A Game Theoretic Approach to the Pursuit Problem

Ran Levy
Jeffrey S. Rosenschein
Computer Science Department, Hebrew University
Givat Ram, Jerusalem, Israel
phone:(+972)-2-585-353
fax:(+972)-2-585-439
ranlevy@shum.huji.ac.il, jeff@cs.huji.ac.il

Abstract

Empty abstract.

To appear in "Decentralized Artificial Intelligence III," Y. Demazeau and E. Werner, editors, Elsevier Science Publishers B.V./North-Holland, 1992.

A GAME THEORETIC APPROACH TO DISTRIBUTED ARTIFICIAL INTELLIGENCE AND THE PURSUIT PROBLEM

Ran Levy and Jeffrey S. Rosenschein

Computer Science Department
Hebrew University
Givat Ram
91904 Jerusalem
ISRAEL
email: ranlevy@shum.huji.ac.il, jeff@cs.huji.ac.il

Research in Distributed Artificial Intelligence (DAI) often considers the problem of how best to utilize multiple automated agents to accomplish a given task. A canonical problem, the Pursuit Problem [1], has been suggested as a useful tool for evaluating alternative approaches to the distribution of knowledge and control among intelligent, cooperative problem-solvers. Work on the Pursuit Problem has been carried out by several researchers; for example, Stephens and Merx [16, 17] compare alternative approaches to solving the problem.
In [5], a game theoretic model was proposed for DAI in which each agent works for his own selfish goals. In this paper, we suggest a method for solving the Pursuit Problem using game theoretic techniques, by incorporating the global goal of a group of agents into their local interests. Although applied to the Pursuit Problem, the technique has wider applicability throughout DAI. We present the results of a simulation of the game theory model for the Pursuit Problem, and compare results to those of Stephens and Merx's models.

1 Introduction

Research in Distributed Artificial Intelligence (DAI) often considers the problem of how best to utilize multiple automated agents to accomplish a given task. Work in DAI has made use of several approaches to the distribution of knowledge and control among intelligent, cooperative problem-solvers. The contract-net model of Davis and Smith [3] uses bidding and contracting to achieve cooperation in the assignment of tasks to processors in a multiprocessor system. Malone *et al.* [9] refined the above framework by introducing a more explicit economic model to cooperation.

Another direction of research is presented in [1]. It is assumed that cooperation among agents is a function of their organizational structure and the way they communicate with one another. Benda *et al.* examined several organizational structures, consisting of three basic components (communicating agents, negotiating agents, and a controlling agent) in connection with a pursuit problem, first presented in that research.

The Pursuit Problem as formulated by Benda models a configuration of four intelligent agents, together performing a given task that requires some sort of coordination (explicit or implicit). Four blue agents, positioned at different locations on a grid, are attempting to surround a fifth red agent. See Figure 1. The red agent moves randomly in any possible direction (i.e., one not blocked by a blue agent). In addition to the global goal of the system, each agent may have some local goal of its own. Those local needs may contradict both the common goal and local goals of other agents. Some sort of negotiation (explicit or implicit) might be used to achieve a compromise. Benda *et al.* concluded that an organization with one controlling agent and three communicating agents was best for solving the problem.

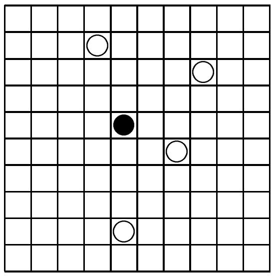

Figure 1: The Pursuit Problem

Gasser and Rouquette [4] developed a framework for representing organizational knowledge, using the pursuit game, to investigate decentralized coordination mechanisms. Their analysis introduced a six-phase solution, where each phase involved a different organization of agents. Singh [15] applied his theory of "group intentions" to the pursuit problem to demonstrate that theory's general utility to DAI.

Conclusions similar to those of Benda are suggested by Stephens and Merx [16, 17]. In [17] they compare the results of using different control strategies for solving the Pursuit Problem (local control, distributed control, and central control), and show that the central control strategy is superior to the others. It guarantees capture of the red agent in all 30 experiments, with best capture efficiency and success ratio.

Rosenschein *et al.* [13, 5, 19] analyze various rationality assumptions on multi-agent system behavior, along with their implications. These results (such as the existence of dominance analysis, and iterated dominance analysis) suggest that Game Theory techniques, along with appropriate rationality assumptions, might allow greater flexibility in the autonomous agents model of coordination. One of the goals of this work is to suggest a method for incorporating global goals into the local interests of all agents through the use of Game Theory techniques. Specifically, we are interested in considering how the Pursuit Problem might be solved by game theoretic agents.

2 Necessary Concepts from Game Theory

In order to clarify our discussion in the following sections, we here give a brief overview of concepts, definitions and theorems of Game Theory. The material covered here is mainly due to Von Neumann and Morgenstern [11], Nash [10], and Shapley [14].

2.1 Non-Cooperative Games

Let N be a set of n players, $N = \{1, \dots, n\}$. Let S^i be a finite set of strategies for player i, $i \in N$. We will denote by S the cartesian product $S^1 \times \cdots \times S^n$.

Definition 1 *A* Finite Non-Cooperative Game in Normal Form *is a system*

$$G = (S^1, \dots, S^n; H^1, \dots, H^n)$$

where S^i is a non-empty set of strategies of player i, and H^i is a payoff function, $H^i : S \to R$ for player i, $i \in N$.

One of the interesting questions about non-cooperative games is whether a player can determine a specific strategy that puts a maximal lower bound on his income. In order to determine this strategy, a player should reason as follows: the strategy he chooses should be the one that, for every coordinated strategy of all other agents, provides him with the best payoff he can get. Now suppose that all other agents choose *their* best strategies. Possibly, depending on the game's payoffs, these strategies are the worst for our player (i.e., what the other players are trying to do is minimize the maximum amount our player can get). However, the strategy that our player chose guarantees him at least some pre-determined income.

Keeping in mind that the same argument holds for *all* players, we now turn our attention to the concept of equilibrium points, which addresses the above question. Definition 2 states the following about an equilibrium point: for each player, as long as the other players maintain their same strategies, his current strategy is the best he's got.

Definition 2 *An n-tuple of strategies $s \in S$ is a* Nash Equilibrium Point *if for each player i, $i \in N$, and for each strategy $\widehat{s^i}$ of player i,*

$$H^i(s|\widehat{s^i}) \leq H^i(s)$$

where the strategy $s|\widehat{s^i}$ is derived from s by substituting $\widehat{s^i}$ for s^i.

In pure strategies, such equilibrium points do not always exist. Even when they do exist, they are not necessarily ideal solutions for games (e.g., there may be other outcomes that are better for some players and no worse for anyone). The well-known Prisoners' Dilemma is an example of this; if the agents played a specific different, non-equilibrium strategy, they would both get higher utility than when they play an equilibrium strategy.

2.2 Cooperative Games

Games with side-payments are games in which the participants get immediate and transferable utility (that is, money). In a cooperative game (with side payments), some players may form a coalition and together design their strategies so that they can increase their total income. This total income then has to be shared—in some manner—among all the coalition's members. The theory of cooperative games (with side-payments) assumes that a coalition achieves some payment, and deals with the different methods for sharing the profit.

Definition 3 *We define a* Cooperative Game in Coalitional Form with side-payments *as a pair (N, v), where N is the set of players, and $v : 2^N \to R$ is a payoff function, where $v(T)$ is the payment to coalition T, and $v(\emptyset) = 0$.*

A cooperative game is therefore stated in terms of the *coalitional function* v. Note that no strategies appear in the definition. Once a coalition is formed, it is of no concern how it gains its utility. A player should take into account the coalitions he may participate in and their incomes, and decide which coalition is the best for him. We will be using the Shapley Value [14] in order to determine the way in which the income is divided; this technique provides us with a single payoff vector specifying the income distribution.

Definition 4 *A* Payoff Vector *for N is a function $x : N \to R$.*

The i-th coordinate in the payoff vector x corresponds to the payment for player i. Let G^N be the set of all cooperative games over N.

Definition 5 *A* solution *to cooperative games is a function $\psi : G^N \to R^N$.*

The function ψ represents a method for sharing $v(N)$ among all players in N.

The Shapley Value satisfies three properties (see [18]). The *symmetry* property guarantees that there is no preference by the solution of one player over another player. What is important to the solution concept is not the identity of a player, but the role it plays. *Efficiency* guarantees that players cannot get (according to the solution) more than N can afford. Finally, the *marginality principle* states that a player's "value" to the coalition is measured by his contribution to the coalition.

The Shapley Value may be calculated using the following formula:

$$\psi^i(v) = \sum_{T \subset N, i \notin T} \frac{(n - |T| - 1)!\ |T|!}{n!} [v(T \cup \{i\}) - v(T)]$$

The Shapley Value thus is computed by considering a player's contribution to *all* possible other coalitions (that do not originally include himself), and averaging these contributions. For example, in a four-player model, we would compute player 3's Shapley Value by considering his contribution to the coalition $\{1, 2, 4\}$, to the coalition $\{1, 2\}$, to the coalition $\{4\}$, etc., and averaging over them all.

3 A Model for the Pursuit Problem

From our game theoretic point of view, the Pursuit Problem has two aspects to be considered. First of all, the players should be given methods for solving the game, so that they have the ability to cope with problems they face. In addition, we must also set the payment policy according to which the players are being paid for their actions. The idea that local interests of the players should not interfere with the global goal of the group of players should be incorporated into the payment policy of the game.

Our assumption is that the local goals of agents are represented (as in a free market) by the desire to earn money. If the players receive money for participating in a coalition, that money motivates them, but each player must also take into account any other payments he might receive or give as a result of his participation (e.g., other local considerations). Thus, we are *integrating* a global goal into agents' local goals, not totally replacing their local goals with a global goal.[1] Another of our design aims for the system was to have the agents reach a solution with minimal message-passing, although not limiting the amount of "non-verbal" communication. For example, players are assumed to know where other players are located (e.g., visually).

3.1 Payment Policy

The payment policy we suggest incorporates two ideas. First, the convergence to the final state should be as close as possible to optimal (i.e., fast). Moreover, it should ultimately represent the global interest of the system (i.e., capture). Therefore, the payoff function is the sum of two distinct arguments.

For every move, a player is paid an amount of money proportional to the difference of his distance from the red agent before and after the move. Distance is measured using city block metrics. This amount of money may be positive or negative (or zero, if the player decides to remain in his position). This argument of the payoff function causes the agent to prefer moves that tend to bring the player closer to the red agent.

The second argument of the payoff function encourages agents to coordinate their actions and block the red agent. The red agent has four possible directions along which he can move. Each direction may be represented by a

[1] In the current paper, for simplicity, our agents have no other local goals.

vector in that direction, emanating from the location of the red agent. Thus, for each player i, and for each direction d, two magnitudes are well defined: d_a^i, which is the distance of player i from the red agent *along* direction d, and d_p^i, which is the distance of i from the red agent along a *perpendicular* line to direction d.

Definition 6 *An agent i is said to be* blocking as escape direction d *if both the following conditions are true:*

1. $d_p^i < d_a^i$
2. *the player is moving towards the direction opposite to d.*

This definition of "blocking" tends to encourage the blue agents to gravitate towards one of the diagonals emanating from the red agent. For example, for a blue agent to be blocking on the East, it must be actively moving West towards the red agent.

Given a coordinated strategy of a coalition $T \subseteq N$, we thus use the following arguments:

- The sum of differences in players' distance from the red agent (using the city-block metrics). We denote this sum by $\sum_{i \in T} d^i$.
- The number of blocked escape directions of the red agent, using the strategy of T. Let k_T designate this number.

Note that these calculations use the position of the red agent as it was after the previous move. In order to emphasize the importance of cooperation among players, it is suggested that the second argument of the payoff function will be the k_T's power of some number (unless k_T is zero, in which case the second argument will also be zero). There is, however, one point which must be taken into account: the payoff function should make a distinction between a cooperative and a non-cooperative coalition. This distinction is essential to characterizing equilibrium points, as discussed in Section 4.1. For example, consider the second argument to be 2^{k_T}. There is no way to distinguish between a coalition of two players who together block one escape-direction (the payoff to this coalition should be $2 + 2^1 = 4$), and four players moving separately, blocking no escape-direction (the non-cooperative "coalition" of four thus gets $4 + 0 = 4$). A similar argument holds for both 3^{k_T} and 4^{k_T}. As

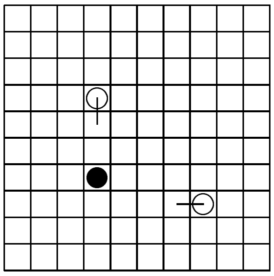

Figure 2: An Example With Two Players

shown in Section 4.1, the proof of the characterization lemma is based on a base of 5 for k_T. Hence it was chosen for the payoff function.

Example. Consider Figure 2, in which only two players appear, for simplicity. N, therefore, is the set $\{1, 2\}$, and:

$$v(\emptyset) = 0$$
$$v(\{1\}) = \sum_{i \in \{1\}} d^i + 5^{k_{\{1\}}} = 1 + 5 = 6$$
$$v(\{2\}) = \sum_{i \in \{2\}} d^i + 5^{k_{\{2\}}} = 1 + 5 = 6$$
$$v(\{1,2\}) = \sum_{i \in \{1,2\}} d^i + 5^{k_{\{1,2\}}} = 2 + 5^2 = 27$$

4 Suggested Solution

We derive a solution that mixes cooperative games with non-cooperative games. As each player has to make its own decision, without explicit coordination with the others, each player has to solve a four-player non-cooperative game with side payments. This game will be denoted by G, and what follows is a description of how it should be constructed.

The set of agents (or players) is:

$$N = \{1, 2, 3, 4\}$$

Each player $i \in N$ has a set of pure strategies, given by:

$$S^i = \{\text{South, West, North, East, Stay}\}$$

Let S be defined by:

$$S = \prod_{i \in N} S^i$$

We use $H^i(s)$ in order to specify the payoff to player i for a strategy $s \in S$. Using the payment policy described in the previous section, notice that each move $s \in S$ actually defines a four-player cooperative game with side-payments in the following manner: let $T \subseteq N$ be a coalition of players. Then

$$v_s(T) = \begin{cases} \sum_{i \in T} d^i + 5^{k_T} & \text{if } k_T \neq 0 \\ \sum_{i \in T} d^i & \text{if } k_T = 0 \end{cases}$$

and

$$v_s(\emptyset) = 0.$$

We refer to this cooperative game as the "local game" for that move. The amount of utility, represented by $v_s(N)$, will be shared among the four players in accordance with the Shapley Value. Player i (for $i \in N$) will get $\psi^i(v_s)$, where ψ represents the Shapley Value, and

$$\sum_{i \in N} \psi^i(v_s) = v_s(N).$$

Having solved the local game, a player should now return to the "global" non-cooperative game, and define the payoff function for player i ($i \in N$) for each move $s \in S$ by:

$$H^i(s) = \psi^i(v_s).$$

Hence, we complete the formalism of the non-cooperative game, and let G be:

$$G = (S^1, S^2, S^3, S^4; \quad H^1, H^2, H^3, H^4).$$

The game G may be constructed, analyzed and solved by each player on its own. Because of the desirable properties of equilibrium points, we design our agents so that they choose to play a strategy that corresponds to an equilibrium point in G. The actual algorithm is presented in Section 4.1.

4.1 Equilibrium Points

In this section we consider the relation of equilibrium points to what we shall call "optimal points." For our purposes, an optimal point represents moves that are best for the system as a whole. It turns out that the game G defined above has at least one equilibrium point, depending on the specific scenario.

Definition 7 *A coordinated strategy* $x \in S$ *is said to be an* optimal point *in* G *(defined above) if for all* $y \in S$ *the following condition holds:*

$$x(N) = \sum_{i \in N} H^i(x) \geq \sum_{i \in N} H^i(y) = y(N)$$

Optimal points represent system-wide optimization. At an optimal point, the coalition of all players N gets the maximum amount of utility it can get in the game G. Interestingly, although the concept of equilibrium points arises from local optimization by players, the nature of the game G (and especially the payoff function) enables us to prove the following theorems about system-wide optimality.

Theorem 1 *If* $x \in S$ *is an equilibrium point of* G*, then* x *is also an optimal point in* G*.*

The point here is that although ostensibly the players don't care about system-wide optimality, the payoff policy of the game makes them act in a way that guarantees it. For proof of this theorem and subsequent ones, see [8]. One important corollary of Theorem 1 enables us to improve the algorithm of Section 4:

Corollary 1 *The set of equilibrium points in* G *is a subset of the set of optimal points in* G*.*

Corollary 2 (Decision Algorithm) *One need not explicitly compute equilibrium points in* G*. Rather, one should compute* $v_s(N)$ *for each strategy* $s \in S$*, sort out the results to find optimal points, and only then check those optimal points to see if they are also equilibrium points.*

Thus, the complexity of checking for equilibrium points is greatly reduced. We need only confine our search to the optimal points, which are easily identified.

We now turn our attention to the opposite direction. We first state a degenerate version of the theorem, and then extend it to the general case.

Theorem 2 *Let* $x \in S$ *be a unique optimal point in* G*. Then* x *is an equilibrium point in* G*.*

Example. Consider the board in Figure 3. The following two coordinated strategies are equivalent with respect to the payoff to N:

$$(SOUTH, WEST, WEST, EAST)$$

$$(SOUTH, WEST, NORTH, EAST)$$

One can easily see that both strategies are optimal (and therefore N, the whole group of agents, is paid the same for both of them). With both strategies, the South direction is not blocked. However, player 3 contributes more to coalitions in which he participates if he uses the first coordinated strategy. To understand why this is true, consider that if player 3 moves West, he "double-blocks" East, that is, both he *and* player 2 are blocking East (but of course, the coalition of four doesn't benefit from the double-blocking). If, on the other hand, player 3 moves North, he is not blocking any direction by our definition. However, the four-player coalition isn't harmed, since East is already blocked by player 2.

According to the Shapley Value, however, player 3 should get more money for using the first strategy (whereas other players in N will get less money, to keep the sum equal). This is true because, by blocking East, player 3 will be seen as contributing more to coalitions that involve only players 1 and 4 (without player 2), and will thus raise his Shapley Value, at the expense of the other agents. Hence the second strategy is not an equilibrium point, though it is optimal.[2]

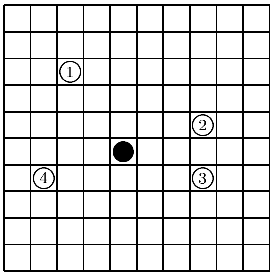

Figure 3: Example of an Optimal but not Equilibrium Point

The example above demonstrates that if there are several optimal points in G, then some of them may not be equilibrium points in G. We therefore extend the above theorem and state the following one.

[2]Note that the other three agents will not benefit by changing their moves even when player 3 chooses West rather than North.

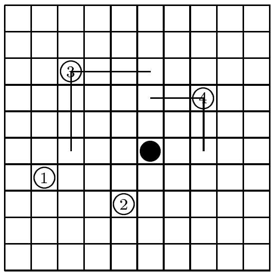

Figure 4: A Situation With Two Equilibrium Points

Theorem 3 *Let $O \subset S$ be a set of optimal points in G. Then there is a point $x \in O$ that is also an equilibrium point in G.*

A straightforward corollary of Theorem 3 is the following:

Corollary 3 G does have *an equilibrium point, since surely it has at least one optimal point.*

There are situations in which there exist more than one equilibrium point. Such a situation occurs in the following example.

Example.

Consider Figure 4. Players 3 and 4 have two symmetric directions in which they can move (regarding distance from the red agent, and blocking escape-directions). Therefore, every two strategies of the form

$$(s^1, s^2, EAST, SOUTH)$$

$$(s^1, s^2, SOUTH, WEST)$$

will be equal in worth to each other.

The payment to the group of all players, N, is the same for all optimal points (this follows immediately from the definition of optimal points). Can there be different situations in which N receives the same payment, and what characterizes such situations?

Lemma 1 *If at two points in G, N gets the same payment, then:*

1. *The number of blocked escape directions in the two points is the same.*
2. *The total difference in the distance of players from the red agent is the same in the two points.*

Figure 5: Typical Problems, Cases 1 and 2

For the proof, see [8]. The proof uses the constant 5 as the base of the second argument in the payoff function as a result of the fact that there are 4 blue agents.

4.2 Scope and Limitations of the Algorithm

One of the main goals of our system design was to minimize the need for explicit communication among agents. From our experiments we concluded that in our model, a general solution without any communication is impossible (see discussion in Section 6).[3]

Consider the left side of Figure 5 in which the two blue agents are at distance 3 from each other. Agent 1 will surely select $SOUTH$ as his next move. Agent 2 may choose $SOUTH$ or $WEST$ (ignore for the moment agents 3 and 4, and the exact equilibrium points). If agents 1 and 2 would not coordinate themselves via a negotiation process, agent 2 may move to the $WEST$ and block agent 1, who cannot recover from this afterwards. In the right side of Figure 5 (distance 2 between agents) coordination is needed to prevent a situation where both agents move to the same location (agent 1 moves to the $SOUTH$, agent 2 to the $WEST$).

Experimental results supported the hypothesis that if the distance between agents is 3 or less (using city block metrics), explicit negotiation is needed to synchronize them.

[3]Obviously, another model, even one that continues to use Game Theory, might have different properties, and may well have different limitations as well.

5 Experimental Results

A computer program to implement our model was written in Common Lisp. Two questions were to be tested and answered by the simulation:

- What are the initial states of the problem for which the model does provide suitable solutions?
- How does the existence of more than a single equilibrium point affect the need for communication?

The question of "convergent" initial states, i.e., those states for which we know the solution to converge, depends on the definition of the payoff function. Therefore, the following discussion is closely tied to our particular payoff function. Once a player identifies a *single* free escape direction, he does not give it up. If two (or more) players compete over that direction, the solution will probably not converge (we say "probably" because random movements of the red agent may change one situation into another one, in which the solution may converge). In other words, if the initial state is such that resources are *available* to players, and all they have to do is decide on the best method to share them, then success is guaranteed (subject to random behavior of the red agent). In order to cope with situations in which resources are not directly available, one should correct the payoff function in an appropriate way.

In order to answer the question of how multiple equilibrium points affect the need for communication, let us turn to an example (the middle of the game is seen on the left side of Figure 6). The original problem was solved in 11 moves, but from step 8 on, interference of an outside user (who simulates an inter-agent discussion to resolve symmetrical equilibrium points) is necessary. The answer of the outside user on step 3 and 4 is actually an arbitrary selection. However, from step 8 in test-case I and step 7 in test-case II (see Figure 6), a bit of thought must be given in order to prevent the game from reaching a dead end. This is necessary because in those steps, blue agents 1 and 2 get too close to each other. When agents sense that they are very close, they should coordinate via communication, so that they can prevent themselves from competing over the same resource.

In Section 4.1 we discussed the symmetry of multiple equilibrium points, and showed that this symmetry indicates the existence of equivalent actions by one or more players. Test cases I and II show that as long as the blue

Figure 6: Step 8 of Case I and Step 7 of Case II

agents who have symmetric actions are far from each other and from the red agent, there is no importance to the equilibrium point that is selected. However, as the blue agents get closer to the red agent (up to a distance of 3 or less from the red agent), the significance of intra-agent communication grows, and sometimes becomes necessary. These phenomena may also be observed from other test cases. See [8] for full experimental test results.

One problem that needed to be addressed in the simulation was that of calculating equilibrium points in non-cooperative games.[4] This subject is discussed in the literature (see, for example, the work of Chin *et al.* [2], Harsanyi [6] and Rosenmüller [12]). There are several mathematical algorithms to find and calculate equilibrium points in such games. In the actual implementation, however, we used a simple exhaustive search over all points in the non-cooperative game in order to find equilibrium points. In a more realistic system, simplicity would be abandoned in favor of efficiency.

6 Comparison and Conclusions

In [16], Stephens and Merx summarize the performance results of four methods of solution on six different scenarios. In [17], they compare three different control strategies, experimenting with them over thirty test cases. In both works, they specify three possible outcomes, *capture*, *stalemate*, and *escape*.

At this point in our research, we have used the six scenarios in [16] as test cases for our simulation.[5] The results are summarized in Appendix B.

[4]In this specific model, the agents could have used the shortcut of optimal points, but the simulation had them calculate, more generally, equilibrium points.

[5]Future work will evaluate the performance of our model on the thirty test cases used in [17].

In three of the cases the game ended in the capture of the red agent, and in two others in stalemate. In one case (scenario 1 of [16]) the simulation did not even get close to the solution, with all the agents clustered far away from the red agent and satisfied to remain there. The payoff function was not designed for that kind of situation and failed to motivate the blue agents to capture the red agent. Table 2 extends Table 4 in [16] to include our results.

Interestingly, our model fails in scenario 6 where *all* other models succeed. This result demonstrates the nature of the underlying model. In the first few stages of the game, the agents appear to be proceeding correctly. However, at a particular point near the end of the game, the agents continue in their rush towards the original capture positions, even though the red agent has meanwhile scurried away. In Section 5 we pointed out that communication should be used to select one of several equilibrium points. This example shows that there are cases in which communication might also be used to prevent the system agents from taking a misguided path.

Although there are random elements involved (i.e., the movement of the red agent), we can still reach some preliminary conclusions regarding the effectiveness of our model. In those cases for whom our payoff function is known to be appropriate, the results show convergence to the solution, sometimes with a shorter number of moves than in other methods. However, the definition of the payoff function has the disadvantage of not covering all cases, as seen by cases 1, 5 and 6. To its credit, however, agents in this model do not need to explicitly communicate with one another most of the time (see below). Explicit communication is needed when agents get close to other agents.

Two important aspects of Stephens and Merx's work, as related to our own, need to be pointed out. First of all, both control and pursuit strategies are varied in their work at the same time, so it is difficult to isolate the influence of control strategy over performance. Second, the agents in Stephens and Merx's models cannot see one another; our model requires that each agent, at all times, has complete knowledge of all agent locations. One may see this information updating as another kind of communication, e.g., visual, implicit communication among agents.

See Appendix A for the application of these techniques to the real-world problem presented originally in [7].

Acknowledgements

We would like to thank Ed Durfee for his constructive comments on an earlier draft of this paper.

A Application to Collision Avoidance at Sea

The discussion in this article was very much oriented towards the Pursuit Problem. In order to give a less narrow perspective on the subject, this section gives an analysis of a somewhat more real-world problem, presented by Jin and Koyama in [7]. We intend to give here only an illustration of a more practical use of the techniques presented above.

We open the discussion by quoting Jin and Koyama as they introduced the collision avoidance problem in [7]:

> "The ship collision avoidance problem can be briefly described as follows. While navigating, a ship should keep a certain range around her clear to prevent collisions. The specification of the range is related to the size, speed and maneuverability of the ship and environmental conditions at the time. The radius of the range is called *safety distance* which is usually about two to four times the ship length."

When two ships enter into the range of observation of one another, they must calculate their paths so as to avoid collision. A complete analysis and explanation can be found in [7].

This problem (like the Pursuit Problem) involves multiple agents, sharing a common task—avoiding collision. In addition, each agent might also have private interests. For example, the interests of a ship may be listed as follows:

1. Avoid Collision, since you'll sink if you collide.

2. Avoid delays in your schedule.
Maneuvering a large ship is slower than maneuvering a small ship. As a result, the time spent when a large ship maneuvers is longer than the time spent when a small ship maneuvers. So if you're a large ship—avoid maneuvers as much as you can.

3. Lower fuel consumption.
Your fuel consumption is directly related to your path. Consume as little fuel as you can (obviously, maneuvers influence fuel consumption).

The ranking of these interests varies from ship to ship. Unlike the Pursuit Problem, two potential scenarios lead to "improper" results, the first being the case where both ships give way to one another, and the second the one where both ships ignore one another. In the first case time is wasted, and in the second, collision will occur. So, if all ships are "being kind" to others, it does not necessarily provide the best solution.

The following example shows how two ships may reach an agreement using our method. Although the results in this case are quite trivial, they give a hint for using the method in more complicated situations. Following Jin and Koyama, we give each ship two optional actions: *give-way* and *ignore*. Ship 1 pays an amount of a_1 if it gives way, and ship 2 pays a_2. Wasting time costs both ships the amount W, until they realize they are wasting time. A collision causes a disaster, which costs D for both ships. Each ship gets an amount S upon reaching a good solution, and a special prize p for "being kind" to other ships. In order to get rational results, we assume the following relations hold:

$$D > W > S > p > 0$$

Let us consider the four possible outcomes of the game:

1. <u>Ship 1 **gives way**, Ship 2 **gives way**</u>

   In this case the coalition $\emptyset$ gets, of course, 0 for its action. The coalition $\{1\}$ gets $p - a_1$ as it behaves kindly. Similarily, $\{2\}$ gets $p - a_2$. The joint coalition $\{1, 2\}$ gets $-2W + 2p - a_1 - a_2$, since in addition to the prizes for its actions, it also wastes time.

2. <u>Ship 1 **gives way**, Ship 2 **ignores**</u>

   As before, $\emptyset$ gets 0. The coalition $\{1\}$ gets $p - a_1$, and $\{2\}$ gets 0 (as it actually does nothing). The joint coalition $\{1, 2\}$ gets $2S + p - a_1$, as a good solution has been found.

3. <u>Ship 1 **ignores**, Ship 2 **gives way**</u>

   This case is symmetric to case 2. $\emptyset$ gets 0. $\{1\}$ gets 0, and $\{2\}$ gets $p - a_2$. $\{1, 2\}$ gets $2S + p - a_2$.

4. <u>Ship 1 **ignores**, Ship 2 **ignores**</u>

   All three coalitions $\emptyset$, $\{1\}$ and $\{2\}$ gets 0, since both ships ignore and do nothing. However, the joint coalition $\{1, 2\}$ gets $-2D$, as a disaster would result.

Table 1 lists the scores of each agent in each case according to Shapley Value. Consider a non-cooperative game, in which each ship has two pure strategies: *give-way* and *ignore*. The payoff matrix for the game is determined by the payment to each combination of strategies as described in Table 1. The game has two equilibrium points: {*give, ignore*} and {*ignore, give*}. One method to decide on the preferable equilibrium point is to evaluate a_1 and a_2. The negotiation among ships may be reduced to exchanging the values of a_1 and a_2, if the decision rule would be to select {*ignore, give*} if $a_1 < a_2$, and {*give, ignore*} otherwise.[6]

Table 1: Scores of Agents According to Shapley Value

Case	ψ^1	ψ^2
Case 1	$-W + p - a_1$	$-W + p - a_2$
Case 2	$S + p - a_1$	S
Case 3	S	$S + p - a_2$
Case 4	$-D$	$-D$

[6] This is a perfect example of when it would benefit each agent i to lie about its a_i value. See [20] for a treatment of this issue.

B Test Results

Table 2: Comparison Of Performance Results

DAI System	Scenario 1		Scenario 2	
	Moves	Outcome	Moves	Outcome
Autonomous-Agent	10	Escape	10	Stalemate
Limited-Communication	10	Escape	10	Stalemate
Negotiating-Agent	11	Stalemate	11	Capture
Controlling-Agent	17	Capture	12	Capture
Game-Theory	None	Escape	8	Capture

DAI System	Scenario 3		Scenario 4	
	Moves	Outcome	Moves	Outcome
Autonomous-Agent	7	Stalemate	10	Stalemate
Limited-Communication	7	Stalemate	10	Stalemate
Negotiating-Agent	8	Capture	11	Capture
Controlling-Agent	10	Capture	12	Capture
Game-Theory	7	Capture	15	Capture

DAI System	Scenario 5		Scenario 6	
	Moves	Outcome	Moves	Outcome
Autonomous-Agent	7	Stalemate	7	Capture
Limited-Communication	7	Stalemate	7	Capture
Negotiating-Agent	8	Capture	8	Capture
Controlling-Agent	10	Capture	10	Capture
Game-Theory	6	Stalemate	6	Stalemate

References

[1] M. Benda, V. Jagannathan, and R. Dodhiawalla. On optimal cooperation of knowledge sources. Technical Report BCS-G2010-28, Boeing AI Center, Boeing Computer Services, Bellevue, Washington, August 1985. Cited in Les Gasser and Nicolas Rouquette, Representing and Using Organizational Knowledge in Distributed AI Systems. *Proceedings of the 1988 Workshop on Distributed Artificial Intelligence*, May 1988.

[2] H. H. Chin, T. Parthasarathy, and T. E. S. Rayhaven. Structure of equilibria in n-person non-cooperative games. *International Journal of Game Theory*, 3:1–19, 1974.

[3] Randall Davis and Reid G. Smith. Negotiation as a metaphor for distributed problem solving. *Artificial Intelligence*, 20(1):63–109, 1983.

[4] Les Gasser and Nicolas Rouquette. Representing and using organizational knowledge in DAI systems. In Les Gasser, editor, *Proceedings of the 1988 Workshop on Distributed Artificial Intelligence*, Lake Arrowhead, California, May 1988.

[5] Michael R. Genesereth, Matthew L. Ginsberg, and Jeffrey S. Rosenschein. Cooperation without communication. In *Proceedings of The National Conference on Artificial Intelligence*, pages 51–57, Philadelphia, Pennsylvania, August 1986.

[6] J. C. Harsanyi. Oddness of the number of equilibrium points: A new proof. *International Journal of Game Theory*, 2:235–250, 1973.

[7] Yan Jin and Takeo Koyama. Multiagent planning through expectation based negotiation. In *Proceedings of the 10th International Workshop on Distributed Artificial Intelligence*, Bandera, Texas, October 1990.

[8] Ran Levy. A game theoretic approach to distributed artificial intelligence and the pursuit problem. Master's thesis, Hebrew University, 1991.

[9] Thomas W. Malone, Richard E. Fikes, and M. T. Howard. Enterprise: A market-like task scheduler for distributed computing environments. In B. A. Huberman, editor, *The Ecology of Computation*, pages 177–205. North-Holland Publishing Company, Amsterdam, 1988.

[10] John Nash. Non-cooperative games. *Annals of Mathematics*, 54(2), 1951.

[11] John Neumann and Oskar Morgenstern. *Theory of Games and Economic Behaviour*. Princeton University Press, 1947.

[12] J. Rosenmüller. On a generalization of the Lemke-Howson algorithm to noncooperative n-person games. *SIAM Journal on Applied Mathematics*, 21(1):73–79, 1971.

[13] Jeffrey S. Rosenschein and Michael R. Genesereth. Deals among rational agents. In *Proceedings of the Ninth International Joint Conference on Artificial Intelligence*, pages 91–99, Los Angeles, California, August 1985.

[14] 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.

[15] Munindar Singh. Group intentions. In *Proceedings of the 10th International Workshop on Distributed Artificial Intelligence*, Bandera, Texas, October 1990.

[16] L. Stephens and M. Merx. Agent organization as an effector of DAI system performance. In Miroslav Benda, editor, *Proceedings of the 9th Workshop on Distributed Artificial Intelligence*, pages 263–292, Bellevue, Washington, September 1989.

[17] Larry M. Stephens and Matthias B. Merx. The effect of agent control strategy on the performance of a DAI pursuit problem. In *Proceedings of the 10th International Workshop on Distributed Artificial Intelligence*, Bandera, Texas, October 1990.

[18] 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.

[19] Gilad Zlotkin and Jeffrey 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, 1989.

[20] Gilad Zlotkin and Jeffrey S. Rosenschein. Incomplete information and deception in multi-agent negotiation. In *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence*, pages 225–231, Sydney, Australia, August 1991.