Game Theory Based Coevolutionary Algorithm: A New Computational Coevolutionary Approach

Kwee-Bo Sim, Dong-Wook Lee, and Ji-Yoon Kim

Abstract: Game theory is a method of mathematical analysis developed to study the decision making process. In 1928, Von Neumann mathematically proved that every two-person, zero-sum game with many pure finite strategies for each player is deterministic. In the early 50's, Nash presented another concept as the basis for a generalization of Von Neumann's theorem. Another central achievement of game theory is the introduction of evolutionary game theory, by which agents can play optimal strategies in the absence of rationality. Through the process of Darwinian selection, a population of agents can evolve to an Evolutionary Stable Strategy (ESS) as introduced by Maynard Smith in 1982. Keeping pace with these game theoretical studies, the first computer simulation of coevolution was tried out by Hillis. Moreover, Kauffman proposed the NK model to analyze coevolutionary dynamics between different species. He showed how coevolutionary phenomenon reaches static states and that these states are either Nash equilibrium or ESS in game theory.

Since studies concerning coevolutionary phenomenon were initiated, there have been numerous other researchers who have developed coevolutionary algorithms. In this paper we propose a new coevolutionary algorithm named Game theory based Coevolutionary Algorithm (GCEA) and we confirm that this algorithm can be a solution of evolutionary problems by searching the ESS. To evaluate this newly designed approach, we solve several test Multiobjective Optimization Problems (MOPs). From the results of these evaluations, we confirm that evolutionary game can be embodied by the coevolutionary algorithm and analyze the optimization performance of our algorithm by comparing the performance of our algorithm with that of other evolutionary optimization algorithms.

Keywords: Coevolutionary algorithm, evolutionary stable strategy, game theory, multiobjective optimization problem.

1. INTRODUCTION

Game theory is divided into two categories, cooperative and noncooperative. Noncooperative game theory seeks to fully explain cooperation as well as noncooperation [1]. So in this paper, we bring noncooperative game theory into focus. Von Neumann laid the foundation for a noncooperative game theory in 1928 [2]. As well, in 1951, Nash introduced another concept as the basis for a generalization of Von Neumann's theorem [3]. In his paper, as a minimum requirement for a pair of strategies to be a candidate for the solution of a two-person game, he suggested that each strategy had to be the best reply against the other. Such a pair of strategies, which are called Nash equilibrium, became the basis of modern noncooperative game theory [4].

Since Nash equilibrium was proposed as a solution of the noncooperative game, studies to seek for game equilibrium have begun in earnest. Among these studies, evolutionary game theory is seen as a way of thinking about evolution at the phenotypic level when the fitness of particular phenotypes depend on their frequencies in the population. Lewontin first explicitly applied game theory in evolutionary biology [5]. His approach, however, was to picture a species as playing a game against nature, and to seek strategies that minimized the probability of extinction. Slobodkin and Rapoport have also taken up a similar study [6]. As well, Hamilton sought for an unbeatable strategy, which is essentially the same as an Evolutionary Stable Strategy (ESS) as defined by Maynard Smith and Price [7].

Manuscript received February 2, 2004; accepted October 6, 2004. Recommended by Editorial Board member Eun Tai Kim under the direction of Editor Jin Bae Park. This work was supported by the ITRI of Chung-Ang University in 2004.

Kwee-Bo Sim and Ji-Yoon Kim are with the School of Electrical and Electronic Engineering, Chung-Ang University, 221, Heukseok-dong, Dongjak-gu, Seoul 156-756, Korea (e-mails: kbsim@cau.ac.kr, jonathan@wm.cau.ac.kr).

Dong-Wook Lee is with the Department of Electrical and Computer Engineering, The University of Tennessee, 1508 Middle Drive, Knoxville, TN, 37996-2100, U.S.A. (e-mail: dwlee@wm.cau.ac.kr).

Keeping pace with these researches, the coevolutionary algorithm as a trial of using the evolutionary algorithm among different species has been studied. Hillis [8] demonstrated how simulated evolution can be applied to a practical optimization problem and more specifically how coevolving parasites can improve the coevolution procedure. Simulated evolution represents an idealization of certain aspects of a biological system. As well Hamilton used both computer simulation and mathematical arguments to present how such coevolution is capable of generating genetic diversity. This improved coevolution procedure resulted in an increase in optimization efficiency.

Several researchers of the co-evolutionary algorithm studied this phenomenon from the evolutionary game theory point of view. Kauffman [9] introduced co-evolution based on the NK class of statistical models. He indicated how readily coevolving ecosystems achieve Nash equilibria and how stable to perturbations such equilibria are. In his paper, he described a new class of models with which to investigate the coevolutionary problems. The class of models was related to ESS introduced by Maynard Smith and Price [7]. As well, Rosin and Belew [10] proposed that coevolution was hypothesized by game-theoretic constructions such as Maynard Smith's ESS [11] and the Prisoners' Dilemma [12]. They alleged that it also arises in the evolution of AI game strategies, where the range of potential opponents makes it difficult to establish a single, fixed, exogenous fitness function as is typically used in genetic algorithms [13].

2. GAME THEORY

Game theory is the study involving multi-player decision problems in conflict situations. Such a situation is called the "Game" and game theory provides a mathematical process for selecting an optimum strategy in the face of an opponent who has a strategy of his own. The game is composed of several factors, player, strategy, action and payoff. The player who is the decision maker of the game chooses specific strategy and takes an action that is then rewarded by with a payoff from the game result. In the game theory one usually makes the following assumptions. All players are rational, that is, each player selects the strategy that yields him the greater payoff. The matrix of payoffs can represent various conflicts.

2.1. Maximin criterion and solution of game

Several terminologies connected with game theory are defined as follows. A game is a sequence of plays, some of which may be simultaneous. A strategy is a description of the decisions that a player will make in possible situations. The game is said to be zero-sum if the sum of the players' payoffs is always zero. Let's consider zero-sum games between two players, labeled A and B. Each player has a finite collection of pure strategies. Player A has strategies $a_1, a_2, \cdots, a_n$ and player B has strategies $b_1, b_2, \cdots, b_m$. Let e_{ij} denote the expected payoff to player A when he uses strategy A_i and player B uses strategy B_j. The representation of the game is given by the payoff matrix. It does not include detailed information about the sequences of plays. In this game, we must record both players' payoffs, say $e_1(i,j) = -e_2(i,j) = e_{ij}$. Player A wins and player B loses. Thus, when player B tries to maximize his payoff, he is also attempting to minimize the payoff of player A. This means that player A should look at the payoff he would receive if he plays strategy A_i, i.e., $\min_j e_{ij}$, and choose the strategy that has the largest of these minimum payoffs. This is known as the maximin criterion. Using this criterion, player A can guarantee that his payoff is at least, v_L, the lower value of the game, where $v_L = \max_i \min_j e_{ij}$. Similarly, player B can guarantee that player A's payoff is no more than, v_U, the upper value of the game, $v_U = \min_j \max_i e_{ij}$. If in this game, we have $v_L = v_U$ for a pair of pure strategies, there is said to be a saddle point. Consider a two-person, zero-sum game, in which A has n strategies and B has m strategies. Then $v_L^M = \max_p \min_q e(p,q)$ $= \min_q \max_p e(p,q) = v_U^M$. If p^* and q^* achieve the maximin criterion of the theorem then $e(p^*,q^*) = v_L^M = v_U^M = v$. We say that v is the value of the game and that the value together with the optimal strategies, p^* and q^* are the solution to the game. A pair of strategies p^* and q^* is an equilibrium pair if for any p and q

$$e(p,q^*) \le e(p^*,q^*) \le e(p^*,q). \tag{1}$$

It is possible that there is more than one equilibrium-pair. A pair of strategies (p^*,q^*) in a two-person, zero-sum game is an equilibrium pair if and only if $(p^*,q^*,e(p^*,q^*))$ is a solution to the game. These are available in the case of non-zero-sum games [14].

2.2. Concepts of evolutionary games and evolutionary stable strategy

Nash introduced a new concept of game theory that results from a solution of the non-cooperative game. In his papers [3,4], he said that any two-person, zero-sum or non-zero-sum game with a finite number of pure strategies has at least one equilibrium pair. This is referred to as Nash's theorem and he proved it mathematically. In 1952 Nash introduced Nash equilibrium, which is the solution of a non-cooperative game. According to Nash, each participant of the game has his own strategy set and objective function. Then during the game each player searches for the optimal strategy while other players' strategies are fixed. The game is conducted in this frame and when no player can further improve his criterion, the system is regarded as having reached a state of equilibrium, known as Nash equilibrium [3]. Differently from classical game theory, in evolutionary game theory, there are no rational players involved in selecting a strategy. Instead, strategies of players are selected by Darwinian selection. The primary contribution of evolutionary game theory is the concept of ESS. ESS is proposed by biologist Maynard Smith. He defined ESS as a strategy such that, if all the members of a population adopt it, then no mutant strategy could invade the population under the influence of natural selection [11]. As well, ESS is a refinement concept of Nash equilibrium that does away with the traditional assumption of agent rationality. Instead, Maynard Smith demonstrates that game theoretic equilibrium can be achieved through a process of Darwinian selection [15]. Nevertheless, the ESS is defined as a static concept, and since its introduction numerous other stability concepts have been proposed [16], including those that are more properly rooted in dynamical systems theory [17]. The ESS corresponds to a dynamical attractor [18].

3. GAME THEORY BASED CO-EVOLUTIONARY ALGORITHM

3.1. Coevolutionary algorithm

As far as the author is aware of, Hillis [8] was the first to propose the computational use of predator-prey coevolution. He tested coevolving sorting network architectures and sets of lists of numbers on the sorting networks. The computational study of coevolution initiated by Hillis gave birth to competitive coevolutionary algorithms. In 1994, Paredis introduced Coevolutionary Genetic Algorithms (CGAs). In contrast with the typical all-at-once fitness evaluation of Genetic Algorithms (GAs), CGAs employ a partial but continuous fitness evaluation. Furthermore, the power of CGAs was demonstrated on various applications such as classification [19,20], process control [21], and constraint satisfaction [22]. In addition to this, a number of symbiotic applications have been developed [23-25].

The use of multiple interacting subpopulations has also been explored as an alternate mechanism for coevolving niches using the so-called island model [26-30]. In the island model a fixed number of subpopulations evolve competing rather than cooperating solutions. In addition, individuals occasionally migrate from one subpopulation to another, resulting in a mixing of genetic material. The previous work that has looked at cooperating rather than competing subpopulations has involved a user-specified decomposition of the problem into species [31].

Potter and De Jong have also explored the use of multiple cooperative interaction subpopulations as an alternate mechanism for representing the coevolution of species. The previous work that has looked at coevolving multiple cooperative species in separate subpopulations involved a user-specified decomposition of the problem into species [32]. In this coevolutionary approach, multiple instances of GAs are run in parallel, each instance of which evolves a species of individuals, which are good at particular tasks. This is accomplished by selecting a representative from each of the GA populations and combining them into a single composite agent, which is capable of evaluating the top level goal. These composite agents were called collaborations. Credit from evaluating the composite agent flows back to the individual subcomponents reflecting how well they collaborate with the other subcomponents to achieve the top level goal. This credit is then used by the GA instances to evolve better subcomponents. Such systems are called Cooperative Coevolutionary Genetic Algorithms [33].

3.2. Idea of game theory based Coevolutionary Algorithm

As previously stated, from a mathematical point of view, coevolution has both game theoretical properties and dynamics. For that reason coevolution finally reaches the stable equilibrium state and this state is thought of as an optimal solution because of the dominance property of the game. From these properties, we assume that the coevolutionary algorithm can be made using a game matrix, and as an optimal solution of the game, the equilibrium state of this coevolutionary algorithm can be found. As well, our aim is to combine the coevolutionary algorithm with evolutionary game theory and confirm that this Game theory based Coevolutionary Algorithm (GCEA) can be used in optimization. Although, in particular, we suppose that the population dynamics of evolutionary game theory can be used to most advantageously control the ratio of agents having diverse strategy according to the change of environment. As such, firstly we apply this algorithm to Multiobjective Optimization Problems (MOPs) for an optimization performance evaluation.

Most of the real-world problems encountered by engineers involve simultaneous optimization of several competitive objective functions [34]. The traditional optimization problems attempt to simultaneously minimize cost and maximize fiscal return. In searching solutions for these problems, we discover that there is not a single optimal solution but rather a set of solutions. These solutions are optimal in the wider sense that no other solutions in the search space are superior to them when all objectives are considered. They are generally known as Pareto-optimal solutions [35].

3.3. Definition of multiobjective optimization problem

General MOPs contain a set of n decision variables, a set of k objective functions, and a set of m constraints. In this case, objective functions and constraints respectively become functions of the decision variables. If the goal of MOPs is to maximize the objective functions of the y vector, then

$$\begin{aligned} &\text{maximize } y = f(x) = (f_1(x), \dots, f_i(x), \dots, f_k(x)), \\ &\text{subject to } e(x) = (e_1(x), \dots, e_j(x), \dots, e_m(x)) \le 0, \end{aligned} \tag{2}$$

where $x = (x_1, x_2, \dots, x_n) \in X$, $y = (y_1, y_2, \dots, y_k) \in Y$.

In (2), x is called a decision variable vector and y is called an objective function vector. The decision variable space is denoted by X and the objective function space is denoted by Y. The constraint condition $e(x) \le 0$ determines the set of feasible solutions [34]. The feasible set X_f is defined as the set of decision vectors x that satisfy the constraints $e(x)$: $X_f = \{x \in X \mid e(x) \le 0\}$. The image of X_f, i.e., the feasible region in the objective space, is denoted as $Y_f = f(X_f) = \bigcup_{x \in X_f} \{f(x)\}$ [36]. The set of solutions of MOPs consist of all decision vectors for which the corresponding objective vectors cannot be improved in any dimension without degradation in another [37]. Differently from Single-objective Optimization Problems (SOPs), MOPs have a set of solutions known as the Pareto optimal set. This solution set is generally called non-dominated solutions and is optimal in the sense that no other solutions are superior to them in the search space when all objectives are considered.

3.4. Several approaches to solve MOPs

Classical methods for generating the Pareto-optimal set aggregate the objective functions of MOPs into a single parameterized objective function. Then the optimizer systematically varies the parameters of this function. Several optimizations are performed in order to achieve a set of solutions that approximate the Pareto-optimal set [34]. Some representatives of this class of techniques include the weighting method [39], the constraint method [39], goal programming [40], and the min-max approach [41].

The first exploration for treating objective functions separately in Evolutionary Algorithms (EAs) was launched by Schaffer. In his dissertation [42,43], Schaffer proposed the Vector Evaluated Genetic Algorithm (VEGA) for searching a solution set to solve MOPs. He created VEGA to find and maintain multiple classification rules in a set-covering problem. VEGA attempted to achieve this goal by selecting a fraction of the next generation using one of each of the attributes (e.g., cost, reliability) [44]. Other approaches that search solutions for MOPs include those of Fourman [45], Kursawe [46], and Hajela and Lin [47]. However, as none of them makes direct use of the actual definition of Pareto-optimality, different non-dominated individuals are generally assigned different fitness values [48].

Goldberg [49] proposed a Pareto-based fitness assignment approach known as the Pareto Genetic Algorithm (Pareto GA). The idea of this algorithm is to assign high probability to all non-dominated individuals in the population. This method consists of assigning rank 1 to the non-dominated individuals and removing them from contention, then finding a new set of non-dominated individuals, ranked 2, and so forth. He named these rankings as Pareto ranking. Fonseca and Fleming [50] have proposed a different scheme, whereby an individual's rank corresponds to the number of individuals in the current population by which it is dominated. Therefore non-dominated individuals are assigned the same rank, while dominated ones are penalized according to the population density of the corresponding region of the trade-off surface [51]. Horn and Nafpliotis also proposed a tournament selection based on Pareto dominance [44]. Moreover distributive search is very important in Pareto GA. The goal of Pareto GA is to explore all Pareto optimal solution sets distributed along the Pareto frontier. To achieve this goal Goldberg and Richardson introduced the concept of fitness sharing in their paper [52]. It is within the range of possibility to search distributive solutions using the fitness sharing that makes highly fitted candidates share fitness with others in their surroundings [53]. With the introduction of non-dominance Pareto-ranking and fitness sharing, Pareto GA has now become a type of standard in the sense that the Pareto GA provides a very efficient way to find a wide range of solutions to a given problem. Although this approach proposed by Goldberg was further developed in [54] and led to many applications [55-57], all of these approaches are based on the concept of Pareto ranking and use either sharing or mating restrictions to ensure diversity.

3.5. Design game theory based Co-evolutionary Algorithm to solve MOPs

In this section, we design a Game theory based Coevolutionary Algorithm to solve MOPs. Through the evolutionary game, players try to optimize their own objective function and all individuals of the population are regenerated after players have been rewarded. The reward value is determined from the game matrix. For example, in the case of minimization MOPs, which have two variables x, y and objective functions $f_1(x,y)$, $f_2(x,y)$, the architecture of populations for GCEA is designed as follows. In Fig. 1, fitness F_i is determined from the game matrix where $i = 0,1,\cdots,n$. The game matrix is defined in the previous tables and two populations coevolve with each other through the game. Payoff of the game for each population, G_i, is calculated from the differences between two objective functions.

$$\begin{aligned} G_1(v_i, v'_i) &= G_1((x_i, y_i),(x'_i, y'_i)) \\ &= f_1(x_i, y_i) - f_2(x'_i, y'_i), \\ G_2(v_i, v'_i) &= G_2((x_i, y_i),(x'_i, y'_i)) \\ &= f_2(x'_i, y'_i) - f_1(x_i, y_i). \end{aligned} \tag{3}$$

From these payoffs, the fitness of each player is calculated

$$\begin{aligned} F_i &= 100 \times \frac{G_1((x_i, y_i),(x'_i, y'_i))}{\alpha}, \\ F'_i &= 100 \times \frac{G_2((x_i, y_i),(x'_i, y'_i))}{\alpha}, \end{aligned} \tag{4}$$

where α is constant to normalize the fitness of F_i or F'_i so that α must be $\max_i |G_k((x_i, y_i), (x'_i, y'_i))|$. From these establishments, GCEA is as follows:

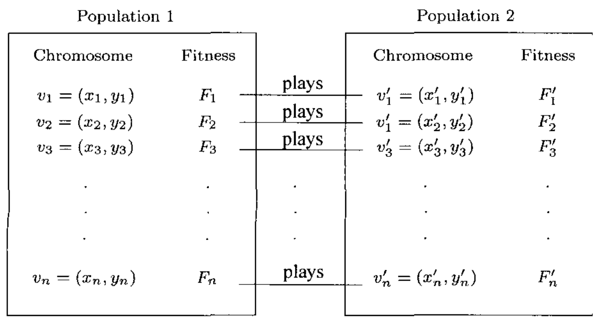


Fig. 1. Architecture of population for GCEA.

Step 1: Two populations are randomly generated as in Fig. 1.
Step 2: The Player selected in the first population plays with that from the second population and then he is paid off using Table 1 and (3).
Step 3: The Player in the second population is paid off using Table 2 and (3).
Step 4: The fitness of each player F_n and F'_n is updated using (4).
Step 5: The process from **Step 2** to **Step 3** is executed for all individuals of each population one by one.
Step 6: Each population is regenerated separately using genetic algorithms.
Step 7: The process from **Step 2** to **Step 6** is executed.

Keeping these ideas, we apply GCEA to MOPs.

4. TEST PROBLEMS AND EVALUATION

While an assortment of evolutionary approaches and their variations have been successfully applied to solving MOPs, in recent years some researchers have investigated particular topics of evolutionary multiobjective search. In spite of this variety of approaches, there is a lack of studies that compare the performance and different aspects of these approaches. In this chapter, we provide a systematic comparison of several multiobjective evolutionary algorithms. The test problems considered here are used in Zitzler's paper [58], and cover six representative MOPs, which mention a corresponding test function and are constructed in the following the guidelines in [59].

Table 1. The game matrix for population 1 of GCEA.

	v'_1	v'_2	$\cdots$	v'_n
v_1	$G_1(v_1, v'_1)$	$G_1(v_1, v'_2)$	$\cdots$	$G_1(v_1, v'_n)$
v_2	$G_1(v_2, v'_1)$	$G_1(v_2, v'_2)$	$\cdots$	$G_1(v_2, v'_n)$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
v_n	$G_1(v_n, v'_1)$	$G_1(v_n, v'_2)$	$\cdots$	$G_1(v_n, v'_n)$

Table 2. The game matrix for population 2 of GCEA.

	v'_1	v'_2	$\cdots$	v'_n
v_1	$G_2(v_1, v'_1)$	$G_2(v_1, v'_2)$	$\cdots$	$G_2(v_1, v'_n)$
v_2	$G_2(v_2, v'_1)$	$G_2(v_2, v'_2)$	$\cdots$	$G_2(v_2, v'_n)$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
v_n	$G_2(v_n, v'_1)$	$G_2(v_n, v'_2)$	$\cdots$	$G_2(v_n, v'_n)$

4.1. Test MOPs

In the previous chapters, we introduced various established evolutionary algorithms for solving MOPs. In spite of this variety, there is a lack of studies that compare the performance and different aspects of these approaches. From among these studies we introduce several researches. On the theoretical side, Fonseca and Fleming discussed the influence of different fitness assignment strategies on the selection process [51]. Zitzler provides a systematic comparison of multiobjective EAs, including a random search strategy as well as a single objective evolutionary algorithm using objective aggregation. The basis of this empirical study is formed by a set of well-defined, domain-independent test functions that allow the investigation of independent problem features. The functions considered here cover the range of convexity, nonconvexity, discrete, multimodal, deceptive, and non-uniform Pareto fronts. Deb has identified several features that may cause difficulties for multiobjective evolutionary algorithms in converging to the Pareto-optimal front and maintaining diversity within the population [36,38]. The test functions used in this paper is as follows.

- Test function T_1 has a convex Pareto-optimal front:

$$f_1(x_1) = x_1,$$
$$g(x_2,\dots,x_n) = 1 + 9 \cdot \frac{\sum_{i=2}^{n} x_i}{n-1},$$
$$h(f_1, g) = 1 - \sqrt{\frac{f_1}{g}},$$

where $n = 30$, and $x_i \in [0,1]$.

- Test function T_2 has a non-convex Pareto-optimal front:

$$f_1(x_1) = x_1,$$
$$g(x_2,\dots,x_n) = 1 + 9 \cdot \frac{\sum_{i=2}^{n} x_i}{n-1},$$
$$h(f_1, g) = 1 - \left(\frac{f_1}{g}\right)^2,$$

where $n = 30$, and $x_i \in [0,1]$.

- Test function T_3 represents the discreteness feature; its Pareto-optimal front consists of several non-contiguous convex parts:

$$f_1(x_1) = x_1,$$
$$g(x_2,\dots,x_n) = 1 + 9 \cdot \frac{\sum_{i=2}^{n} x_i}{n-1},$$
$$h(f_1, g) = 1 - \sqrt{\frac{f_1}{g}} - \left(\frac{f_1}{g}\right)\sin(10\pi f_1),$$

where $n = 30$, and $x_i \in [0,1]$.

- Test function T_4 contains 21^9 local Pareto-optimal sets and therefore tests for the evolutionary algorithm's ability to deal with multimodality:

$$f_1(x_1) = x_1,$$
$$g(x_2,\dots,x_n) = 1 + 10(n-1) + \sum_{i=2}^{n} (x_i^2 - 10\cos(4\pi x_i))$$
$$h(f_1, g) = 1 - \sqrt{\frac{f_1}{g}},$$

where $n = 30$, $x_i \in [0,1]$ and $x_2, \cdots, x_n \in [-5,5]$.

- Test function T_5 describes a deceptive problem and distinguishes itself from the other test functions in that x_i represents a binary string:

$$f_1(x_1) = 1 + u(x_1),$$
$$g(x_2,\dots,x_n) = \sum_{i=2}^{n} v(u(x_i)),$$
$$h(f_1, g) = \frac{1}{f_1},$$

where $u(x_i)$ gives the number of ones in the bit vector x_i,

$$v(u(x_i)) = \begin{cases} 2 + u(x_i) & if \quad u(x_i) < 5 \\ 1 & if \quad u(x_i) = 5 \end{cases}$$

and $n = 11$, $x_1 = \{0,1\}^{30}$ and $x_2, \dots, x_n \in \{0,1\}^5$.

- Test function T_6 includes two difficulties caused by the non-uniformity of the objective space: Firstly, the Pareto-optimal solutions are non-uniformly distributed along the global Pareto front. Secondly, the density of the solutions is least near the Pareto-optimal front and most away from the front:

$$f_1(x_1) = 1 - \exp(-4x_1)\sin^6(6\pi x_1),$$
$$g(x_2,\dots,x_n) = 1 + 9 \cdot \left(\frac{\sum_{i=2}^{n} x_i}{n-1}\right)^{0.25},$$
$$h(f_1, g) = 1 - \left(\frac{f_1}{g}\right)^2,$$

where $n = 10$ and $x_i \in [0,1]$.

We apply GCEA proposed in our paper to these six test MOPs and analyze the experimental results.

4.2. Experimental results and analysis

Figs. 2-13 display optimized solutions of MOPs by evolutionary algorithms introduced in Zitzler's paper and our GCEA. To analyze these results we cite figures, which display optimized solutions using the several evolutionary algorithms proposed by Zitzler in his paper [60]. He used 8 different evolutionary algorithms to optimize six test MOPs. In these cited figures, the evolutionary algorithms used are as follows:

- SPEA: The Strength Pareto Evolutionary Algorithm.
- SOEA: A Single-Objective Evolutionary Algorithm using weighted-sum aggregation.
- NSGA: The Nondominated Sorting Genetic Algorithm.
- VEGA: The Vector Evaluated Genetic Algorithm.
- HLGA: Hajela and Lin's weighted-sum based approach.
- NPGA: The Niched Pareto Genetic Algorithm.
- FFGA: Fonseca and Fleming's multiobjective EA.
- RAND: A random search algorithm.

Figs. 2, 4, 6, 8, 10, and 12 show optimized solutions of MOPs by eight different EAs previously introduced in Zitzler's paper [60]. To analyze the evaluation of our GCEA, we cite these results. Figs. 3, 5, 7, 9, 11, and 13 display simulated optimization results using our GCEA. GA parameters used are as follows, the number of generations is 500, population size is 100, one-point crossover rate is 0.8, and

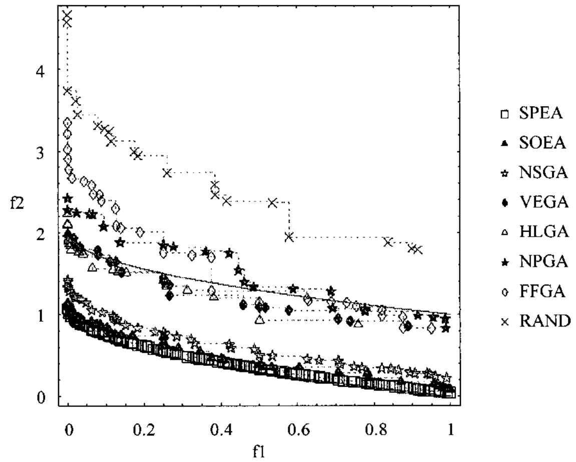


Fig. 2. The Pareto fronts of T_1 searched by 8 different EAs [58].

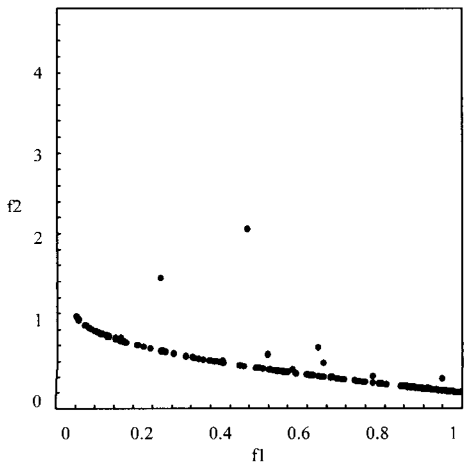


Fig. 3. The ESS of T_1 searched by GCEA.

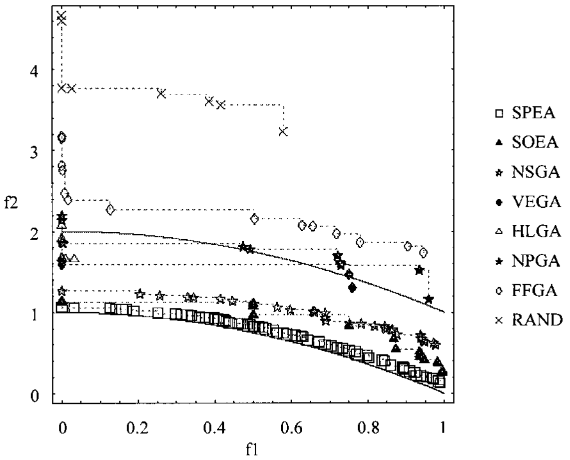


Fig. 4. The Pareto fronts of T_2 searched by 8 different EAs [58].

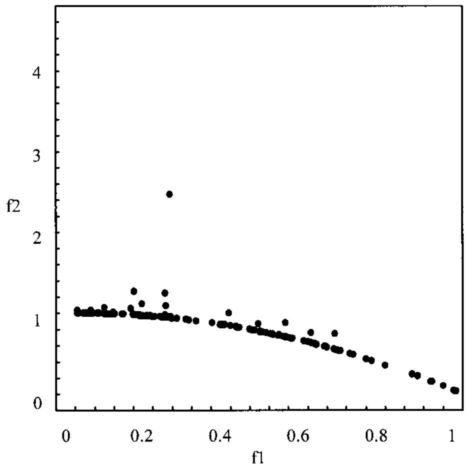


Fig. 5. The ESS of T_2 searched by GCEA.

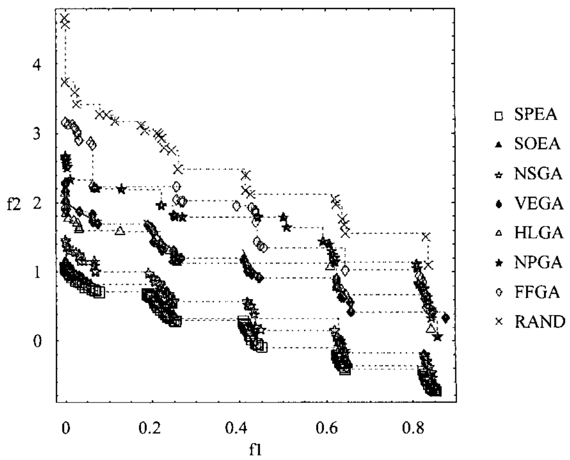


Fig. 6. The Pareto fronts of T_3 searched by 8 different EAs [58].

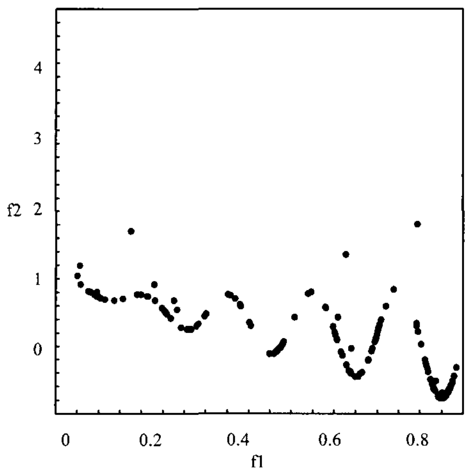


Fig. 7. The ESS of T_3 searched by GCEA.

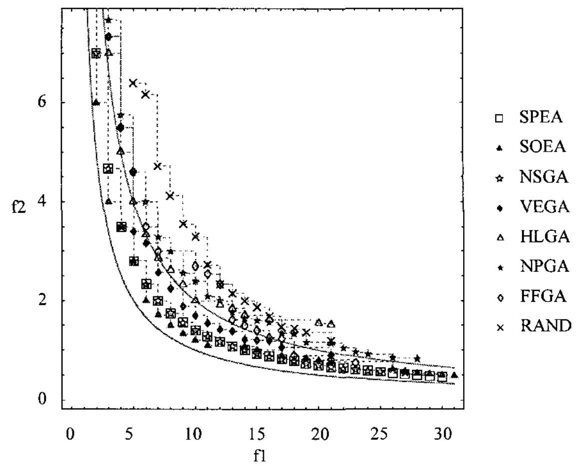


Fig. 10. The Pareto fronts of T_5 searched by 8 different EAs [58].

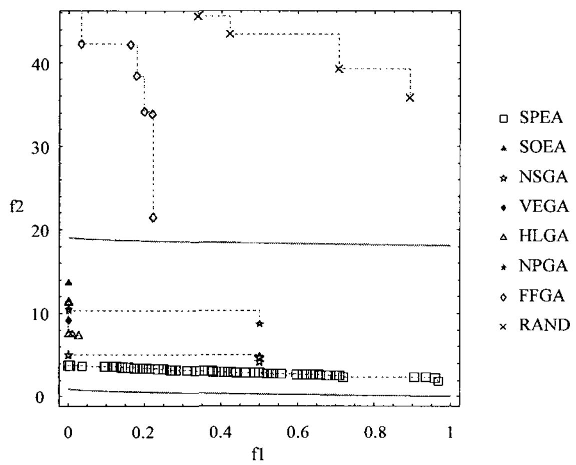


Fig. 8. The Pareto fronts of T_4 searched by 8 different EAs [58].

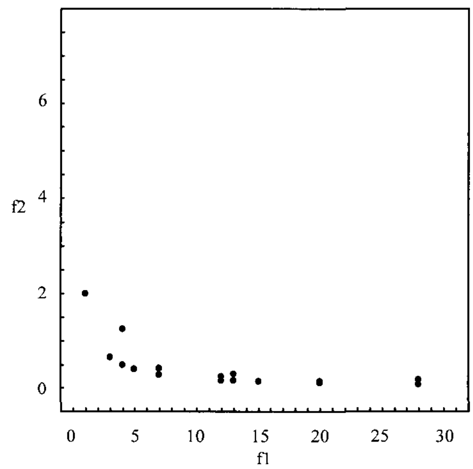


Fig. 11. The ESS of T_5 searched by GCEA.

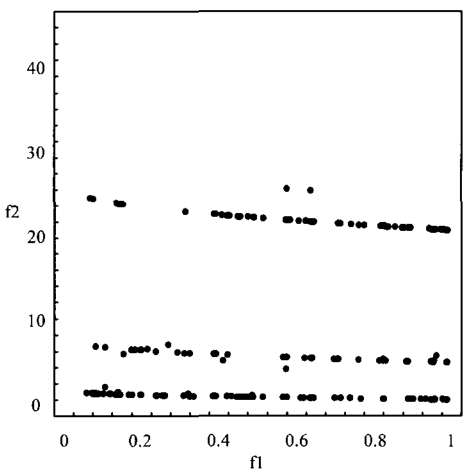


Fig. 9. The ESS of T_4 searched by GCEA.

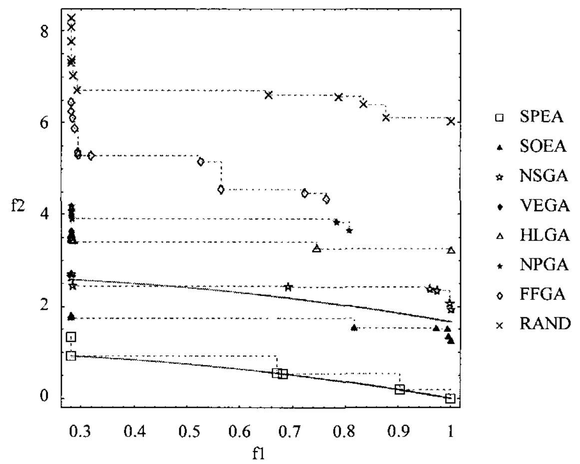


Fig. 12. The Pareto fronts of T_6 searched by 8 different EAs [58].

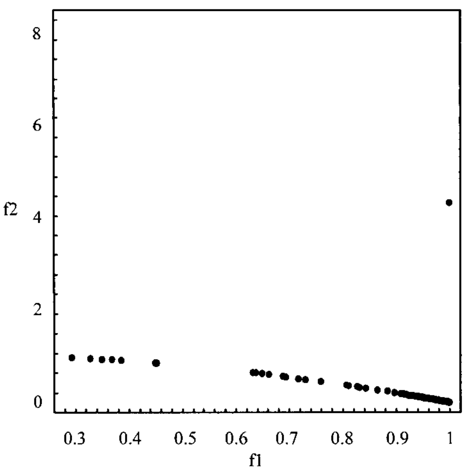


Fig. 13. The ESS of T_6 searched by GCEA.

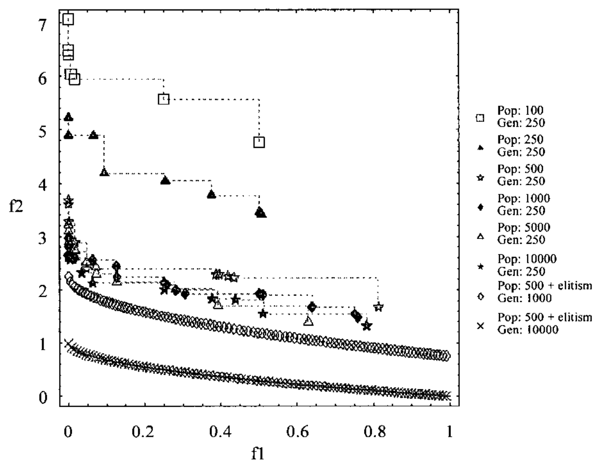


Fig. 14. Comparison of different population sizes on T_4 using NSGA [58].

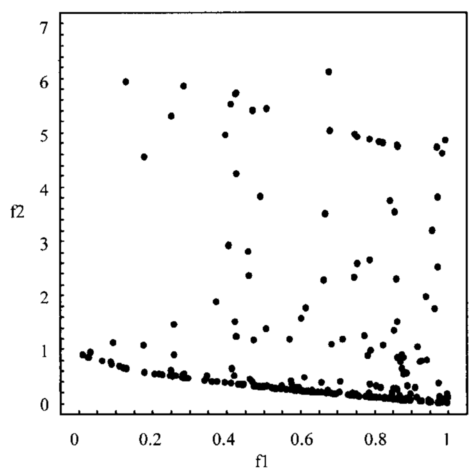


Fig. 15. The ESS of T_4 searched by GCEA.

mutation ratio is 0.01. These are the same parameter values used in Zitzler's simulations. From comparing the Pareto fronts in the cited figures with ESS found by GCEA, we can see that ESS exists in the Pareto front or Pareto optimal set. So we conclude that GCEA can determine MOP's solution set. For every test MOPs, GCEA can find the optimized solution set of these problems.

Though these results are successful, finding the Pareto-optimal front of T_4 test MOP is too hard to find and so Zitzler applied elitism to his algorithm. But GCEA does not need this method. In place of this concept, a larger population size is needed. Figs. 14 and 15 show another experimental result using only adapted algorithm and population size for T_4 test MOP.

In this experiment GCEA does not use elitism but only increased population size. From Fig. 14, though the optimization performance of NSGA is better according to increasing population size, the Pareto-optimal front can determine when the elitism is appended. But from Fig. 15, we confirm that GCEA can find the Pareto-optimal front simply by increasing population size. From these previous results, we can see that ESS found by GCEA is very similar to the Pareto front. So we conclude that this algorithm newly proposed by us can search the Pareto-optimal front of MOPs, as well, this GCEA is more concise than other evolutionary algorithms used in Zitzler's experiments.

5. CONCLUSIONS

In this paper, we introduce a brief history and several concepts of game theory and coevolutionary algorithms. Some researchers have studied the relation between these two fields. As well, we proposed Game theory based Coevolutionary Algorithm, which is based on Evolutionary Game Theory, as a new approach to solve evolutionary problems that are particularly involved in these fields. Moreover Evolutionary Stable Strategy is the basis of GCEA. To evaluate the performance of GCEA, we used Multiobjective Optimization Problems. Although ESS is the equilibrium solution of the evolutionary game involved in mathematics and economics, we confirm that GCEA can be used as a new optimization approach from the simulation result. So we presume that GCEA may be useful in implementing the real robot controller for the environment, which has several conflict objective functions. In our future works, we will focus our study on the real robot controller implementations.

REFERENCES

[1] L. A. Dugatkin and H. K. Reeve, Eds, *Game Theory and Animal Behavior*, Oxford University Press, New York, 1998.

[2] J. Von Neumann and O. Morgenstern, *The Theory of Games and Economic Behavior*,

Princeton University Press, 1944.
[3] J. F. Nash, "Noncooperative games," *Annals of Mathematics*, vol. 54, p. 289, 1951.
[4] J. F. Nash, *Essays on Game Theory*, Edward Elgar Pub., 1997.
[5] R. C. Lewontin, "Evolution and the theory of games," *Journal of Theoretical Biology*, vol. 1, pp. 382-403, 1961.
[6] L. B. Slobodkin and A. Rapoport, "An optimal strategy of evolution," *Quarterly Review of Biology*, vol. 49, pp. 181-200, 1974.
[7] J. Maynard Smith and G. R. Price, "The logic of animal conflict," *Nature*, vol. 246, pp. 15-18, 1973.
[8] W. D. Hillis, "Co-evolving parasites improve simulated evolution as an optimization procedure," *Artificial Life II*, Addison-Wesley, pp. 313-324, 1991.
[9] S. A. Kauffman and S. Johnsen, "Co-evolution to the edge of chaos: Coupled fitness landscapes, poised states, and co-evolutionary avalanches," *Artificial Life II*, Addison-Wesley, pp. 325-369, 1991.
[10] C. D. Rosin and R. K. Belew, "New methods for competitive coevolution," *Evolutionary Computation*, vol. 5, no. 1, pp. 1-29, 1997.
[11] J. Maynard Smith, *Evolution and the Theory of Games*, Cambridge University Press, 1982.
[12] R. Axelrod, "The evolution of strategies in the iterated prisoner's dilemma," *Genetic Algorithms and Simulated Annealing*, L. Davis, Ed., Pitman Publishing, London, pp. 32-41, 1987.
[13] P. J. Angeline and J. B. Pollack, "Competitive environments evolve better solutions for complex tasks," *Proc. of the 5th Int. Conf. on Genetic Algorithms*, Morgan Kaufmann, pp. 264-270, 1993.
[14] R. Weber, "Mathematics for operational research," http://www.statslab.cam.ac.uk/~rrw1/mor/, November, 2001.
[15] S. G. Ficici and J. B. Pollack, "Game theory and the simple coevolutionary algorithm: Some preliminary results on fitness sharing," *Proc. of GECCO 2001 Workshop on Coevolution*, 2001.
[16] S. Lessard, "Evolutionary stability: One concept, several meanings," *Theoretical Population Biology*, vol. 37, pp. 159-170, 1990.
[17] G. W. Rowe, I. F. Harvey, and S. F. Hubbard, "The essential properties of evolutionary stability," *Journal of Theoretical Biology*, vol. 115, pp. 269-285, 1985.
[18] S. H. Strogatz, *Nonlinear Dynamics and Chaos*, Addison-Wesley, 1994.
[19] J. Paredis, "Coevolutionary computation," *Artificial Life*, vol. 2, no. 4, pp. 355-379, 1995.
[20] J. Paredis, "Steps towards coevolutionary classification neural networks," *Artificial Life IV*, MIT Press, pp. 102-108, 1994.
[21] J. Paredis, "Coevolutionary process control," *Proc. of Int. Conf. on Artificial Neural Networks and Genetic Algorithms,* Springer, Vienna, pp. 394-398, 1997.
[22] J. Paredis, "Coevolutionary constraint satisfaction," *Proc. of PPSN-III, Lecture Notes in Computer Science* 866, Springer Verlag, pp. 46-55, 1994.
[23] J. Paredis, "The symbiotic evolution of solutions and their representations," *Proc. of the 6th Int. Conf. on Genetic Algorithms*, Morgan Kaufmann Publishers, pp. 359-365, 1995.
[24] J. Paredis, "Symbiotic coevolution for epistatic problems," *Proc. of European Conf. on Artificial Intelligence*, W. Wahlster, Ed., John Wiley & Sons, pp. 228-232, 1996.
[25] J. Paredis, "Coevolving cellular automata: Be aware of the red queen!," *Proc. of the 7th Int. Conf. on Genetic Algorithms*, Morgan Kaufmann Publishers, pp. 393-400, 1997.
[26] P. Grosso, *Computer Simulations of Genetic Adaptation: Parallel Subcomponent Interaction in a Multilocus Model*, Ph. D. dissertation, University of Michigan, Ann Arbor, MI, 1985.
[27] J. Cohoon, S. Hegde, W. Martin, and D. Richards, "Punctuated equilibria: A parallel genetic algorithm," *Proc. of the 2nd Int. Conf. on Genetic Algorithms*, pp. 148-154, 1987.
[28] C. Petty, M. Leuze, and J. Grenfenstette, "A parallel genetic algorithm," *Proc. of the 2nd Int. Conf. on Genetic Algorithms*, pp. 155-161, 1987.
[29] R. Tanese, "Distributed genetic algorithms," *Proc. of the 3rd Int. Conf. on Genetic Algorithms*, Morgan Kaufmann Publishers, pp. 434-439, 1989.
[30] D. Whitley and T. Starkweather, "Genitor II: A distributed genetic algorithm," *Journal of Experimental and Theoretical Artificial Intelligence*, vol. 2, pp. 189-214, 1990.
[31] M. A. Potter, K. A. De Jong, and J. J. Grefenstette, "A coevolutionary approach to learning sequential decision rules," *Proc. of the 6th Int. Conf. on Genetic Algorithms*, Morgan Kaufmann, San Francisco, pp. 366-372, 1995.
[32] M. A. Potter and K. A. De Jong, "A cooperative coevolutionary approach to function optimization," *Proc. of Parallel Problem Solving from Nature*, Springer, Berlin, pp. 249-257, 1994.
[33] M. A. Potter, K. A. De Jong, and J. J. Grefenstette, "A coevolutionary approach to learning sequential decision rules," *Proc. of the 6th Int. Conf. on Genetic Algorithms*, Morgan Kaufmann, San Francisco, pp. 366-372, 1995.
[34] E. Zitzler, K. Deb, and L. Thiele, "Comparison of multiobjective evolutionary algorithms: Empirical results," *Proc. of Genetic and*

Evolutionary Computation Conf. Workshop Program, pp. 121-122, 1999.

[35] E. Zitzler and L. Thiele, "Multiobjective optimization using evolutionary algorithms - A comparative case study," *Proc. of 5th Int. Conf. on Parallel Problem Solving from Nature*, A. E. Eiben, T. Beack, M. Schoenauer, and H. -P. Schwefel, Eds, Springer, Berlin, pp. 292-301, 1998.

[36] E. Zitzler, *Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications*, Ph. D. dissertation, The Swiss Federal Institute of Technology, Zurich, November, 1999.

[37] E. Zitzler, "Multiobjective evolutionary algorithms: A comparative case study and the strength pareto approach," *IEEE Trans. on Evolutionary Computation*, vol. 3, no. 4, pp. 257-271, 1999.

[38] K. Deb, "Multi-objective genetic algorithms: Problem difficulties and construction of test functions," *Technical Report No. CI-49/98*, Department of Computer Science, University of Dortmund, Germany, 1998.

[39] J. L. Cohon, *Multiobjective Programming and Planning*, Academic Press, New York, 1978.

[40] R. E. Steuer, *Multiple Criteria Optimization: Theory, Computation, and Application*, Wiley, New York, 1986.

[41] J. Koski, "Multicriterion optimization in structural design," *New Directions in Optimum Structural Design*, E. Atrek, R. H. Gallagher, K. M. Ragsdell, and O. C. Zienkiewicz, Eds., Wiley, pp. 483-503, 1984.

[42] J. D. Schaffer, *Some Experiments in Machine Learning Using Vector Evaluated Genetic Algorithms*, Ph. D. dissertation, Vanderbilt University, 1984.

[43] J. D. Schaffer, "Multiple objective optimization with vector evaluated genetic algorithms," *Proc. of Int. Conf. on Genetic Algorithms and Their Applications*, pp. 93-100, 1985.

[44] J. Horn and N. Nafpliotis, "Multiobjective optimization using the niched Pareto genetic algorithm," *IlliCAL Report 93005*, University of Illinois, Urbana, Illinois, 1993.

[45] M. P. Fourman, "Compaction of symbolic layout using genetic algorithms," *Proc. of the 1st Int. Conf. on Genetic Algorithms*, pp. 141-153, 1985.

[46] F. Kursawe, "A variant of evolution strategies for vector optimization," *Proc. of Parallel Problem Solving from Nature, Lecture Notes in Computer Science 496*, H.-P. Schwefel and R. Maenner, Eds., Springer-Verlag, Berlin, pp. 193-197, 1991.

[47] P. Hajela, and C.-Y. Lin, "Genetic search strategies in multicriterion optimal design," *Struct. Optim.*, vol. 4, pp. 99-107, 1992.

[48] C. M. Fonseca and P. J. Fleming, "Multiobjective optimization and multiple constraint handling with evolutionary algorithms - Part I: A unified formulation," *IEEE Trans. on Systems, Man, and Cybernetics Part A: Systems and Humans*, vol. 28, no. 1, pp. 26-37, 1998.

[49] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley, Massachusetts, 1989.

[50] C. M. Fonseca and P. J. Fleming, "Genetic algorithms for multiobjective optimization: Formulation, discussion and generalization," *Proc. of the 5th Int. Conf. on Genetic Algorithms*, Morgan Kaufmann, San Mateo, pp. 416-423, 1993.

[51] C. M. Fonseca and P. J. Fleming, "An overview of evolutionary algorithms in multiobjective optimization," *Evolutionary Computation*, vol. 3, no. 1, pp. 1-16, 1995.

[52] D. E. Goldberg and J. J. Richardson, "Genetic algorithms with sharing for multi-modal function optimization," *Proc. of the 2nd Int. Conf. on Genetic Algorithms*, pp. 41-49, 1987.

[53] J. Horn, N. Nafpliotis, and D. E. Goldberg, "A niched Pareto genetic algorithm for multiobjective optimization," *Proc. of IEEE World Congr. on Computational Intelligence*, vol. 1, pp. 82-87, 1994.

[54] N. Srinivas and K. Deb, "Multiobjective optimization using non-dominated sorting in genetic algorithms," *Evolutionary Computation*, vol. 2, no. 3, pp. 221-248, 1994.

[55] C. Poloni, "Hybrid GA for multiobjective aerodynamic shape optimization," *Genetic Algorithms in Engineering and Computer Science*, M. Galan, G. Winter, J. Periaux, and P. Cuesta, Eds., John Wiley, Las Palmas, Spain, pp. 397-415, 1995.

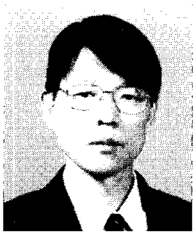
[56] R. Makinen, P. Neittaanmaki, J. Periaux, M. Sefrioui, and J. Toivonen, "Parallel genetic solution for multiobjective MDO," *Parallel CFD 96*, Elsevier, Capri, 1996.

[57] M.-O. Bristeau, R. Glowinski, B. Mantel, J. Periaux, and M. Sefrioui, "Genetic algorithms for electromagnetic backscattering: Multi-objective optimization," *System Design Using Evolutionary Optimization: Genetic Algorithms*, John Wiley, 1999.

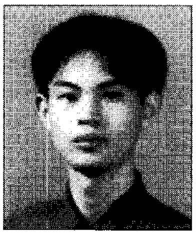
[58] E. Zitzler, K. Deb, and L. Thiele, "Comparison of multiobjective evolutionary algorithms: Empirical results," *Proc. of Genetic and Evolutionary Computation Conf. Workshop Program*, Orlando, pp. 121-122, 1999.

[59] K. Deb, "Multi-objective genetic algorithms: Problem difficulties and construction of test problems," *Evolutionary Computation*, vol. 7, no 3, pp. 205-230, 1999.

[60] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach," *IEEE Trans. on Evolutionary Computation*, vol. 3, no. 4, pp. 257-271, 1999.



Kwee-Bo Sim received the B.S. and M.S. degrees in the Department of Electronic Engineering from Chung-Ang University, Seoul, Korea, in 1984 and 1986 respectively, and the Ph.D. degree in the Department of Electronic Engineering from the University of Tokyo, Japan, in 1990. Since 1991, he has been a Faculty Member at the School of Electrical and Electronic Engineering at Chung-Ang University, where he is currently a Professor. His research interests are Artificial Life, Intelligent Robot, Multi-Agent System, Distributed Autonomous Robotic System, Learning & Adaptational Algorithms, Soft Computing (Neural Network, Fuzzy System, and Evolutionary Computation), Artificial Immune System, Network Intrusion Detection System, Evolvable Hardware, Artificial Brain, etc. He is a member of IEEE, SICE, RSJ, KITE, KIEE, ICASE, and KFIS.



Dong-Wook Lee received the B.S., M.S., and Ph.D. degrees in the Department of Control and Instrumentation Engineering from Chung-Ang University in 1996, 1998, and 2000, respectively. He is currently a Post-doctoral Researcher in the Department of Electrical and Computer Engineering at the University of Tennessee, Knoxville, U.S.A. His research interests include Artificial Life, Evolutionary Computation, Evolutionary Robotics, Artificial Brain, and Artificial Immune System.

Ji-Yoon Kim received the B.S. and M.S. degrees in the School of Electrical and Electronic Engineering from Chung-Ang University, Seoul, Korea, in 2002 and 2004 respectively, and is currently working towards his doctorate. His research interests are Artificial Life, Evolutionary Computation, Distributed Autonomous Robotic Systems, Machine Learning, and Artificial Intelligence.