and help each other to reduce the cost of achieving their goals. Either agent 2 or 3 can achieve agent 1's goal with a single step. Agent 1 can achieve ether A_2's or A_3's goal with a single step, or both of their goals with 3 steps.

There are several kinds of joint plans that the agents can execute that will reduce the cost of achieving their goals. Some of those joint plans are listed in the table in right side of Figure 1.

Coalitions	u_1	u_2	u_3
{1},{2},{3}	0	0	0
{1,2},{3}	6	6	0
{1,3},{2}	6	0	6
{2,3},{1}	0	0	0
{1,2,3}	4	$6\frac{1}{2}$	$6\frac{1}{2}$

Table 1: Possible Coalitions in the Tileworld Example

The *coalition structure* {1,3},{2} means that there are two *coalitions*, one consisting of the agents 1 and 3, and the other consisting only of agent 2. When two agents form a coalition it means that they are coordinating their actions. The utility of an agent from a joint plan that achieves his goal is the difference between the cost of achieving his goal alone and the cost of his part of the joint plan [0].

In the case where no agents reach any agreement (First line in the Table at Figure 1), each agent needs to achieve its goal alone (and can do so in our scenario). Each agent carries out its original plan, and no utility is gained.

When agents 1 and 2 agree to help each other, and execute the joint plan where A_1 achieves A_2's goal (with a cost of 1) while A_2 achieves A_1's goal (with a cost of 1), they each achieve their own goals at a cost of 1, with a resulting utility gain of 6 (Second line in the Table at Figure 1). The third line of the table similarly shows the utility gain when agents 1 and 3 agree to help each other.

The fourth line in the Table at Figure 1 shows that agents 2 and 3 cannot gain any utility by coordinating their actions. The final line of the table shows that if all agents coordinate their actions, they can execute the joint plan where agent 1 achieves both 2 and 3's goals (with cost of 3), while either agent 2 or 3 achieves 1's goal (each with expected cost of $\frac{1}{2}$).

The coalition that gives the maximal total utility is the full coalition that involves all 3 agents, where they all coordinate their actions to mutual benefit (total utility is 17) [1]. Although this full coalition is globally optimal, Agent 1's utility is only 4, and he would prefer to reach agreement with either agent 2 or agent 3 (with utility of 6), but not with both.

The agents in the above scenario are able to transfer utility to each other, but in a non-continuous way. Agent 1, for example, can "transfer" to agent 2 seven points of utility by achieving his goal. He cannot, however, transfer an arbitrary amount. Without this arbitrary, continuous utility transfer capability, agent 1 will prefer to form a coalition with either one of the other two agents, rather than with both.

While the coalition structures given in Table at Figure 1 are exhaustive, with continuously transferable utility there are additional distributions of utility that can be put into effect. For example, we might have the coalition structure {1,3},{2}, and distribution (8, 0, 4) (where position i in the tuple signifies utility of agent i)—the maximal utility, of course, still sums to 12, which is the most that the coalition {1,3} can gain by coordinating their actions. Another possibility is the full coalition, with distribution (7, 7, 3). Note that agent 3 would be dissatisfied with the latter coalition and utility, and could attempt to cause the former coalition to arise (tempting away agent 1 with the promise of 8 points of utility). However, in the former coalition agent 2 is dissatisfied—and can attempt to cause the {1,2},{3} coalition structure to form with payoff (9, 3, 0). In contrast, the full coalition with utility distribution (9, 5, 3), while perhaps leaving agent 3 dissatisfied, does not allow him the opportunity of tempting away any other member of the coalition (e.g., to give agent 1 more than 9, agent 3 would have to get less than 3). We will see later that this utility distribution is *stable*.

Note also that, intuitively, agent 1 has a lot of power in the coalition formation—he seems to have more options, and his cooperation is valuable to the other

[1] The joint plan where agent 1 achieves both 2 and 3's goals (with cost of 3), while either agent 2 or 3 achieves 1's goal (each with expected cost of $\frac{1}{2}$

agents. If utility can be continuously transferred, it would be to agent 1's benefit, as can be seen from the above discussion.

Coalition Games

The definitions below are standard ones from coalition theory [0].

Definition 1 *A* coalition *game with transferable utility in normal characteristic form is* (N, v) *where:* $N = \{1, 2, \ldots, n\}$ *set of agents, and* $v: 2^N \rightarrow \mathbb{R}$. *For each coalition which is a subset of agents* $S \subseteq N$, $v(S)$ *is the* value *of the coalition* S*, which is the total utility that the members of* S *can achieve by coordinating and acting together.*

The Tileworld example from Figure 1 can be described as a coalition game (N, v) such that: $v(\{1\}) = v(\{2\}) = v(\{3\}) = v(\{2,3\}) = 0$, $v(\{1,2\}) = v(\{1,3\}) = 12$, and $v(\{1,2,3\}) = 17$.

Note that the value derived by a coalition is independent of the coalition structure. A given coalition is guaranteed to get a certain utility, regardless of what coalitions are formed by the other agents. In the Tileworld domain this assumption is not necessarily true—though it is true in the example we gave above. For example, if the number of tiles is less than the number of holes to be filled, the utility derived by a coalition might be affected by the actions of other agents outside of the coalition (e.g., whether they will use a tile the coalition wants).

In general, a coalition's utility in a State Oriented Domain [0; 0] can be influenced by the actions of other non-coalition members, and in particular by the kinds of coalitions those other members will form. We will see below that in Task Oriented Domains [0] this definition of the coalition value is directly applicable.

Task Oriented Domains

Definition 2 *A* Task Oriented Domain *(TOD) is a tuple* $< \mathcal{T}, \mathcal{A}, c >$ *where:* $\mathcal{T}$ *is the set of all possible tasks;* $\mathcal{A} = \{A_1, A_2, \ldots A_n\}$ *is an ordered list of agents;* c *is a* monotonic *function* $c: [2^{\mathcal{T}}] \rightarrow \mathbb{R}^+$. $[2^{\mathcal{T}}]$ *stands for all the finite subsets of* $\mathcal{T}$. *For each finite set of tasks* $X \subseteq \mathcal{T}$, $c(X)$ *is the cost of executing all the tasks in* X *by a* single *agent.* c *is monotonic, i.e., for any two finite subsets* $X \subseteq Y \subseteq \mathcal{T}, c(X) \leq c(Y); c(\emptyset) = 0$.

An encounter *within a TOD* $< \mathcal{T}, \mathcal{A}, c >$ *is an ordered list* $(T_1, T_2, \ldots, T_n)$ *such that for all* $k \in \{1 \ldots n\}, T_k$ *is a* finite *set of tasks from* $\mathcal{T}$ *that* A_k *needs to achieve.* T_k *will also be called* A_k*'s* goal.

The Postmen Domain [0] is one classic example of a TOD. In this domain, each agent is given a set of letters to deliver to various nodes on a graph; starting and ending at the Post Office, the agents are to traverse the graph and make their deliveries. There is no cost associated with carrying letters (they can carry any number), but there is a cost associated with graph traversal. The agents are interested in making short trips. Agents can reach agreements to carry one another's letters, and save on their travel.

In multi-agent Task Oriented Domains, agents can reach agreements about the re-distribution of tasks among themselves. When there are more than two agents, the agents can also form coalitions such that tasks are re-distributed only among the members of the same coalition. There will be no exchange of tasks among agents that belong to different coalitions; otherwise, we will consider the union of both coalitions as a single coalition. Given an encounter in a multi-agent TOD, agents need to decide which coalition to form, and how to redistribute the tasks among the members of the coalition. When mixed deals are being used by agents (those are agreements where agents settle on a probabilistic distribution of tasks), it can be useful to conceive of the interaction as a coalition game with transferable utility. The use of probability smooths the discontinuous distribution of tasks, and therefore of utility.

In other words, by choosing the probability appropriately, the agents can distribute utility among themselves continuously. However, utility is still not money in a classic TOD; utility is the difference between the cost of achieving your goal alone, and the cost of your part of the deal. Therefore, there is an upper bound on the amount of utility that each agent can get—no agent can get more utility than his stand-alone cost. As we shall see below, however, our model never attempts to violate this upper bound on utility.

Subadditive Task Oriented Domains

In some domains, by combining sets of tasks we may reduce (and can never increase) the total cost, as compared with the sum of the costs of achieving the sets separately. The Postmen Domain, for example, has this property, which is called *subadditivity*. If X and Y are two sets of addresses, and we need to visit all of them $(X \cup Y)$, then in the worst case we will be able to do the minimal cycle visiting the X addresses, then do the minimal cycle visiting the Y addresses. This might be our best plan if the addresses are disjoint and decoupled (the topology of the graph is against us). In that case, the cost of visiting all the addresses is equal to visiting one set plus the cost of visiting the other set. However, in some cases we may be able to do better, and visit some addresses on the way to others.

Definition 3 *TOD* $< \mathcal{T}, \mathcal{A}, c >$ *will be called* **subadditive** *if for all finite sets of tasks* $X, Y \subseteq \mathcal{T}$*, we have* $c(X \cup Y) \leq c(X) + c(Y)$.

A relatively minor change in a domain definition, however, can eliminate subadditivity. If, in the Postmen Domain, the agents were not required to return to the Post Office at the end of their deliveries, then the domain would not be subadditive.

Coalitions in Subadditive Task Oriented Domains

In a TOD, a group of agents (a coalition) can coordinate by redistributing their tasks among themselves. In a subadditive TOD, the way to minimize total cost is to aggregate as many tasks as possible into one execution batch (since the cost of the union of tasks is always less than the sum of the costs). Therefore, the maximum utility that a group can derive in a subadditive TOD is the difference between the sum of standalone costs and the cost of the overall union of tasks. This difference will be defined to be the value of the coalition.

Definition 4 *Given an encounter* $(T_1, T_2, \ldots, T_n)$ *in a subadditive TOD* $< \mathcal{T}, \mathcal{A}, c >$, *we will define the coalition game* induced *by this encounter to be* (N, v), *such that* $N = \{1, 2, \ldots, n\}$, *and* $\forall S \subseteq N$, $v(S) = \sum_{i \in S} c(T_i) - c(\bigcup_{i \in S} T_i)$.

According to the above definition the value of any single agent's coalition is zero. This means that in the case of total conflict, when no task is exchanged and the coalition configuration is ({1},{2}, ...,{n}), each agent gets 0 utility. We will call this coalition the conflict coalition.

Superadditive Coalition Games

It seems intuitively reasonable that agents in a coalition game should not suffer by coordinating their actions with a larger group. In other words, if you take two disjoint coalitions, the utility they can derive together should not be less than the sum of their separate utilities (at the worst, they could "coordinate" by ignoring each other). This property (which, however, is not always present) is called *superadditivity*.

Definition 5 *A* coalition *game with transferable utility in normal characteristic form* (N, v) *is* superadditive *if for any disjoint coalitions* $S, V \subset N, S \cap V = \emptyset$, *then* $v(S) + v(V) \leq v(S \cup V)$

Theorem 1 shows us that a superadditive coalition game always arises in a subadditive TOD.

Theorem 1 *Any encounter* $(T_1, T_2, \ldots, T_n)$ *in a subadditive TOD induces a* superadditive *coalition game* (N, v).

For any encounter $(T_1, T_2, \ldots, T_n)$ in a subadditive TOD we can conclude from Theorem 1 that all coalitions have positive value and that the full coalition has the maximal value.

Mechanisms for Subadditive TODs

We would like to set up rules of interaction such that communities of self-interested agents will form beneficial coalitions. There are several attributes of the rules of interaction that might be important to the designers of these self-interested agents (as discussed further by Rosenschein in [0]):

1. Efficiency: The agents should not squander resources when they come to an agreement; there should not be wasted utility when an agreement is reached. Since the coalition game is superadditive it means that the sum of utilities of the agents should be equal to $v(N)$.

2. Stability: Since the coalition game is superadditive, the full coalition can always satisfy the efficiency condition, and therefore we will assume that the full coalition will be formed. The stability condition then relates to the payoff vector $(u_1, u_2, \ldots, u_n)$ that assigns to each agent i a utility of u_i. There are three levels of stability (rationality) conditions: individual, group and coalition rationality. *Individual Rationality* means that that no individual agent would like to opt out of the full coalition; i.e., $u_i \geq v(\{i\}) = 0$. *Group Rationality (Pareto Optimality)* means that the group as a whole would not prefer any other payoff vector over this vector; i.e., $\sum_{i=1}^{n} u_i = v(n)$. This condition is equivalent to the efficiency condition above. *Coalition Rationality* means that no group of agents should have an incentive to deviate from the full coalition and create a sub-coalition; i.e., for each subset of agents $S \subseteq N$, $\sum_{i \in S} u_i \geq v(S)$.

3. Simplicity: It will be desirable for the overall interaction environment to make low computational demands on the agents, and to require little communication overhead.

4. Distribution: Preferably, the interaction rules will not require a central decision maker, for all the obvious reasons. We do not want our distributed system to have a performance bottleneck, nor collapse due to the single failure of a special node.

5. Symmetry: Two symmetric agents should be assigned the same utility by the mechanism (two agents are symmetric when they contribute exactly the same value to all possible coalitions).

How are these attributes satisfied in the coalition game that constitutes a multi-agent subadditive TOD? The criterion of efficiency requires that the full coalition be formed—nothing else guarantees efficiency (from Theorem 1). The requirement of stability means that the utility assignment for the full coalition should satisfy the following condition: there does not exist a group of agents all of whom can do better by forming a sub-coalition with one another.

We will develop a mechanism for subadditive TODs such that agents agree on the all-or-nothing deal, in which each agent has some probability of executing all the tasks. The question that we will try to answer now is "What should be the division of utilities among all agents in the full coalition?"

Coalition rationality is the strongest stability condition, and implies individual rationality and group rationality[2]. However, this condition is very strong,

[2] All payoffs that satisfy the coalition rationality conditions are called the *core* of the game in the game theory literature. See, for example, [0].

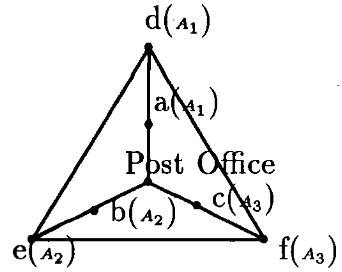


Figure 2: Example of an Unstable Encounter

and cannot always be satisfied.

Consider the encounter from a three-agent Postmen Domain that can be seen in Figure 2.

The Post Office is in the center. The length of each arch is 1. The encounter is $(T_1 = \{a, d\}, T_2 = \{b, e\}, T_3 = \{c, f\})$ [3]. Each agent can deliver his letters with a cost of 4 (i.e., $c(T_1) = c(T_2) = c(T_3) = 4$). The cost of delivering the union of the letters of any two agents is 5 (i.e.,$c(\{T_1 \cup T_2\}) = c(\{T_1 \cup T_3\}) = c(\{T_2 \cup T_3\}) = 5$).

Therefore, the value of any two agents' coalition is $(2*4) - 5 = 3$ (i.e., $v(\{1,2\}) = v(\{1,3\}) = v(\{2,3\}) = 3$).

The cost of delivering all the letters is 8. Therefore, the value of the full coalition is $(3*4) - 8 = 4$. We would like to find a payoff vector (u_1, u_2, u_3) that satisfies the following conditions:

(1) $u_1 \geq v(\{1\}) = 0, u_2 \geq v(\{2\}) = 0, u_3 \geq v(\{3\}) = 0$

(2) $u_1 + u_2 \geq v(\{1,2\}) = 3, u_1 + u_3 \geq v(\{1,3\}) = 3, u_2 + u_3 \geq v(\{2,3\}) = 3$

(3) $u_1 + u_2 + u_3 \geq v(\{1,2,3\}) = 4$.

Since the full coalition is also the maximal valued configuration, condition (3) is satisfied by equality (i.e., $u_1 + u_2 + u_3 = 4$). If we add up all the inequalities, we will have $u_1 + u_2 + u_3 >= 4\frac{1}{2}$, which cannot be satisfied. This means that in any division of the value of the full coalition among the agents there will be at least two agents that will prefer to opt out of the coalition and form a sub-coalition! For example, assume that the full coalition is formed with payoff vector $(1, 1, 2)$. Agents 1 and 2 can get more by forming a coalition (i.e., by excluding agent 3 from the coalition). The new payoff vector can then be $(1\frac{1}{2}, 1\frac{1}{2}, 0)$. This coalition and payoff vector is also not stable, since now agent 3 can tempt agent 2 (for example) to form a coalition with 3 by promising 2 more utility. The new payoff vector can then be $(0, 2, 1)$. However, now agent 1 can convince the two agents that they all can do better by forming the full coalition again. The new payoff vector can then be $(\frac{1}{3}, 2\frac{1}{3}, 1\frac{1}{3})$. This coalition is also not stable...

Shapley Value

The Shapley Value [0; 0] for agent i is a weighted average of all the utilities that i contributes to all possible coalitions. The weight of each coalition is the probability that this coalition will be formed in a random process that starts with the one-agent coalition, and in which this coalition grows by one agent at a time such that each agent that joins the coalition is credited with his contribution to the coalition. The Shapley Value is actually the expected utility that each agent will have from such a random process (assuming any coalition and permutation is equally likely).

Definition 6 *Given a superadditive coalition game with transferable utility in normal characteristic form (N, v), the Shapley Value is defined to be:* $u_i = \sum_{S \subset N, i \notin S} \frac{(n-|S|-1)!|S|!}{n!} v(S \cup \{i\}) - v(S)$.

The Shapley Value satisfies the efficiency, symmetry, and individual rationality conditions [0; 0]. However, it does not necessarily satisfy the coalition rationality condition. The Shapley Value divides the utility of the full coalition among the agents according to the average contributions of agents to all possible coalitions. Agents that contribute more (on the average) get a bigger share of the group utility. Agents that do not contribute at all get no utility. Agents that contribute in the same way (symmetric agents) get the same utility.

Definition 7 *We define the additional* value *that agent i adds to coalition S as $\Delta_v^i(S) \equiv v(S \cup \{i\}) - v(S)$.*

The Shapley Value can then also be defined as:

$$u_i = \sum_{S \subset N, i \notin S} \frac{(n - |S| - 1)!|S|!}{n!} \Delta_v^i(S).$$

Definition 8 *For a given encounter $(T_1, T_2, \ldots, T_n)$ in a TOD $< \mathcal{T}, \mathcal{A}, c >$ we will define $\forall S \subseteq N, c(S) = c(\bigcup_{i \in S} T_i)$.*

According to the above definition, we can see that: $\forall S \subseteq N,\ v(S) = \sum_{i \in S} c(T_i) - c(S)$.

$$\Delta_v^i(S) = v(S \cup \{i\}) - v(S) = c(T_i) + \sum_{j \in S} c(T_j) - c(T_i \cup \bigcup_{j \in S} T_j) - (\sum_{j \in S} c(T_j) - c(\bigcup_{j \in S} T_j)).$$

$$\Delta_v^i(S) = c(T_i) - (c(T_i \cup \bigcup_{j \in S} T_j) - c(\bigcup_{j \in S} T_j)).$$

Definition 9 *We define the additional* cost *that agent i adds to a coalition S as $\Delta_c^i(S) \equiv c(S \cup \{i\}) - c(S)$.*

If we use this definition, we can see that: $\Delta_v^i(S) = (c(T_i) - \Delta_c^i(S))$. Therefore, the Shapley Value can be defined now by:

$$u_i = \sum_{S \subset N, i \notin S} \frac{(n - |S| - 1)!|S|!}{n!} \Delta_v^i(S) =$$

$$\sum_{S \subset N, i \notin S} \frac{(n - |S| - 1)!|S|!}{n!} c(T_i) - \Delta_c^i(S)$$

[3] Agent 1 has to deliver letters to addresses a and d, agent 2 has to deliver letters to addresses b and e, and agent 3 has to deliver letters to addresses c and f.

Theorem 2 *The following holds:*

$$\sum_{S\subset N, i\notin S} \frac{(n-|S|-1)!|S|!}{n!} = 1$$

Since

$$u_i = \sum_{S\subset N, i\notin S} \frac{(n-|S|-1)!|S|!}{n!} c(T_i) - \sum_{S\subset N, i\notin S} \frac{(n-|S|-1)!|S|!}{n!} \Delta_c^i(S),$$

then from Theorem 2 we can conclude that:

$$u_i = c(T_i) - \sum_{S\subset N, i\notin S} \frac{(n-|S|-1)!|S|!}{n!} \Delta_c^i(S).$$

Theorem 3 *The Shapley Value is also:* $u_i = c(T_i) - \sum_{S\subset N, i\notin S} \frac{(n-|S|-1)!|S|!}{n!} \Delta_c^i(S)$. $\Delta_c^i(S) \equiv c(S \cup \{i\}) - c(S)$, *i.e., the additional* cost *that agent* i *adds to a coalition* S.

Agent i's Shapley Value is the difference between the cost of its goal and its weighted average cost contribution to all possible coalitions. The cost that agent i can contribute to a coalition, is bounded by $c(T_i)$. Therefore, the average contribution is also bounded by $c(T_i)$, which also means that the Shapley Value is positive (i.e., satisfies the individual rationality contribution) and bounded by $c(T_i)$ (which is also the maximal utility that an agent can get according to our model). Thus (as we promised above in Section), our model never attempts to transfer to an agent more utility than he can get by simply having his tasks performed by others. Even though our "transferable utility" is limited by the agent's stand-alone cost (since no actual money is being transferred), this poses no problem in our model.

Mechanisms for Subadditive TODs

We can define a Shapley Value-based mechanism for subadditive TODs that forms the full coalition and divides the value of the full coalition using the Shapley Value. The mechanism simply chooses the following (all-or-nothing) mixed deal, $(p_1, p_2, \ldots, p_n)$, such that $p_i = \frac{\sum_{S\subset N, i\notin S} \frac{(n-|S|-1)!|S|!}{n!} \Delta_c^i(S)}{c(N)}$.

To show that this all-or-nothing deal is well-defined, we need to show that $\forall i \in N: 0 \le p_i \le 1; \sum_{i=1}^n p_i = 1$.

p_i is the ratio between the weighted average cost contribution and the cost of the union of all tasks. The weighted average cost contribution is always positive, and it is bounded by $c(T_i)$, which (due to the monotonicity of c) is less than or equal to $c(N)$. Therefore, $0 \le p_i \le 1$. The Shapley Value is $u_i = c(T_i) - \sum_{S\subset N, i\notin S} \frac{(n-|S|-1)!|S|!}{n!} \Delta_c^i(S)$. Therefore, $p_i = \frac{c(T_i)-u_i}{c(N)}$.

The sum of the probabilities is therefore

$$\sum_{i=1}^n p_i = \sum_{i=1}^n \frac{c(T_i)-u_i}{c(N)} = \frac{1}{c(N)}(\sum_{i=1}^n c(T_i) - \sum_{i=1}^n u_i).$$

Since we know that the Shapley Value satisfies the efficiency condition, i.e., $\sum_{i=1}^n u_i = v(N)$, we can conclude that

$$\sum_{i=1}^n p_i = \frac{1}{c(N)}(\sum_{i=1}^n c(T_i) - v(N)).$$

¿From the definition of $v(N)$ we can additionally conclude that

$$\sum_{i=1}^n p_i = \frac{1}{c(N)}(\sum_{i=1}^n c(T_i) - (\sum_{i=1}^n c(T_i) - c(N))) = \frac{c(N)}{c(N)} = 1.$$

Thus, executing this all-or-nothing deal will give each agent i an expected utility of $c(T_i) - p_i c(N)$, which is exactly the Shapley Value u_i.

Theorem 4 *The above all-or-nothing deal is well-defined, (i.e.,* $\forall i \in N: 0 \le p_i \le 1; \sum_{i=1}^n p_i = 1$.*) and gives each agent* i *an expected utility which is exactly the Shapley Value* u_i.

Evaluation of the Mechanism

The above mechanism gives each agent its Shapley Value. The mechanism is thus symmetric and efficient (i.e., satisfying group rationality), and also satisfies the criterion of individual rationality. However, as was seen in Example 2, no mechanism can guarantee coalition rationality. Besides failing to guarantee coalition rationality, the mechanism also does not satisfy the simplicity condition. It requires agents to calculate the Shapley Value, a computation which has exponential computational complexity.

The computational complexity of a mechanism should be measured relative to the complexity of the agent's standalone planning problem. This relative measurement would then signify the computational overhead of the mechanism. Each agent in a Task Oriented Domain needs to calculate the cost of his set of tasks, i.e., to find the best plan to achieve them. Calculation of the value of a coalition is linear in the number of agents in the coalition[4]. The calculation of the Shapley Value requires an evaluation of the value of all (2^n) possible coalitions. In Section below we will show that there exists another Shapley-based mechanism that has linear computational complexity.

Concave TODs

We here review definitions of concave Task Oriented Domains, discussed by Zlotkin and Rosenschein in [0].

[4]The cost of set of task need to be calculated only linear number of times.

Definition 10 [Concavity]: *TOD* $< \mathcal{T}, \mathcal{A}, c >$ *will be called* **concave** *if for all finite sets of tasks* $X \subseteq Y$, $Z \subseteq \mathcal{T}$, *we have* $c(Y \cup Z) - c(Y) \leq c(X \cup Z) - c(X)$.

In other words, the cost that arbitrary set of tasks Z adds to set of tasks Y cannot be greater than the cost Z would add to a subset of Y.

Theorem 5 *All concave TODs are also subadditive.*

All concave TODs are also subadditive. It turns out that general subadditive Task Oriented Domains can be restricted, becoming concave Task Oriented Domains. For example, the Postmen Domain is subadditive, when the graphs over which agents travel can assume any topology. By restricting legal topologies to trees, the Postmen Domain become concave.

Definition 11 *A* coalition *game with transferable utility in normal characteristic form* (N, v) *is* convex *if for any coalitions* S, V, $v(S) + v(V) \leq v(S \cup V) + v(S \cap V)$.

In convex coalition games, the incentive for an agent to join a coalition grows as the coalition grows. Just as there is a relationship between subadditive TODs and superadditive coalition games, so there is a relationship between concave TODs and convex coalition games.

Theorem 6 *Any encounter* $(T_1, T_2, \ldots, T_n)$ *in a concave TOD induces a* convex *coalition game* (N, v).

Theorem 7 [Shapley (1971)]: *In convex coalition games, the Shapley Value always satisfies the criterion of coalition rationality.*

In concave TODs, the Shapley-based mechanism introduced above is fully stable, i.e., satisfies individual, group and coalition rationality.

The Random Permutation Mechanism

The Shapley Value is equal to the expected contribution of an agent to the full coalition, assuming that all possible orders of agents joining and forming the full coalition are equally likely.This leads us to a much simpler mechanism called the *Random Permutation Mechanism*: Agents choose a random permutationand form the full coalition, one agent after another, according to the chosen permutation. Each agent (i) gets utility (w_i) that equal to its contribution to the coalition, at the time he joined it. This is done by agreeing on the all-or-nothing deal, $(p_1, p_2, \ldots, p_n)$, such that: $p_i = \frac{c(T_i) - w_i}{c(N)}$.

Theorem 8 *If each permutation has an equal chance to be chosen, then the Random Permutation Mechanism gives each agent an expected utility that is equal to its Shapley Value.*

The Shapley-based Random Permutation Mechanism does not explicitly calculate the Shapley Value, but instead calculates the cost of only n sets of tasks. Therefore, it has linear computational complexity. The problem of coalition formation is reduced to the problem of reaching consensus on a random permutation.

Consensus on Permutation

No agent would like to be the first one that starts the formation of the full coalition (since this agent by definition gets zero utility). If the domain is concave (and therefore the coalition game is convex), each agent has an incentive to join the coalition as late as possible. To ensure stability, we need to find a consensus mechanism that is resistant to any coalition manipulation. No coalition should be able, by coordination, to influence the resulting permutation such that the members of the coalition will be the last ones to join the full coalition. For example, this means that no coalition of $n - 1$ agents could force the single agent that is out of the coalition to go first.

We will use the simple cryptographic mechanism that allows an agent to encrypt a message using a private key, to send the encrypted message, and then to send the key such that the message can be unencrypted.

Using these tools, each agent chooses a random permutation and a key, encrypts the permutation using the key, and broadcasts the encrypted message to all other agents. After he has received all encrypted messages, the agent broadcasts the key. Each agent unencrypts all messages using the associated keys. The consensus permutation is the combination of all permutations.

Each agent can make sure that each permutation has an equal chance to be chosen even if he assumes that the rest of the agents are all coordinating their permutations against him (i.e., trying to make him be the first). All he needs to do is to choose a *random* permutation. Since his permutation will also be combined into the final permutation, everything will be shuffled in a way that no one can predict.

The Random Permutation Mechanism has only linear computational complexity and uses a linear number of broadcasts. When it is implemented using cryptographic tools, it is a non-manipulable Shapley-based mechanism, and therefore symmetric, efficient, and individual rational. In concave TODs, it is also coalition rational.

Conclusions

We have considered the kinds of n-agent coordination mechanisms that can be used in Task Oriented Domains (TODs), when any sub-group of agents may engage in task exchange to the exclusion of others. We concentrated attention on subadditive TODs, and showed that the full coalition is the most efficient coalition. We discussed attributes that might be desirable in a mechanism that divides group utility among its members.

We presented a simple Shapley Value-based coalition formation mechanism that uses cryptographic techniques for subadditive TODs. The mechanism is efficient, symmetric, and individual rational. When the domain is also concave, the mechanism also satisfies coalition rationality.

Future research will consider non-subadditive TODs. It will also consider issues of incentive compatibility in multi-agent coalition formation, investigating mechanisms that can be employed when agents have partial information about the goals of their group members and can deceive one another about this private information.

References

S. Conry, R. Meyer, and V. Lesser. Multistage negotiation in distributed planning. In A. Bond and L. Gasser, editors, *Readings in Distributed Artificial Intelligence*, pages 367–384. Morgan Kaufmann Publishers, Inc., San Mateo, 1988.

E. Ephrati and J. S. Rosenschein. The Clarke Tax as a consensus mechanism among automated agents. In *Proceedings of the Ninth National Conference on Artificial Intelligence*, Anaheim, California, July 1991.

E. Ephrati and J. S. Rosenschein. Reaching agreement through partial revelation of preferences. In *Proceedings of the Tenth European Conference on Artificial Intelligence*, pages 229–233, Vienna, Austria, August 1992.

E. Ephrati and J. S. Rosenschein. Distributed consensus mechanisms for self-interested heterogeneous agents. In *First International Conference on Intelligent and Cooperative Information Systems*, pages 71–79, Rotterdam, May 1993.

James P. Kahan and Amnon Rapoport. *Theories of Coalition Formation*. Lawrence Erlbaum Associates, London, 1984.

Steven Paul Ketchpel. Coalition formation among autonomous agents. In *Pre-Proceedings of the Fifth European Workshop on Modeling Autonomous Agents in a Multi-Agent World*, Neuchatel, Switzerland, August 1993.

S. Kraus and J. Wilkenfeld. Negotiations over time in a multi agent environment: Preliminary report. In *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence*, pages 56–61, Sydney, August 1991.

S. Kraus, J. Wilkenfeld, and G. Zlotkin. Multiagent negotiation under time constraints. Computer Science Technical Report Series CS-TR-2975, University of Maryland, College Park, Maryland, October 1992.

Thomas Kreifelts and Frank von Martial. A negotiation framework for autonomous agents. In *Proceedings of the Second European Workshop on Modeling Autonomous Agents and Multi-Agent Worlds*, pages 169–182, Saint-Quentin en Yvelines, France, August 1990.

Kazuhiro Kuwabara and Victor R. Lesser. Extended protocol for multistage negotiation. In *Proceedings of the Ninth Workshop on Distributed Artificial Intelligence*, pages 129–161, Rosario, Washington, September 1989.

Martha E. Pollack and Marc Ringuette. Introducing the Tileworld: Experimentally evaluating agent architectures. In *Proceedings of the National Conference on Artificial Intelligence*, pages 183–189, Boston, Massachusetts, August 1990.

Jeffrey S. Rosenschein. Consenting agents: Negotiation mechanisms for multi-agent systems. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 792–799, Chambery, France, August 1993.

Lloyd S. Shapley. A value for n-Person games. In Alvin E. Roth, editor, *The Shapley Value*, chapter 2, pages 31–40. Cambridge University Press, Cambridge, 1988.

On Shechory and Sarit Kraus. Coalition formation among autonomous agents: Strategies and complexity. In *Pre-Proceedings of the Fifth European Workshop on Modeling Autonomous Agents in a Multi-Agent World*, Neuchatel, Switzerland, August 1993.

Reid G. Smith. *A Framework for Problem Solving in a Distributed Processing Environment.* PhD thesis, Stanford University, 1978.

K. P. Sycara. Resolving goal conflicts via negotiation. In *Proceedings of the Seventh National Conference on Artificial Intelligence*, pages 245–250, St. Paul, Minnesota, August 1988.

H. P. Young. Individual contribution and just compensation. In Alvin E. Roth, editor, *The Shapley Value*, chapter 17, pages 267–278. Cambridge University Press, Cambridge, 1988.

G. Zlotkin and J. S. Rosenschein. Negotiation and task sharing among autonomous agents in cooperative domains. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, pages 912–917, Detroit, Michigan, August 1989.

G. Zlotkin and J. S. Rosenschein. Cooperation and conflict resolution via negotiation among autonomous agents in noncooperative domains. *IEEE Transactions on Systems, Man, and Cybernetics*, 21(6):1317–1324, December 1991.

G. Zlotkin and J. S. Rosenschein. A domain theory for task oriented negotiation. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 416–422, Chambery, France, August 1993.

G. Zlotkin and J. S. Rosenschein. Negotiation with incomplete information about worth: Strict versus tolerant mechanisms. In *Proceedings of the International Conference on Intelligent and Cooperative Information Systems*, pages 175–184, Rotterdam, May 1993.