Evolutionary Learning of Multiagents Using Strategic Coalition in the IPD Game

Seung-Ryong Yang and Sung-Bae Cho

Department of Computer Science, Yonsei University
134 Shinchon-dong, Sudaemoon-ku, Seoul 120-749, Korea
{saddo,sbcho}@candy.yonsei.ac.kr

Abstract. Social and economic systems consist of complex interactions among its members. Their behaviors become adaptive according to changing environment. In many cases, an individual's behaviors can be modeled by a stimulus-response system in a dynamic environment. In this paper, we use the Iterated Prisoner's Dilemma (IPD) game, which is a simple model to deal with complex problems for dynamic systems. We propose strategic coalition consisting of many agents and simulate their emergence in a co-evolutionary learning environment. Also we introduce the concept of confidence for agents in a coalition and show how such confidences help to improve the generalization ability of the whole coalition. Experimental results show that co-evolutionary learning with coalitions and confidence can produce better performing strategies that generalize well in dynamic environments.

1 Introduction

In biology, co-evolution refers to the evolution of multiple species that affect one another. As one species evolves it changes the relationship that it has with surrounding species. In game theory and evolutionary game, the Iterated Prisoner's Dilemma (IPD) game and its variants have been used widely in modeling various social and economic phenomena [1][2]. It has also been used to study co-evolutionary learning under various conditions, where the primary purpose is not to model a dynamic system but to study how co-evolution can be used in learning novel strategies. For example, co-evolutionary learning of strategies for N (N >2) player IPD games has been studied extensively [3][4][5]. Recently, a new approach to automatic design of modular learning systems has been proposed based on a combination of co-evolutionary learning and fitness sharing [6][7]. The IPD games with multiple choices of cooperation levels and reputation for each player have introduced a new dimension to the study of co-evolutionary learning and brought the IPD game one step closer to the real life situations [8].

This paper investigates the issue of generalization in a co-evolutionary learning system using the IPD game as an example. This issue was first closely examined in [9] using the classical 2-player IPD game. In this paper, we will show how the generalization ability of a coalition can be improved through player's confidence.

J. Lee and M. Barley (Eds.): PRIMA 2003, LNAI 2891, pp. 50–61, 2003.

Table 1. Axelrod's payoff matrix for the 2-player IPD game. It must satisfy following conditions: $T > R > P > S, \quad 2R > T + S$

		Opponent's move	
		Cooperation	Defection
Player's move	Cooperation	Player: R (3 point) Opponent: R (3 point)	Player: S (0 point) Opponent: T (5 point)
	Defection	Player: T (5 point) Opponent: S (0 point)	Player: P (1 point) Opponent: P (1 point)

The rest of this paper is organized as follows: Section 2 introduces the IPD game and the co-evolutionary approach to it. The representation of the IPD game strategy is given. Section 3 explains strategic coalition in the IPD game and how they are formed in evolution. Section 4 presents the experimental results. Finally, Section 5 concludes the paper with a brief summary and a few remarks.

2 Evolutionary Approach to the IPD Game

2.1 Iterated Prisoner's Dilemma Game

In the 2-player IPD game, each player can choose one of the two choices, defection (D) or cooperation (C). Table 1 shows the payoff matrix of the 2-player IPD game. The game is non-zerosum and non-cooperative. The game can be repeated infinitely. None of the players know when the game is supposed to end. According to the conditions given in Table 1, if the game is played for one round only, the optimal strategy is definitely defection. However, if the game is played for many rounds, mutual defection may not be the optimal strategy. Mutual cooperation will perform better than mutual defection for the IPD game.

2.2 Representation of Strategies

The representation of IPD strategies affects various aspects of evolutionary study, including the distribution of behaviors in the initial randomly-chosen population of players and the manner in which recombination and mutation create new players. Several different types of representation have been proposed in the IPD game such as finite state machine, logic tree, If-Skip-Action (ISAc) [10], Markov chain, and neural nets. However, the bit representation is generally used for the convenience of computer-based simulation. Figure 1 describes an example of bit string representation scheme used in this paper. The evolutionary approach to strategy learning for the 2IPD game was popularized by Axelrod [11].

2.3 Evolution of Strategies

In the 2IPD game, players are randomly selected from the population and then play the game. Generally, IPD game is composed of several processes such as

	Look-up Table	History Table
Tit-for-Tat	0 0 1 1 1 1 0 1 1 0 0 1 1 0 0 1	1 0 0 1
CDCD	0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1	1 1 0 1
AllD	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	0 1 0 1

Fig. 1. Bit string representation examples

selecting action, moving and updating history. After enough games are played in a generation, each player's fitness is evaluated according to his total score obtained during the games.

3 Evolving Coalition of Strategies

Evolved strategies may overfit to their current population and perform poorly against good strategies that are not present in the current population [7][9]. One effective approach to address this problem and improve generalization ability of co-evolutionary learning is to automatically learn a group of cooperative strategies, each of which is a specialist in dealing with a certain type of opponents, so that the whole group becomes very robust and generalizes well [6][7]. Since such groups and specialists within a group are not partitioned manually but evolved, this approach can be regarded as a method of automatic design of modular systems, where each module consists of similar specialists. In this section, we define the strategic coalition and then how to form coalition.

3.1 Strategic Coalition

Cooperation among autonomous agents may be mutually beneficial even if the agents are selfish and try to maximize their own expected payoffs [12]. Coalition formation among those agents is an important method for cooperation in multi-agent environment [13]. For example, David and Jeffrey *et al.* have applied coalition formation to model future scenarios in international military coalition as a DARPA project [14][15].

To form coalition among agents, two main principles must be resolved in advance. One is which procedure a group of autonomous agents should use to coordinate their actions and cooperate, that is, how they form a coalition? The other is that among all possible coalitions, what coalition will form, and what reasons and processes will lead the agents to form that particular coalition [12].

In the real society, strategic coalition like this game can be shown easily between individual and individual, individual and group, and group and group. Let $I = \{A_1, A_2, \ldots, A_n\}$ be the collection of individuals in a population that participate in the IPD game. Let $C = \{C_i, C_j, \ldots, C_k\}$, $|C| \geq 2$ be the strategic coalition that can be formed among players. The coalition C is the element of the individual group I: $C \subseteq I$, $|C| \leq |I|$. Every player has his own payoff, p_i, that earns from the IPD game against his opponents. The coalition has the vector, $C = \langle S_C, N_C, f_p, D_C \rangle$. Here, S_C, N_C, f_p, D_C mean the sum of a coalition payoff, the number of agents in the coalition, payoff function, and decision of the coalition, respectively. Now we can define the strategic coalition as follows.

Definition 1. *Coalition Value: Let w be the weight vector of each player's payoff. The payoff of coalition C_p is the average payoff by corresponding confidence of players that participate in the coalition.*

$$S_C = \sum_{i=1}^{|C|} p_i \cdot w_i \tag{1}$$

$$\text{where} \quad w_i = \frac{p_i}{\sum_{i=1}^{|C|} p_i}$$

$$C_p = \frac{S_C}{|C|}$$

Definition 2. *Payoff Function: Agents in the coalition get payoff with a given function. In general, 2-player IPD game follows Axelrod's payoff table.*

Definition 3. *Coalition Identification: Each coalition has its own identification number. This number is generated when the coalition is made with a given condition. This number may be removed when coalition exists no more. This procedure is made autonomously according to evolutionary game process.*

Definition 4. *Decision Making of Coalition: Coalition must have one decision that is made by participants in the coalition. The decision reflects the intention of the whole participants. We have used the weighted voting method for decision making of the coalition in this paper. Decision making of the coalition D_C is determined by function with payoff and confidence.*

$$D_C = \begin{cases} 0 \quad \text{(Cooperation)} & \text{if} \quad 1 < \frac{\sum_{i=1}^{|C|} {C_i}^C \cdot w_i}{\sum_{i=1}^{|C|} {C_i}^D \cdot w_i} \\ \\ 1 \quad \text{(Defection)} & \text{if} \quad 0 < \frac{\sum_{i=1}^{|C|} {C_i}^C \cdot w_i}{\sum_{i=1}^{|C|} {C_i}^D \cdot w_i} \leq 1 \end{cases} \tag{2}$$

Definition 5. *Payoff Distribution: The agents in the coalition get payoffs from the game against another player or coalition. These payoffs must be distributed to each agent with a given function according to its confidence.*

$$p_i = w_i \frac{S_C}{|C|} \tag{3}$$

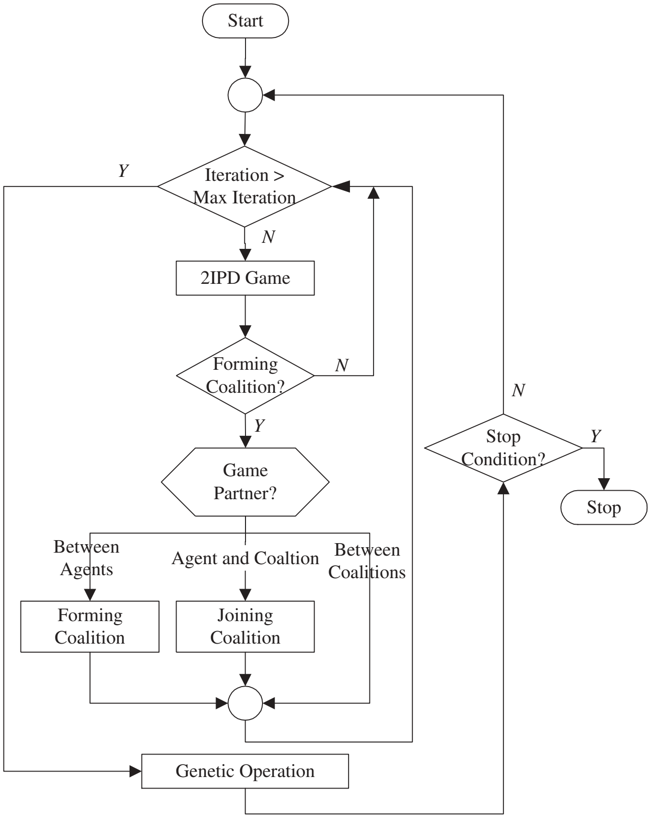

Fig. 2. The 2-player IPD game with coalitions

3.2 Coalition Formation

Coalition formation includes three activities: (1) coalition structure generation, (2) solving the optimization of each coalition, and (3) dividing the value of the generated solution among agents [16]. While the IPD game between players proceeds, each player checks the opponent's intention to form coalition. At this time, the payoff is an important factor to form coalition. In this paper, we propose a different method for evolving coalition. It uses the idea of confidence. A player's confidence is not pre-defined and fixed, but evolved. A player may have different confidences in dealing with different opponents. A player is allowed to take part in deciding the next move of coalition at the rate of his confidence. In other words, coalition's next move is determined by players' confidences.

The IPD game with coalitions is played according to the procedure given in Figure 2. During each generation of evolution, the 2-player game is played between two randomly selected players. After each game, each of the two players considers making (or joining) a coalition to get more payoffs. If all conditions

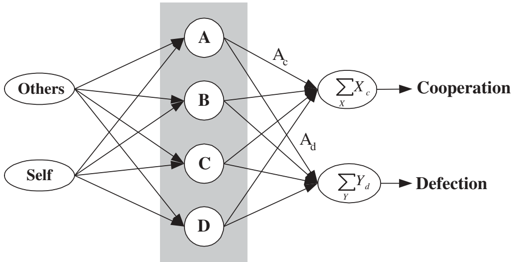

Fig. 3. Combining the confidences of multiple players within a coalition

are satisfied, they make a coalition. As the game is played over and over again, there may be many coalitions that players make. Therefore a player can play a game against a coalition. A coalition can also play against another coalition. Combining coalitions is not considered in our study. There is no action after a game between coalitions.

Once a player has joined a coalition, players within the coalition (including the new one) will play IPD games in round-robin. For k players in a coalition, a total of $k(k\text{-}1)/2$ games will be played. If the total number of players (i.e., k) is greater than a pre-defined maximum coalition size, the weakest player (in terms of the total payoff he/she obtained in all round-robin games) will be removed from the coalition. All players within a coalition are ranked (sorted) according to the total payoff. A player's confidence is assigned in proportion to his/her ranking in the coalition. Figure 3 shows how to combine the confidences of different players in a coalition to produce the coalition's next move. In the figure, **Others** indicates opponent's move and **Self** indicates coalition's move in previous steps. **A**, **B**, **C** and **D** are players within a coalition and $\mathbf{A_d}$ is player's next move, reflected by their confidences, for a given history. For example, if **A**'s next move is cooperation, $\mathbf{A_c}$ is $1\times$**A**'s confidence and $\mathbf{A_d}$ is 0. There is no weight for inputs to a player.

3.3 Evolution of Coalition

Coalitions may be formed or removed from population during evolution. In our experiments, coalition below the average fitness of players in population is removed. In addition to coalition formation as described previously, new coalition may also be generated by crossover of existing coalitions. Crossover of coalitions is done by exchanging the same number of players, as shown in Figure 4. It is important to point out that a player will only exist in the coalition once it has joined in. As a result, weak players may be in the population without belonging to any coalitions (separate from the population) while strong players join coalition. To remedy this problem, new players in the population are generated by recombining players from coalitions. We do not use weak players in the population to generate new players.

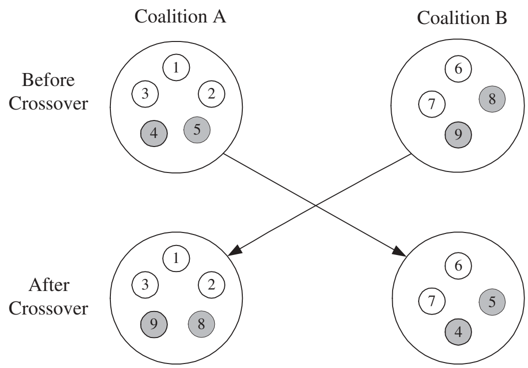

Fig. 4. Crossover used in evolving coalitions

3.4 Gating Strategies in Coalitions

To improve the generalization ability of a coalition, techniques such as opponent modeling and gating can be used. Opponent modeling predicts opponent's next move. Such prediction, if sufficiently accurate, can help the coalition to choose the optimal move. However, it is difficult to model opponent's strategy accurately. In this paper, we use player's confidence to decide which move should be taken by the coalition against an opponent.

A population consists of coalitions that have a different confidence table. A coalition has the same players that other coalitions have. Here, good players are obtained during the evolution. A coalition has a history table for its own and opponent's moves and can use the information to change player's confidence dynamically. The confidence is first randomly initialized as real numbers between 0 and 2 because the average confidence is 1. All coalitions have the fixed number of players that are extracted from coalitions in the last generation of the co-evolutionary process but a different confidence table. If a coalition assigns confidences effectively to the players in a given history, it may get a higher score than the others. In this paper, confidence is represented as real numbers and can be changed by mutation. A new confidence table may also be generated by crossover of confidence tables. Crossover is done by changing confidences at crossover point between coalitions keeping the information of players maintained. The training set for adjusting player's confidence consists of several well known game-playing strategies, such as TFT, Trigger, CDCD, etc [17][18]. During evolution, appropriate confidences are determined from coalitions. Crossover mixes the confidences between coalitions in the population, and mutation changes a specified confidence into a random real number between zero and two.

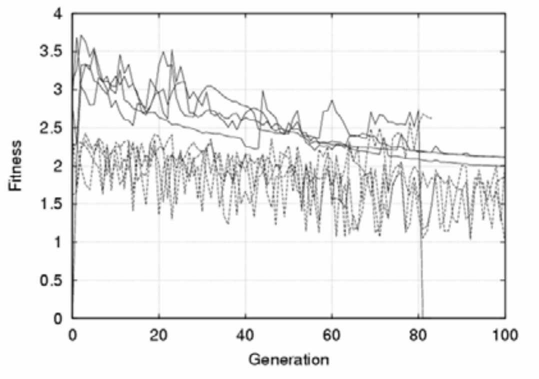

Fig. 5. Average fitness of coalitions and players in the population. Solid lines are for coalitions and dashed lines for players

4 Experimental Results

The first experiment is designed to evolve coalitions using co-evolutionary learning and the second is to evolve player's confidence so that coalitions can play adaptively against different opponents. In our experiments, the payoff matrix used follows the one given in [9].

4.1 Evolving Coalition

In our experiments, we use the population size of 50, crossover rate of 0.6, mutation rate of 0.001 and rank-based selection. A genetic algorithm with one-point crossover and elitism is adopted [19]. For the 2-player game, the history length is 2. The maximum coalition size is 10 and the maximum number of coalitions is one third of the population size (i.e.,16). For the experiments presented here, player's confidence is fixed and not evolved. The evolution of player's confidence will be considered in the next subsection.

Figure 5 shows the average fitness of coalitions for 4 runs during evolution. In the beginning of evolution, the average fitness of coalitions is above that of individual players in the population. However, the difference decreases as time goes by, because players in the population have gradually learned to deal with stronger coalitions. (Notice that new players in the population were generated by recombining strong players from coalitions.) In other words, players in the population also gradually evolve to adapt to their environment.

In the above experiments, the coalition's next move is determined by players' weighted voting at a fixed rate (weight here corresponds to confidence). This means that a coalition's strategy against an opponent can be represented by a single player. A coalition may not be as fit as it should have been in comparison with individual players not in the coalition. Figure 6 shows the number of coalitions during evolution. The number of coalitions could become very small or even zero as a result of fixed player confidence.

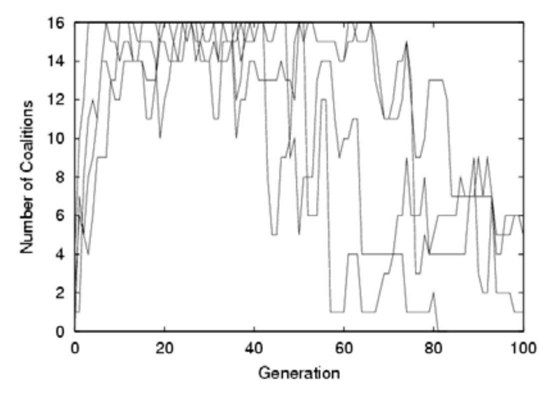

Fig. 6. The number of coalitions in the evolution

Table 2. Training set for evolving player's confidence

Strategy	Characteristics
TFT	Initially cooperate, and then imitate opponent's action
TF2T	Similar TFT, but defect for opponent's 2 defection
Trigger	Initially cooperate. Once opponent defects, continuously defect
AllD	Always defect
CDCD	Cooperate and defect over and over
CCD	Cooperate and cooperate and defect
C10Dall	Cooperate before 10 rounds and always defect after that
Random	Random move

4.2 Evolving Confidences

In this experiment, we use the same experimental setup as before. The training set used in evolving player's confidence consists of seven well-known strategies and a random strategy, as shown in Table 2. Figure 7 shows the average fitness of evolved coalitions.

To evaluate the generalization ability of coalitions with evolving confidences, we have selected 30 top ranked players in a random population of 300 as the test set. Each player in the test set then plays against evolved coalitions in round-robin games. We have experimented it for 10 runs to show the result of evolving confidences. In the test of generalization ability of coalitions, player's confidence is allowed to vary with different opponents. Table 3 shows the performance of a player against the 30 test players. The second and third columns in this table indicate that there is a significant improvement in coalitions' performance when player's confidence is allowed to evolve. The second column indicates the results without the evolution of confidences and the third column indicates those with it. In general, coalitions' winning percentage has increased significantly and the losing percentage decreased. In comparison with players in the training set, the coalitions also perform quite well against the 30 test players. They have only

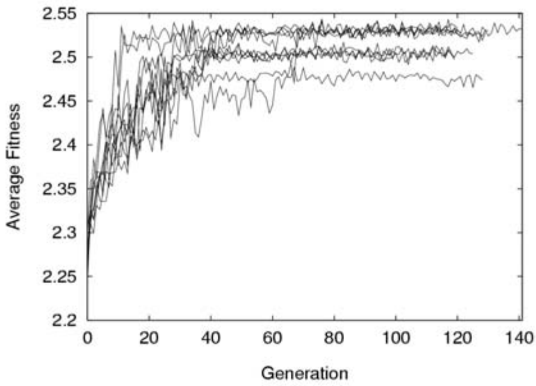

Fig. 7. Average fitness of coalitions

Table 3. Performance against opponent strategies

Strategy	Wins	Ties	Avg.	Opp. Avg.
Before	8.64±4.9	6±2.19	1.84±0.28	1.75±0.59
After	18.55±0.5	4±0.63	2.16±0.07	0.92±0.29
TFT	8	0	1.70	1.77
Trigger	30	0	2.13	0.80
TF2F	7	0	1.54	2.40
AllD	30	0	2.17	0.7
CDCD	0	0	1.05	2.75
CCD	0	0	0.91	3.34
C10Dall	27	0	1.97	1.12

performed worse than AllD on average. They seem to be more likely to tie a game than those players in the training set. This turns out that the coalitions have learned not to lose a game as its top priority. Figure 8 shows the payoff comparison of the evolved strategy with each test strategy.

5 Conclusions

Combining multiple players in a group can be a very effective way of designing a learning system that generalizes well. We have evolved coalitions consisting of a number of players in the IPD game. Each player in a coalition has a confidence which specifies how well he/she is in dealing with an opponent. The confidence is not fixed. It is different for different opponents.

Experiments have been carried out to compare players with and without adaptive confidences. It is found that using adaptive confidences can improve coalitions' performance in testing significantly. Coalitions with adaptive player's confidence can deal with very different opponents, regardless of whether they are cooperators.

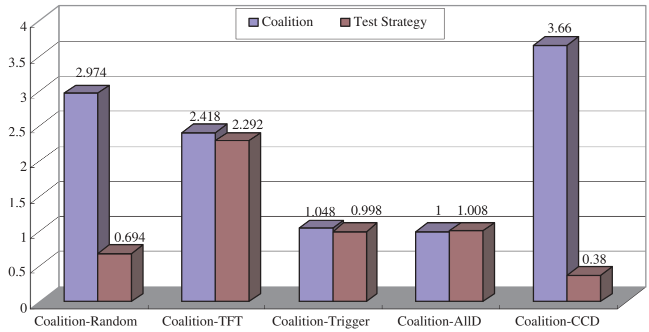

Fig. 8. Payoff comparison of the evolved strategies with each test strategy

Although we have used the 2-player IPD game in this paper, some of the results we have obtained may be applicable to more complex games. It will be very interesting to study the evolution of coalitions and confidences in a real world environment. For example, it is interesting to investigate how coalitions can be formed among different countries in the world, how coalitions can be formed among different parties in a country, how coalitions can be formed in a commercial market, etc.

Acknowledgements. This work was supported by Korea Research Foundation Grant(KRF-2002-005-H20002).

References

1. Ord, T., Blair, A.: Exploitation and peacekeeping: Introducing more sophisticated interactions to the iterated prisoner's dilemma, *Proceedings of the 2002 Congress on Evolutionary Computation*, **2** (2002) 1606–1611
2. Tesfatsion, L.: Agent-based computational economics: Growing economics from the bottom up, *Artificial Life*, **8** (2002) 55–82
3. Yao, X., Darwen, P. J.: An experimental study of N-person iterated prisoner's dilemma games, *Informatica*, **18** (1994) 435–450
4. Seo, Y. G., Cho, S. B., and Yao, X.: Exploiting coalition in co-evolutionary learning, *Proceedings of the Congress on Evolutionary Computation 2000*, **2** (2000) 1268–1275
5. Fletcher, J. A., Zwick, M.: N-Player prisoner's dilemma in multiple groups: A model of multilevel selection, *Proceedings of the Artificial Life VII Workshops*, Portland, Oregon (2000)
6. Seo, Y. G., Cho, S. B., and Yao, X.: The impact of payoff function and local interaction on the N-player iterated prisoner's dilemma, *Knowledge and Information Systems: An International Journal*, **2(4)** (2000) 461–478
7. Darwen, P. J., Yao, X.: Speciation as automatic categorical modularization, *IEEE Transactions on Evolutionary Computation*, **1(2)** (1997) 101–108

8. Yao, X., Darwen, P. J.: How important is your reputation in a multi-agent environment, *Proceedings of IEEE International Conference on Systems, Man, and Cybernetics (SMC'99)*, **2**, IEEE Press, Piscataway, NJ, USA (1999) II-575–II-580
9. Darwen, P. J., Yao, X.: On evolving robust strategies for iterated prisoner's dilemma, *Progress in Evolutionary Computation, Lecture Notes in Artificial Intelligence*, **956**, Springer-Verlag, Heidelberg, Germany (1995) 276–292
10. Ashlock, D., Joenks, M.: ISAc lists, a different representation for program induction, *Genetic Programming 98, Proceedings of the Third Annual Genetic Programming Conference*, Morgan Kaufmann, San Francisco (1998) 3–10
11. Axelrod, R.: The evolution of strategies in the iterated prisoner's dilemma, *Genetic Algorithms and Simulated Annealing*, **3**, Morgan-Kaufmann, San Mateo, CA (1987) 32–41
12. Shehory, O., Kraus, S.: Coalition formation among autonomous agents: Strategies and complexity, *Fifth European Workshop on Modeling Autonomous Agents in a Multi-Agent World*, Springer-Verlag, Heidelberg, Germany (1993) 56–72
13. Shehory, O., Sycara, K., and Jha, S.: Multi-agent coordination through coalition formation, *Agent Theories, Architectures, and Languages* (1997) 143–154
14. Allsopp, D. N., Kirton, P., Bradshaw, M., et al.: Coalition agents experiment: Multiagent cooperation in international coalitions, *IEEE Intelligent Systems*, **17** (2002) 26–35
15. Tate, A., Bradshaw, M., and Pechoucek, M.: Knowledge systems for coalition operations, *IEEE Intelligent Systems*, **17** (2002) 14–16
16. Sandholm, T. W., Lesser, V. R.: Coalitions among computationally bounded agents, *Artificial Intelligence*, **94** (1997) 99–137
17. Axelrod, R.: *The Evolution of Cooperation*, New York: Basic Books (1984)
18. Axelrod, R., Dion, D.: The further evolution of cooperation, *Science*, **242** (1988) 1385–1390
19. Goldberg, D. E.: *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, Reading, MA (1989)