

An Analysis of Efficiency on Multiagent Systems with Symbiotic Learning and Evolution

Md. Tofazzal Hossain^{1*} and Kotaro Hirasawa²

¹Complex Biological Systems Alliance, Japan

²Graduate School of Information, Production, and Systems Research Center, Waseda University, Japan

Abstract

Masbirole (Multiagent Systems with Symbiotic Learning and Evolution) is a new methodology in conventional Multiagent Systems (MASs). In Masbirole, agents evolve considering not only their own benefits and losses, but also the benefits and losses of opponent agents. As a result, Masbirole can escape from Nash Equilibria and obtain better performances than conventional MASs. On the other hand, a newly developed evolutionary computing technique called Genetic Network Programming (GNP) which has the directed graph-type gene structure can develop and design the required intelligence mechanism for agents. As a result, GNP is considered to be well-suited for optimization problems in agents of MASs. Therefore, in this study, a test bed negotiation model is proposed using the evolutionary method of Masbirole as well as the evolutionary method of GNP, with the aim to study the effectiveness and efficiency of Masbirole in dynamic problems. The results obtained by the symbiotic evolution of the Masbirole systems are compared with those obtained by the GNP evolution.

Keywords: Symbiosis; Multiagent systems; Evolutionary computation; Strategy; Negotiation protocol

Introduction

Multiagent Systems with Symbiotic Learning and Evolution (Masbirole) [1,2] is an extension of conventional Multiagent Systems (MASs). Masbirole introduced “Symbiosis”, a biological relationship in the ecosystem in which two species live in close proximity to each other and interact regularly in such a way as to benefit both species, sometimes one species benefits at the other’s expense, and in other cases neither species benefits. Agents in Masbirole can learn or evolve to optimize their objectives considering the benefits/losses of both the agent and opponent agent. Therefore, agents can change their strategies more flexibly.

Conventional Multiagent Systems (MASs) [3,4] are systems composed of multiple interacting intelligent agents. MASs can be used to solve the problems that are difficult for an individual agent to solve. Agents in conventional MASs consider their own objectives, i.e., self-interests only; as a result, the strategy of agents is limited or one-sided. Therefore, the solutions in MASs produce “Nash Equilibria” [5] under the assumption that when one agent is using his self-interested strategy, the other can do no better than using his self-interested strategy. Overcoming the Nash Equilibria is one of the major issues to improve the performance of MAS. For this reason, reaching a solution in dynamic problems is sometimes difficult. Solutions utilizing Masbirole, on the other hand, can achieve Pareto optimality, because the agents consider the objectives of the opponent agents as well as their own. Considering the above background of MAS and biological phenomena in nature, the objective of this study is to examine the effectiveness and efficiency of Masbirole, by comparing its results with those of other intelligent systems of conventional MAS in a real-world complex negotiation problem. Figure 1 shows the basic structure of Masbirole comparing with conventional MAS.

Up to now, the studies of Masbirole have been applied only to MAS test beds, such as the Iterative Prisoner’s Dilemma Game [6], the Tile-world [7], and the Match Type Tile-world [MTT] [8], where Masbirole’s basic characteristics and effectiveness have been examined. However, in these studies the effectiveness of Masbirole is not compared with the other intelligent systems in complex real world problems. Therefore,

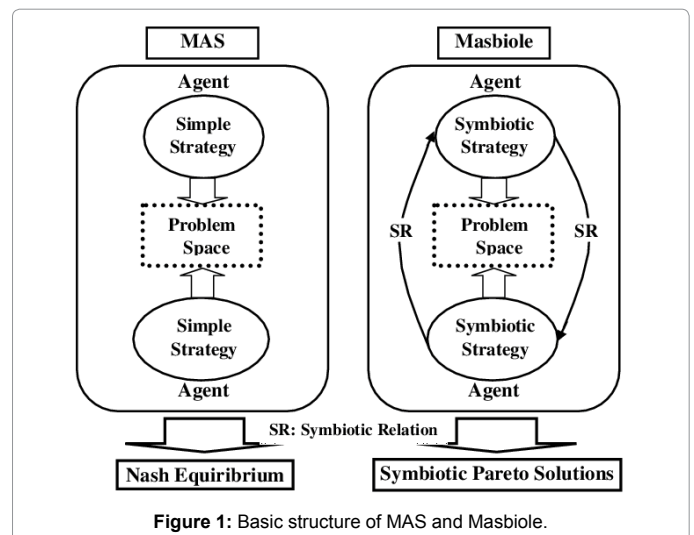


Figure 1: Basic structure of MAS and Masbirole.

in order to study the various effectiveness of Masbirole, a real world negotiation situation, i.e., “Negotiation Model” is proposed and studied that would require the intelligent decision-making ability for agents to achieve solutions. The proposed model is developed by the evolutionary method of Masbirole as well as the evolutionary method of Genetic Network Programming (GNP). As a result, the effectiveness of Masbirole can be directly compared with the results obtained by the GNP system.

***Corresponding author:** Md. Tofazzal Hossain, Kangawaken, Kawasaki Shi, Asao-ku, Takaishi, 4-22-2-503, Japan, 215-0003, Tel: 81-90-1766-8528; E-mail: tapu_290903@yahoo.com

Received July 10, 2015; **Accepted** October 13, 2015; **Published** October 19, 2015

Citation: Hossain MT, Hirasawa K (2015) An Analysis of Efficiency on Multiagent Systems with Symbiotic Learning and Evolution. Int J Swarm Intel Evol Comput 4: 122. doi: 10.4172/2090-4908.1000122

Copyright: © 2015 Hossain MT, et al. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

The evolution method of Masbiole and GNP differs in the evolution process. In Masbiole, the evolutionary method is implemented by the Multiobjective Evolutionary Computation [9], while the evolutionary method of GNP is implemented by the conventional GNP evolution. However, the basic evolution processes of both systems, like initialization of population and the use of genetic operations, are implemented by GNP [10-15]. GNP is an extension of Genetic Algorithms (GA) [16] and Genetic Programming (GP) [17]. The main difference between GA and GP is the representation of solutions. GA evolves strings as solutions and is mainly applied to optimization problems. GP was devised later, aiming to expand the expression ability of GA using tree structures. Instead of strings and trees, GNP has directed graph structures that can generate efficient action sequences based on the partially observable processes. Therefore, GNP is expected to show better performance than GA and GP in complex problems. It has been shown that GNP obtains such better results in some experiments [12-15].

This paper is organized as follows: Section 2 explains the basic concept and study of symbiosis. Section 3 explains the evolutionary process of the proposed model. Section 4 simulates the system with discussions of the results. Finally, section 5 concludes the paper with the future research and extension.

Basic Concept and Study of Symbiosis

Symbiosis comes from the Greek “sym” which means “with” and “bios” which is “life”, so symbiosis means the “state of living with”. It is the interdependence of life [18]. An example of the symbiotic relation is humans and bacteria. A kind of bacteria lives in the intestines of humans. The bacteria partially digest food that the human cannot, allowing the human to finish the digestion. The bacteria benefit by getting food, and the human benefits by being able to digest the food it eats. In the field of Artificial Intelligence (AI) and Evolutionary Algorithm (EA) such as Multi-Objective Genetic Algorithms (MOGAs) [19], several examples of computational models with symbiosis have been presented. Some of them are briefly reviewed below.

The first experiments in this area are reported by Paredis [20]. Paredis introduced the Coevolutionary Genetic Algorithm (CGA). His algorithm is based on the predator-prey model, in which he has a solution population and a test population. He used symbiosis to try to determine the best genetic representation for an individual in a problem. In his CGA, the competitive fitness is the first item discussed in which fitness calculations are dependent on the current population to some degree [21]. The competitive fitness is applied extensively to evolutionary games, including Tic-Tac-Toe [21], Othello [22], and the Iterated Prisoner’s Dilemma [23].

Merry et al. [24] define coevolution by symbiosis as follows: “When the fitness of one system changes, the fitness of another system also changes, and the interdependency is called coevolution. Coevolution is the evolutionary mutual changes of species or organisms that interact with each other”.

According to Agiza et al. [25], coevolution which considers symbiosis is associated with negative and positive interactions. Negative interactions mean predation and competition, while positive interactions include mutualism and sharing. So, there are two main classes of co-evolutionary algorithms that have employed symbiosis: algorithms based on competition relationships and cooperation relationships. Masbiole supposes that there are six kinds of relations possible in ecosystems. This study assumes that these six kinds of symbiotic relations can be used depending on the problems, which go beyond the cooperation and competition.

In the proposed system, when an agent evolves, the interaction with the opponent agent is essential to realize the negotiation between them. In MTT, Mutualism produces the best evolution for both the self-agent and its opponent agent. In this system, Mutualism means sharing the resource mutually. Therefore, it can produce better, but not the best fitness for both agents.

Masbiole can be divided into two kinds of methods: “Learning Masbiole” means an individual changes its own strategy like Reinforcement Learning, and “Evolution Mobile” means several individuals change their strategies by an evolutionary process. Learning Masbiole is suitable for a simple model. Evolution Masbiole can obtain unique Pareto solutions, so it can be applied to complex models such as the ones producing emergent phenomena or total optimization in engineering [8]. In this study evolutionary Masbiole is studied because a relatively complex model is applied using it.

In Masbiole, symbiotic relations between two agents i and j can be defined as the combination of improvement and deterioration in which six kinds of symbiotic relations are considered as shown in Table 1. These relationships can be set arbitrarily between agents. In Masbiole, self improvement and self deterioration are considered as the particular cases of symbiosis, because of the one sided evolution only. In Masbiole, symbiotic evolution is implemented for each symbiotic relation sequentially. In other words, the following procedure is iterated until a terminal condition is reached; two agents are selected as agent i and agent j , and when agent i changes its strategy considering its symbiotic relation toward agent j , the strategy of agent j is fixed. Therefore, the evolution of an agent which has a symbiotic relation toward its opponent agent is implemented by the Masbiole system.

In our previous study [26], a basic bilateral negotiation is studied, where the negotiation system is implemented using the evolutionary computation method of Genetic Network Programming between the buyer and seller. In practical life, multi-issue negotiations are very common. Moreover, we assume that price alone cannot satisfy the demand of the parties where other issues are also relatively important while buying or selling goods or products. Therefore, in [27], a multi-issue negotiation system has been proposed where more than one issue is considered in the negotiation. In another previous study [28], a negotiation system has been developed and implemented considering the evolutionary computation method of Masbiole system. In that study, a kind of fixed strategy is used to realize the symbiotic relation of an agent toward its opponent agent. When the problem is Multiobjective, we find that the use of fixed strategy is not so much effective and interesting way to realize the symbiotic relation of the Masbiole system in the problem. Moreover, to obtain the optimal results, we think that the strategy of agents should be evolved automatically by the evolution; we should not fix or predefine the strategy. This has been realized preciously in this study, which is not realized in the previous studied. In addition, Masbiole considers the Multiobjective ranking method of Evolutionary Computation. Therefore, in the proposed system, Multiobjective Ranking method is used in the evolution; which is not

Symbiotic Relation	Agent i	Agent j	Explanation
Predation	+	-	Agent i benefits, agent j harms
Mutualism	+	+	Both
Competition	+	+	Both agents harm
Altruism	-	+	Agent i harms, agent j benefits
Self-Improvement	+		Self-improvement
Self-Deterioration	-		Self-deterioration

Table 1: Symbiotic relations from own agent i to opponent agent j .

studied in the previous study. Therefore, this is our continuous research efforts to extend and update our previous research study [28]. The most importantly, in the proposed system, the results obtained by the Masbiole system are compared with the results obtained by the GNP based system to realize the effectiveness of Masbiole system precisely.

Proposed Masbiole Model for Negotiation of Buying and Selling

In this section, the Masbiole model for the negotiation of buying and selling process is explained.

Negotiation model

A negotiation model is defined between buying agent i and selling agent j . Agent i and agent j correspond to two organizations negotiating over a product. The negotiation is held via offers and counter-offers. Offers from buying agent i and selling agent j may continue for several rounds in order to fulfill the agreement condition, where the agreement is satisfied when the price offered by buying agent i is equal or larger than the price offered by selling agent j . One round of the negotiation ends when agent i and agent j offer their prices once. In the first round, the negotiation begins from the initial prices by agent i and agent j , which determines the current prices. The next round of the negotiation begins from the prices offered by the agents in the previous round. In this way, the negotiation rounds continue, and the current prices of buying agent i and selling agent j are determined. When the prices from agent i and agent j meet the agreement condition, an agreement is formed and the negotiation ends with a successful deal. If the prices do not meet the agreement condition, the negotiation proceeds to the next round. We experimentally use a maximum of 100 rounds of the negotiation for both agent i and agent j . If the agents cannot reach an agreement by the final round of the negotiation, the negotiation ends with an unsuccessful deal.

Consider a product made by agent j , whose cost is 20.0 yen. An agent i wants to buy the product at a low price as much as possible. In addition, agent i assumes that buying the product at over 80.0 yen will not produce any benefit for him. Therefore, in the negotiation, the private price of agent i and the private price of agent j are set at 80.0 yen and 20.0 yen, respectively. So, it is easy to see that if an agreement is made above the private price of buyer agent i , buyer will lose. In the same way, if an agreement is made below the private price of agent j , i.e., below the production cost of selling agent j , seller will lose. Now both of the buying agent and selling agent need to agree on a certain price. Successful negotiation may generate a profit or a loss for both of the agents, while no profit will be generated when the negotiation is not successful. In this scenario, both buying agent i and selling agent j can utilize various strategies in order to achieve their objectives considering the symbiotic relation to the opponent agent. For example, if agent i has a predation toward agent j , agent i can change its strategy so that the fitness of agent i improves, whereas the fitness of agent j deteriorates.

In order to realize the symbiotic relations to the solutions, when agents reach an agreement, the fitness is calculated considering the benefit or loss agents achieved from the agreement. The terminologies and fitness functions are as follows:

$$f_i = \alpha(B_p^i - A^o) + (1 - \alpha) \beta(\text{Number of times of submitting positive price}), \quad (1)$$

$$f_j = \alpha(A^o - S_p^j) + (1 - \alpha) \beta(\text{Number of times of submitting negative price}), \quad (2)$$

where,

f_i : fitness of agent i ,

f_j : fitness of agent j ,

B_p^i : private price of buying agent i ,

S_p^j : private price of selling agent j ,

A^o : final negotiated price,

α, β : coefficients.

However, considering real world negotiation situations, if the agreement condition is not satisfied until the final round of the negotiation, the fitness is not calculated, because there is no benefit or loss for either agent.

Evolutionary method of masbiole

The evolutionary algorithm of Masbiole is explained below. In evolutionary Masbiole, each individual of an agent has its own strategy. Then, the change of its strategy is implemented by an evolutionary process using Multiobjective evolutionary computation. The whole procedure is as follows:

- I. **Initialization:** Strategies and symbiotic relations between agents are initialized.
- II. **Selection of agents:** The own agent that changes its strategies and its opponent agent are selected.
- III. **Production of offspring:** The strategies of the individuals of the own agent are changed by genetic operations, then offspring of the own agent is generated.
- IV. **Evaluation:** One individual is selected from the own agent and performs the negotiation with the fixed individual of the opponent agent.
- V. **Symbiotic ranking:** From the agreement points of the negotiation model, the ranking of the individuals of the own agent is calculated according to the Multiobjective Ranking method of Evolutionary Computation, considering the symbiotic relation to the opponent agent.
- VI. **Selection of individuals:** New individuals of the own agent are selected for the next generation from the individuals having higher rank.
- VII. **Terminal condition:** The process is returned to procedure 3 until the final generation. In this study, since agents consist of GNP, procedure 1 means the initialization of genes of the GNP population, procedure 2 means the selection of own agent i and opponent agent j for the negotiation and procedure 3 means the implementation of crossover and mutation of GNP. Next, procedures 4, 5 and 6 are explained in the following sub sections in details.

Coevolution method: The evolutionary process of the system uses a method called co-evolution. Akin to the biological phenomena in nature, evolution is held between all individuals of the self-population and the selected individual of the opponent population. Therefore, considering a negotiation between own agent i and opponent agent j , we have 301 individuals for each GNP population. For example, there are two GNP populations, GNP i for buyer agent i and GNP j for seller agent j . In the first generation, one individual from the GNP j population is selected randomly and set as agent j (seller). Then, all the individuals

of the GNP_i population are negotiated with the selected individual of the GNP_j population one by one. After the first generation is finished, one individual with the highest fitness is selected from the GNP_i population to set it as the opponent individual in the next generation, when the individuals of agent *j* evolve. Then, the individuals of the GNP_j population are negotiated with the selected individual of the GNP_i population one by one. After the second generation is finished, one individual with the highest fitness is selected from the GNP_j population. In the third generation, the individual with the highest fitness in the second generation set as the opponent agent. The same procedure is repeated to evolve all the generations.

Ranking method: In each generation, the rank of the individuals of Masbiote agents is calculated according to the symbiotic relation toward the opponent agent. The ranking rule is based on the Multiobjective ranking method of Evolutionary Computation [8]. Therefore, the calculation method of the rank of the individuals is different in different symbiotic relations [8]. In each generation, the better individuals with better ranks are selected for the next population.

For example, Figure 2 shows four simple examples of calculating the rank of the self agent (*Sa*) and its opponent agent (*Oa*) in the case of Mutualism, Harm, Predation and Altruism. In Figure 2, the symbols of A, B, C, ... show the evaluation points and the numbers in () show the ranks of them. Here, the increase of evaluation is regarded as the improvement of it. For example, in the case of Mutualism, the ranks of evaluation point C, F and G become 1, because they are not dominated by any other evaluation point in a sense that both evolutions are improved. In the same way, the ranks of other evaluation points are calculated as follows: A: 3 (dominated by C and F), B: 5 (dominated by C, D, F and G), D: 2 (dominated by F), E: 3 (dominated by F and G). On the other hand, in the cases of Self Improvement and Self Deterioration, the rank is calculated considering *Sa* only.

Evolutionary method of GNP

In this study, agents are constructed by the GNP population, where each individual corresponds to a GNP program. Therefore, in the following, the basic architectures and features of GNP are explained in details.

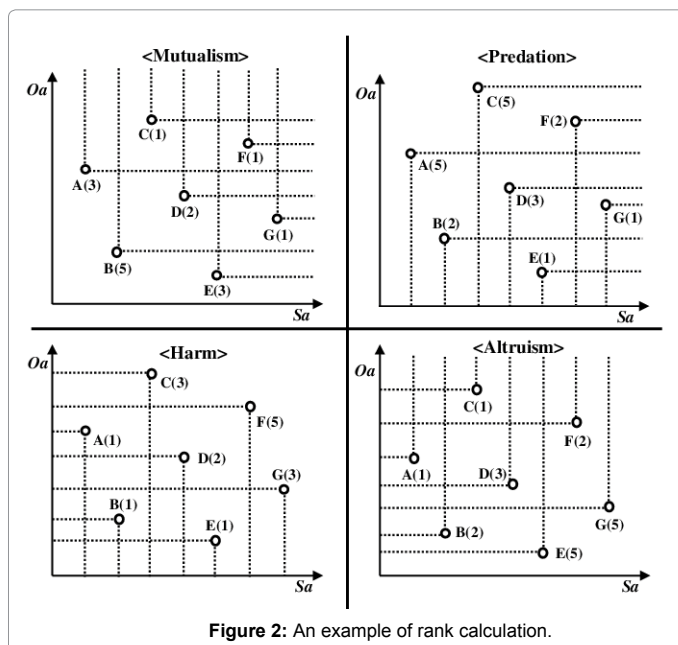


Figure 2: An example of rank calculation.

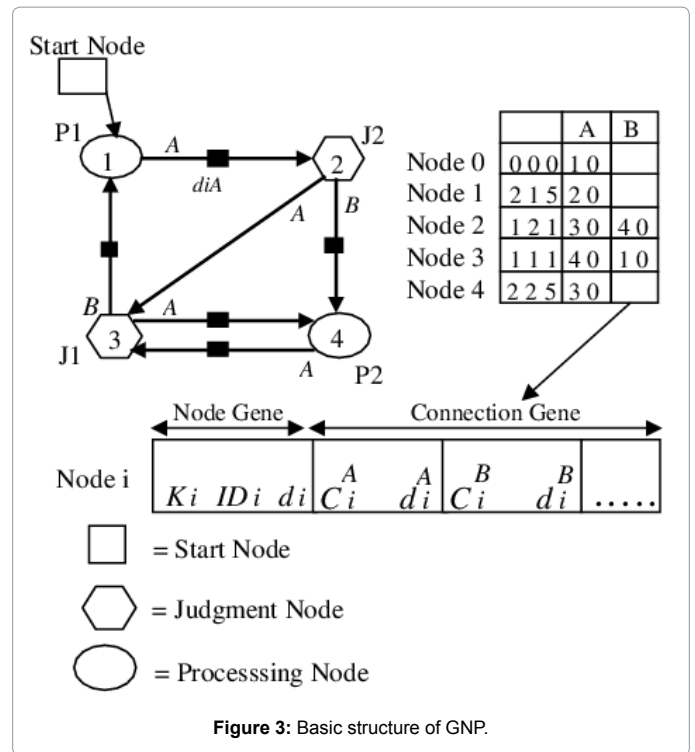


Figure 3: Basic structure of GNP.

Basic structure of GNP: GNP programs are composed of one start node, plural judgment nodes and processing nodes. In Figure 3 of the basic structure of GNP, there are one start node, two judgment nodes and two processing nodes, and they are connected to each other. The start node has no functions and no conditional branch. The only role of the start node is to determine the first node to be executed. Judgment nodes have conditional branch decision functions. Each judgment node returns a judgment result and determines the next node to be executed. Processing nodes work as action/ processing functions. For example, processing nodes determine agents' actions such as increase price, decrease price and so on in the negotiation system. GNP has an ability to use appropriate judgment/processing nodes repeatedly to achieve a task. Therefore, even if the number of nodes is predefined and small, GNP can perform well by making effective node transitions based on reusing nodes. As a result, it is not required to prepare a large number of nodes. The compact structure of GNP is a quite important distinguished characteristic, because it contributes to saving memory consumption and calculation time.

Gene structure of GNP: The graph structure of GNP is determined by the combination of the following node genes. A genetic code of node *i* ($0 \leq i \leq n-1$) is also shown in Figure 3. K_i represents the node type, $K_i = 0$ means start node, $K_i = 1$ means judgment node and $K_i = 2$ means processing node. ID_i represents an identification number of the node function, e.g., $K_i = 1$ and $ID_i = 2$ mean the node is J2 in section 3.3.4. di is the time delay spent on judgment node *i* or processing node *i*. C_i^A, C_i^B, \dots show the node numbers connected from node *i*. d_i^A, d_i^B, \dots mean time delays spent on the transition from node *i* to node C_i^A, C_i^B, \dots , respectively. Judgment nodes determine the upper suffix of the connection genes to refer to depending on their judgment results. For example, if the judgment result is B, GNP refers to C_i^B and d_i^B . However, a start node and processing nodes use only C_i^A and d_i^A , because they have no conditional branch. The above is the basic definition of GNP. More complete architecture and definition of GNP are found in (10 - 15).

Genetic operators in GNP: Like other evolutionary computation, in GNP, genetic operators such as crossover and mutation are used. The most commonly used selection operations are “Roulette Selection”, “Tournament Selection”, “Ranking Selection” and “Elite Preservation”, where Tournament Selection is used and the tournament size is 5. In addition, the best individual is preserved as an elite individual.

Crossover of GNP is executed between two parent individuals to generate two offspring. The flow of a crossover is as follows:

- (1) Two parent individuals are selected using tournament election.
- (2) Some nodes are selected as crossover nodes with the probability of P_c .
- (3) Two parents exchange the selected corresponding nodes having the same node number and two offspring are generated.

Mutation of GNP is executed on an individual basis and a new offspring is generated according to the following.

- (1) An individual is selected using tournament selection.
- (2) Some branches are selected randomly for mutation with the probability of P_m .
- (3) The selected branches are changed randomly and a new offspring is generated.

Node functions of GNP: We used three kinds of processing nodes and two kinds of judgment nodes. The processing nodes increase price (P+), decrease price (P-) and stay at the current price (Not Change). In addition, there are five kinds of P+ nodes to increase the price by 1.0, 2.0, 3.0, 4.0 and 5.0. Processing node P- also includes five kinds of nodes to decrease the price as P+. Two kinds of judgment nodes are J1 and J2. The judgment node J1 compares the difference between the buying price and the selling price in each negotiation round. The judgment node J1 has 4 judgment results, i.e., less than 5% percent difference in the maximum price range of the negotiation, less than 10%percent difference, less than 20% percent difference, or equal or more than 20% percent difference. The judgment node J2 judges/checks the previous action of the agents. The judgment node J2 has 11 judgment results depending on P+, P- and stay. In each round, an action (P+, P- or Not Change) is determined by the GNP evolution.

Simulation Results and Discussion

To realize the effectiveness of Masbiole against GNP, the symbiotic relations between agents are set as shown below:

Case 1: Agent i (Predation) and agent j (Self-improvement).

Case 2: Agent i (Mutualism) and agent j (Self-improvement).

Case 3: Agent i (Self-improvement) and agent j (Self-improvement).

In the above settings, symbiotic relations of an agent are indicated in the parenthesis. Therefore, the evolution of agent i in Case 1 and Case 2 is implemented by the Masbiole system. On the other hand, the evolution of agent j in Case 1 and Case 2, and the evolution of both agent i and agent j in Case 3 are implemented by the GNP evolution, because of the self-improvement strategy. In all the cases, the initial offer of agent i and agent j is set at 1.0 yen and 100.0 yen, respectively, which means the maximum price range of the negotiation is 100.0 yen. In the same way, the private price of agent i and agent j is set at 80.0 yen and 20.0 yen, respectively. The evolution conditions are shown in Table 2. These parameters set are determined experimentally.

Symbiotic Pareto Solution

The solution obtained by the symbiotic evolution can satisfy the Pareto optimality, because they are calculated by the Multi objective ranking method. These Pareto optimal solutions of the Masbiole system can be called “Symbiotic Pareto Solution (SPS)”. As a result, the solutions of Masbiole can be plotted on the Pareto front, because of its Multi objective evolution. Therefore, SPS is calculated on a 2 dimensional space where x-axis represents the fitness of agent i and y-axis represents the fitness of agent j . For example, in the odd generation’s agent i evolved, whereas in the even generation’s agent j evolved (as explained in Section 3.2.1). In this section, only the evolution of odd generations is calculated. To show the behavior of individuals of agents in the Masbiole system concretely, the fitness of the individuals is calculated from the first (1st), middle (250th) and last (500th) generations.

Basic Pareto solutions in Case 1: Figure 4 shows the fitness of agent i and agent j in the first, middle and last generations when agent i evolved using Predation toward agent j . The value of α and β is set at 1.0 and 1.0, respectively. As a result, this is the Pareto solutions, where the second term of Eq. (1) and Eq. (2) are not used. In the first generation (Figure 4 (a)), the results may not show the best performance of the

Generation	= 500
Population Size of an Agent	= 301
Crossover Size	= 120
Mutation Size	= 180
Tournament Size	= 5
Mutation Size	= 0.01
Crossover Size	= 0.1
Processing Nodes	= 11 Kinds
Judgment Nodes	= 2 Kinds
Start Node	=1 Kind

Table 2: Conditions of Evolution.

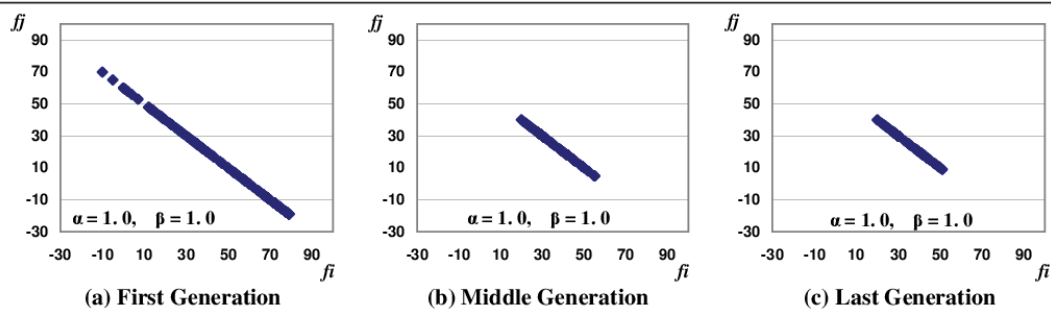


Figure 4: Pareto front in Case 1.

individuals. This is because in early generations agents are usually not intelligent. As the generation goes on, agents have become intelligent through the evolution, which can be realized in the middle and last generation. In the middle generation (Figure 4 (b)), agents have taken more appropriate decisions in the negotiation. Therefore, the agreement is realized in more small area in the two dimensional fitness space. In the last generation (Figure 4 (c)), both agents became more intelligent by an enough number of evolutions. In the last generation, the fitnesses of both agents do not differ so much from the one of the middle generation, which means agents become intelligent enough before the last generation. In Figure 4 (b) and Figure 4 (c), it is shown that the fitness of agent i is higher than the one of agent j , because the predation strategy of agent i improves its own fitness, and deteriorates the opponent's fitness.

Extended Pareto solutions of Case 1: In this simulation, the value of α and β is set at 0.7 and 3.0, respectively. The intelligent decision-making ability of agents can be explained in the same way as the previous experiments. The effect of α and β can be explained as follows: We can see that the fitness values are more distributed than the previous simulations, especially in the first (Figure 5 (a)) generation. This is because agents were not intelligent in the first generation, so agent i and agent j used different numbers of times of positive and negative prices, respectively, to reach an agreement. On the other hand, in the middle and last generation, the fitness values are not distributed widely, because the number of times submitting the positive prices by the intelligent buyer and negative prices by the intelligent seller did not differ so much. Compared with the previous experiments, it is shown that the fitness of agent i and agent j did not distribute so much in the middle and last generations, although we used a large value of β . But, the fitness of agent i is higher than that of agent j , because of the Predation strategy of agent i to agent j .

Basic Pareto solutions in Case 2: Figure 6 shows the fitness of agent i and agent j in the first, middle and last generations in a 2 dimensional

fitness space, when agent i evolved using Mutualism toward agent j . In this case, the value of α and β is set at 1.0 and 1.0, respectively. The intelligent behaviors of agents in the first (Figure 6 (a)), middle (Figure 6(b)) and last (Figure 6 (c)) generation can be explained like the previous experiments. In the last generation, agent i obtained a bit lower fitness than the previous case, because the Mutualism strategy of agent i improves the fitness of the opponent agent as well as its own fitness.

Extended Pareto solutions of Case 2: In this simulation, the values of α and β are set at 0.7 and 3.0 respectively. The intelligent decision making ability of agents (Figure 7) can be explained like the previous experiment of Sensitivity analysis of Case 1. It is relevant to mention that when agent i obtains higher fitness, agent i usually takes a small number of times of positive prices, while agent j takes a large number of times of negative prices. When agent j obtains higher fitness, agent j usually takes a small number of times of negative prices, while agent i takes a large number of times of positive prices.

General Fitness

In this section, the fitnesses of both agent i and agent j are calculated from the agreement points. The fitness averaged over 15 independent simulations is calculated, where the highest fitnesses of both agents are calculated in each generation. The following fitness functions are plotted in this sub section:

$$f_i = (B_p^i - A^O), \dots (3)$$

$$f_j = (A^O - S_p^j), \dots (4)$$

Figure 8 shows the fitness of agent i and agent j in Case 1 (Figure 8 (a)), Case 2 (Figure 8 (b)) and Case 3 (Figure 8 (c)), where the evolution of both agents are considered. Figure 8 (a) shows the fitness of agent i and agent j in Case 1. From the results, it is shown that the fitness of agent i is higher than the fitness of agent j in the last generations,

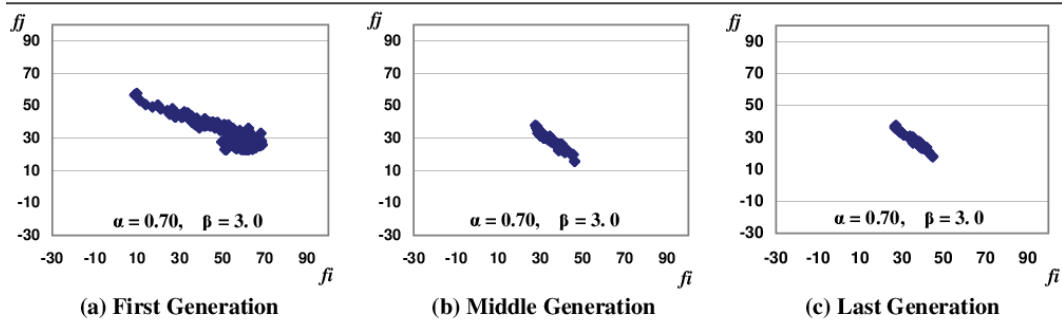


Figure 5: Extended Pareto front in Case 1.

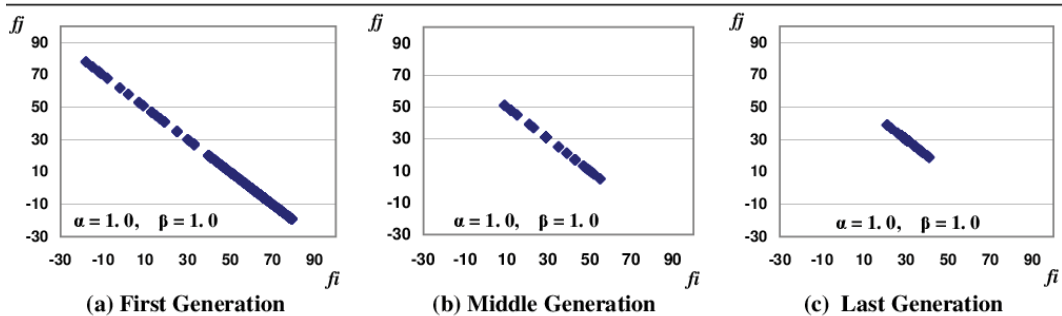


Figure 6: Pareto front in Case 2.

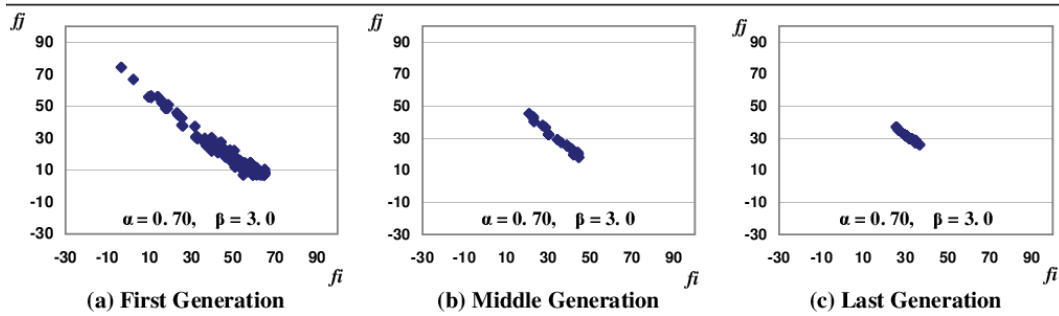


Figure 7: Extended Pareto front in Case 2.

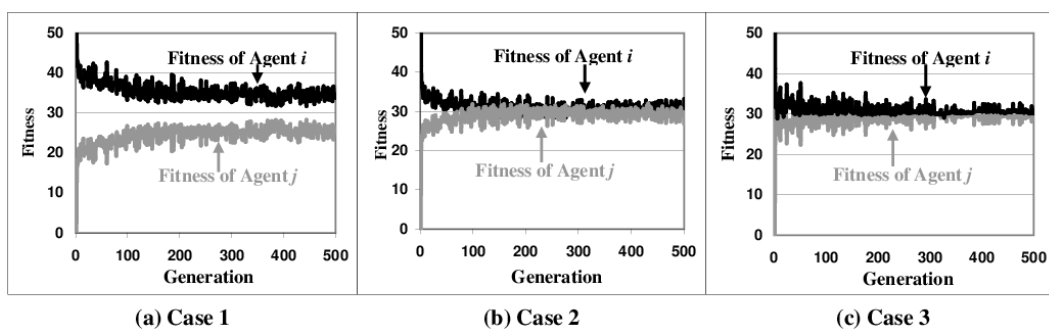


Figure 8: Fitness of agent *i* and *j* in Case 1, 2 and 3.

although agent *j* takes Self Improvement strategy to obtain higher fitness.

Figure 8 (b) shows the fitness of agent *i* and agent *j* in Case 2. From the results it is shown that the fitness of both agents is almost the same, although agent *j* uses Self Improvement strategy to obtain higher fitness. Figure 8 (c) shows the fitness of agent *i* and agent *j* in Case 3, where the objective of both agent *i* and agent *j* is to obtain higher fitness, resulting in a pure competitive negotiation. Accordingly, from the simulation results we can see that both agents obtained competitive fitnesses, i.e., obtained almost the same fitness in the last generations.

Comparison of Fitness Results

To study the performance and effectiveness of symbiotic evolution of the Masbiole system more clearly compared with GNP evolution, we have studied the fitness of agent *i* from the negotiations of Case 1 and Case 3. In case 1, agent *i* uses Predation strategy toward agent *j* to obtain higher fitness for itself and to deteriorate the opponent's fitness.

On the other hand, in Case 3, agent *i* uses Self Improvement strategy to obtain higher fitness for only itself. In Figure 9 (a), it is shown that agent *i* in Case 1 obtained higher fitness compared with the fitness agent *i* obtained in Case 3. This is because the agent *i* with Predation strategy in Case 1 were more intelligent using the symbiotic evolution of the Masbiole system, than the agent *i* with the Self Improvement strategy in Case 3.

In the same way, we have studied the fitness of agent *i* from the negotiations in Case 2 and Case 3. In case 2, agent *i* takes Mutualism strategy toward agent *j* to obtain mutual fitness to agent *j*. On the other hand, in Case 3, agent *i* takes Self Improvement strategy to obtain higher fitness for itself. In Figure 9 (b), we can see that the fitnesses of both agents are almost the same.

From the simulation results, it is found that Masbiole can be applied to the real world complex negotiation problems considering the objectives of both itself and the opponent agent, which is not achievable by GNP.

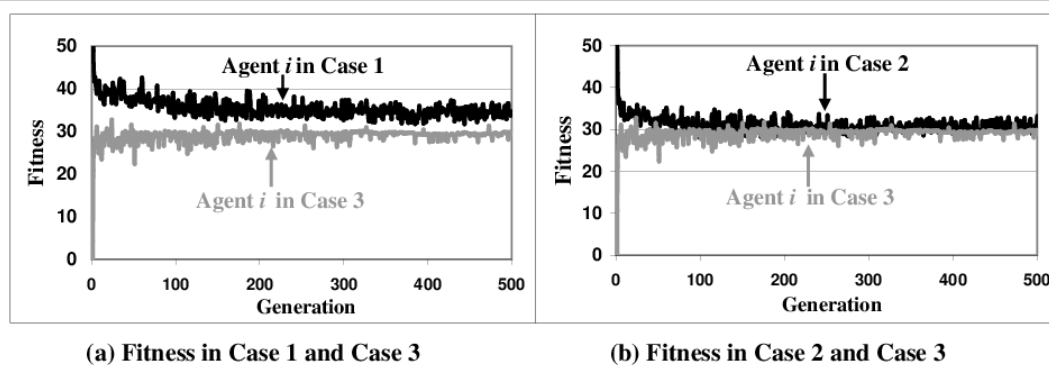


Figure 9. Comparison of fitness of agent *i*.

In the negotiation, Altruism is not applied in a sense that agents usually may not like to take his own loss by using Altruism toward another agent. But depending on the objectives and the problem nature, Altruism can be applicable. For instance, in a negotiation of selling electric power to an industrial company, government may take the Altruism strategy and take a loss for itself (sell for less than production cost), thereby aiming for rapid industrialization that will generate income for citizens and reduce the unemployment rate. Similarly, a seller can choose to utilize Altruism in a situation where there are lots of products in stock, or when the expiration date of the products is very near. In this case Altruism, or taking a loss, will enable the seller to clear his stock as soon possible. In this study, the situations where Altruism would be beneficial are not considered. In the simulation, symbiotic relation Harm i. e., Competition is not included in a sense that both agents cannot deteriorate in the negotiation agreement. Therefore, we find that symbiotic relation Harm is not suitable for a real world negotiation solution, because using the relation Harm, agreement cannot be obtained in the negotiation. As mentioned before, Self-Improvement and Self Deterioration are considered as the particular cases of Masbiole. Therefore, Self-Improvement and Self-Deterioration are not considered for the agent that used the Masbiole system.

Conclusion and Future research

In this study, the effectiveness of Masbiole is studied compared with a newly developed evolutionary computing method named Genetic Network Programming (GNP). In the simulations, instead of depending on MASs test beds, a complex test bed negotiation model is designed considering a competitive negotiation situation. In the simulations, various results are obtained according to the symbiotic relation of an agent toward its opponent agent. These relations could be considered in the real world complex negotiations situation. From the simulation results it is found that Masbiole can obtain better results than GNP in the negotiation situations. In addition, Masbiole could be applied to various situations considering the objective of both itself and the opponent agent. As a result, from this study it is realized that Masbiole is an effective tool for solving complex problems with respect to symbiosis, which could not be solved by the conventional MAS. Moreover, the proposed method could solve the optimization problems more effectively in social science and engineering. To realize the advantage and effectiveness of Masbiole system more precisely and concretely, in future study, we intend to compare the Masbiole system with the other evolutionary approaches, such as Genetic Programming (GP) and Genetic Algorithm (GA) in the real world problem.

References

1. Eguchi T, Hirasawa K, Hu J and Murata J (2002) "Multiagent Systems with Symbiotic Learning and Evolution using Genetic Network Programming", Genetic and Evolutionary Computation Conference, Late Breaking Papers, Erik D. Goodman, Ed., New York, pp.130-137
2. Eguchi T, Hirasawa K and Hu J (2003) "Symbiotic Evolutional Models in Multiagent Systems", In Proc. of the IEEE Congress on Evolutionary Computation, 739-746.
3. Weiss G (2000) Multiagent Systems, MIT Press.
4. Wooldridge M (2002) an Introduction to Multi Agent Systems, Wiley.
5. Nash J (1950) Equilibrium points in n-person games, In Proc. of the Nat. Acad. USA. 36:48-49.
6. Axelrod R (1984): The Evolution of Cooperation, Basic Books.
7. Pollack ME and Ringuette M (1990) "Introducing the tile-world, experimentally evaluating agent architectures", In Proc. of the conference of the American Association for Artificial Intelligence 183-189.
8. Eguchi T, Hirasawa K, Hu J and Ota N (2006) "A Study of Evolutionary Multiagent Models Based on Symbiosis", IEEE Trans. on Systems, Man, and Cybernetics, Part B, 36:179-193.
9. Bagchi TP (1999) Multi objective Scheduling by Genetic Algorithms: Kluwer Academic Publishers.
10. Katagiri H, Hirasawa K and Hu J(2000) "Genetic network programming-application to intelligent agents", In Proc. of the IEEE International Conference on Systems, Man and Cybernetics", 3829-3834.
11. Katagiri H, Hirasawa K, Hu J and Murata J(2001) "Network structure oriented evolutionary model-genetic network programming-and its comparison with genetic programming", In 2001 Genetic and Evolutionary Computation Conference Late Breaking Papers, 219-226, California, USA.
12. Hirasawa K, Okubo M, Hu J and Murata J(2001) "Comparison between genetic network programming (GNP) and genetic programming (GP)", In Proc. of the IEEE Congress on Evolutionary Computation, 1276-1282.
13. Mabu S, Hirasawa K, Hu J and Murata J (2002) "Online learning of Genetic Network Programming", In Proc. of the IEEE Congress on Evolutionary Computation 321-326.
14. Mabu S, Hirasawa K and Hu J (2003) "Genetic Network Programming with Learning and Evolution for adapting to Dynamical Environments", In Proc. of the IEEE Congress on Evolutionary Computation 69-76.
15. Mabu S, Hirasawa K and Hu J (2007) "A Graph-Based Evolutionary Algorithm: Genetic Network Programming (GNP) and Its Extension Using Reinforcement Learning", Journal of Evolutionary Computation 15:3:369-398.
16. Holland J H (1975) Adaptation in Natural and Artificial Systems, University of Michigan Press.
17. Koza J R(1992) Genetic Programming, on the programming of computers by means of natural selection, Cambridge, Mass., MIT Press.
18. Margulis L (1999) The Symbiotic Planet, Contact.
19. Fonseca CM and Fleming PJ(1993) "Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization", In Proc. of the Fifth International Conference on Genetic Algorithms, 416-423, California, Morgan Kaufmann Publishers.
20. Paredis J, Nack T, Fogel DB, Michalewicz Z editors (1998), In Co-evolutionary algorithms The Handbook of Evolutionary Computation, 1st Supplement, 225-238. Institute of Physics Publishing and Oxford University Press.
21. Angeline PJ and Pollack JB (1993): "Competitive Environments evolve Better Solutions for Complex Tasks, In Proc. of the Fifth International Conference on Genetic Algorithms, 264-270, San Mateo, California, and Morgan Kaufmann Publishers.
22. Chong SY, Tan MK and White JD (2005) "Observing the Evolution of Neural Networks Learning to Play the Game of Othello", IEEE Transactions on Evolutionary Computation, 9:3:240-251.
23. Chong SY and Yao X (2005) "Behavioral Diversity, Choices and Noise in the Iterated Prisoner's Dilemma", IEEE Transactions on Evolutionary Computation, 9:6:540-551.
24. Merry U (1999) Organizational strategy on different landscapes: A new science approach, Systemic Practice and Action Research. 12:3:257-278.
25. Agiza HN, Elettrey MF and Ahmed E (1997) "On a generalized model of biological evolution", Journal of Statistical Physics, Vol. 88, No. 3/4, pp. 985-989.
26. Hossain M T, Mabu S, Hirasawa K(2010) "An Evolutionary Negotiation Model using Genetic Network Programming", Journal of Advanced Computational Intelligence and Intelligent Informatics 14:2: 215-223.
27. Hossain MT, Mabu S, Hirasawa K (2011) "An Evolutionary Multi-issue Negotiation Model, IEEJ Trans. on Electrical and Electronic Engineering", EIS 131:1
28. Hossain MT(2012) "An Analysis on a Negotiation Model Based on Multiagent Systems with Symbiotic Learning and Evolution", SICE Journal of Control, Measurement, and System Integration, 5:2:55-62.

Citation: Hossain MT, Hirasawa K (2015) An Analysis of Efficiency on Multiagent Systems with Symbiotic Learning and Evolution. *Int J Swarm Intel Evol Comput* 4: 122. doi: [10.4172/2090-4908.1000122](https://doi.org/10.4172/2090-4908.1000122)