International Journal of Fuzzy Logic and Intelligent Systems, vol. 12, no. 1, March 2012, pp. 29-35
http://dx.doi.org/10.5391/IJFIS.2012.12.1.29 pISSN 1598-2645 eISSN 2093-744X

Adaptive Control of Strong Mutation Rate and Probability for Queen-bee Genetic Algorithms

Sung Hoon Jung*

**Department of Information and Communications Engineering,
Hansung University, Seoul 136-792, Korea**

Abstract

This paper introduces an adaptive control method of strong mutation rate and probability for queen-bee genetic algorithms. Although the queen-bee genetic algorithms have shown good performances, it had a critical problem that the strong mutation rate and probability should be selected by a trial and error method empirically. In order to solve this problem, we employed the measure of convergence and used it as a control parameter of those. Experimental results with four function optimization problems showed that our method was similar to or sometimes superior to the best result of empirical selections. This indicates that our method is very useful to practical optimization problems because it does not need time consuming trials.

Key words : Genetic algorithms, adaptive control, queen-bee evolution, function optimization

1. Introduction

Genetic Algorithms(GAs) have been successfully applied to many optimization problems to date [1–9]. Recently GAs have been used for multiobjective optimization problems as a new application area [10–12]. However, GAs sometimes showed poor performances in the complex multi-modal problems because individuals could easily fall into the local optimum areas and it was difficult to get out of the areas [6, 13–15]. In order to solve this problem and to improve the performances of GAs, we introduced a queen-bee evolution method [16]. Although it showed very good performances than existing methods, it had a critical problem that the strong mutation rate and probability of the method should be selected by a trial and error method empirically. This empirical selection is a quite annoying and time-consuming task because the parameters are selected through a lot of experiments. Since the strong mutation rate and probability are the major factor of the performances of queen-bee GAs, we should carefully devise a new algorithm to solve this problem.

In this paper, we propose an adaptive control method of the strong mutation rate and probability by employing a measure of convergence. If the measure of convergence indicates a converged state (in other words, most individuals fall into local optimum areas), then the strong mutation rate and probability should be increased in order to enforce exploration. Otherwise, they should be decreased. The measure of convergence should be carefully devised for good performances of our method. First, we used the change of average fitness of individuals from the previous generation. That is, if the average fitness of current generation is less than or equal to that of previous generation, then we regard as a converged state. Second, we took the failed ratio of evolution of individuals. That is, a specific percent of whole offsprings are failed to evolve from their parents, we also regard it as a converged state. From this control of the strong mutation rate and probability, the individuals fallen into local optimum areas have more chances to escape the local optimum areas and to approach to the global optimum.

We applied our method to four function optimization problems in order to measure the performances of our method. Experimental results showed that our method was considerably practical in that it produced considerably similar results to the best ones of previous trial and error methods. Although it may be possible to find the better parameters by empirical trial and error methods, it is not practical because it is quite time consuming task. This paper is organized as follows. Section 2 describes the proposed adaptive control method of strong mutation rate and probability. Experimental results and discussion are provided in section 3. We conclude our paper in section 4.

Manuscript received Oct. 6, 2011; revised Nov. 2, 2011; accepted Nov. 3, 2011.
* Corresponding author(shjung@hansung.ac.kr)
A preliminary version of this paper was presented at the 36th Conference of Korea Information Processing Society. This research was financially supported by Hansung University.

2. Adaptive Control of Strong Mutation Rate and Probability

We first describe the background why adaptive control is needed and next introduce the proposed GA with adaptive control of strong mutation rate and probability.

2.1 Background of Adaptive Control

In queen-bee GAs [16], we chose several values for the strong mutation rate and probability and experimented with those values. We could find from those experiments that the proper strong mutation rate and probability were different according to experimental functions. That is, the strong mutation rate and probability should be set to small values for relatively simple uni-modal functions, while those should be set to large values for complex multi-modal functions in order to escape the local optimum areas. In [16], chosen values for strong mutation rate and probability are not changed during the whole evolution process. Since the degree of convergence to local optimum areas is changed according to generations, it is better that changing the strong mutation rate and probability according to the degree of convergence than the constant values.

Individuals in GAs are moved to the global or local optimum areas as the generation is progressed. Thus they are converged to several areas after some generations. If one of the converged areas is the global optimum area and if the individuals converged to the global area are continuously alive until one of the individuals approaches to the global optimum, then the GA will easily find the global optimum. Generally the converged areas, however, do not include the global optimum areas or even if they include global optimum areas, the individuals in the global optimum area are not alive until one of them reaches to the global optimum. Therefore, it is quite difficult for some converged individuals to local optimum areas to get out of the areas and approach to the global optimum areas. This is the reason why most optimization algorithms do not show good performances especially for multi-modal complex function optimization problems.

This problem called a premature convergence problem is a major problem to overcome for fast optimization. A lot of algorithms to solve this problem have been introduced to date [2, 4–9]. Most introduced methods strengthened the exploration, i.e., increasing the mutation probability in GAs. We devised the queen-bee evolution for the good convergence to the global optimum and strong mutation for the fast escaping of the local optimum areas [16]. Our queen-bee GAs showed very good performances in some good parameters of strong mutation rate and probability. However, there are no systematic methods to choose the good values of the strong mutation rate and probability. In this paper, we devised an adaptive control method of the strong mutation rate and probability.

We employed the measure of convergence for adaptive control of the strong mutation rate and probability. First, we used the average fitness for the measure of convergence. If the average fitness of current generation is less than or equal to that of previous generation, then we regard as a converged state and increase the strong mutation rate and probability. Otherwise, we decrease the strong mutation rate and probability. As a second measure of convergence, we took the failed ratio of evolution of individuals. That is, a specific percent of whole offsprings are failed to evolve from their parents, we also regard it as a converged state. The enforcing of strong mutation rate and probability allows the individuals to get out of the local optimum areas and to approach to the global optimum areas. If some individuals are succeed to escape the local optimum areas, then this makes the GAs escape a converged state and finally the strong mutation rate and probability will be decreased for stable evolution to the global optimum. From this adaptive control of the strong mutation rate and probability, the individuals of our GA can easily get out of the local optimum areas and can fast approach to the global optimum.

2.2 Proposed Genetic Algorithm

Algorithm 1 shows the proposed GA with adaptive control of the strong mutation rate and probability. Excepts for the operation of "do control ξ and $p_m^{'}$" marked by the asterisk, it is the same as the original queen-bee GA. The detailed control method of strong mutation rate and probability is shown from the line 11 to line 18.

Algorithm 1 Genetic algorithm with adaptive control method

// t : time //
// n : population size //
// P : populations //
// ξ : strong mutation rate//
// p_m : normal mutation probability //
// $p_m^{'}$: strong mutation probability //
// $m_c(t)$: measure of convergence at t //
// S_c : converged state //
// η : control scale factor, where $\eta >= 1$ //
// I_q : a queen-bee //
// I_m : selected bees //
1 t ← 0
2 initialize $P(t)$
3 evaluate $P(t)$
4 **while** (**not** termination-condition)
5 **do**
6 t ← t + 1
7 select $P(t)$ from $P(t-1)$
8 $P(t) = \{(I_q(t-1), I_m(t-1))\}$
9 recombine $P(t)$
10 do control ξ and $p_m^{'}$ (*)
11 calculate $m_c(t)$

```
12            if m_c(t) = S_c
13                ξ = ξ + δξ, where ξ ≤ 1
14                p'_m = p'_m + δp'_m, where p'_m ≤ 1
15            else
16                ξ = ξ − δξ/η, where ξ ≥ 0.5
17                p'_m = p'_m − δp'_m/η, where p'_m ≥ 0.05
18            end if
19        do crossover
20        do mutation
21            for i = 1 to n
22                if i ≤ (ξ × n)
23                    do mutation with p'_m
24                else
25                    do mutation with p_m
26                end if
27            end for
28    evaluate P(t)
29 end
```

We first briefly address the queen-bee GA and then explain the adaptive control method in detail.

Unlike the original GA, the parents of queen-bee GA are composed of two types of individuals, i.e., the selected parent I_m and the queen-bee I_q, where queen-bee is the best individual in the previous generation. Thus, every parents have one selected individual and the best individual as shown in the line 8 in Algorithm 1. This composition of parents makes the GA follow to the queen-bee direction (in other words, this increases the exploitation). This strong exploitation makes the GA not only fast evolve to the global optimum, but also fast fall into local optimum areas. In order to compensate this negative effect, we employed a strong mutation as shown in lines 21 $\sim$ 26. That is, a specific percent ξ of individuals are strongly mutated with strong mutation probability $p^{'}_m$. The strong mutation rate ξ and probability $p^{'}_m$ are the major factors of the performances of queen-bee GA. In [16], we chose some values for the strong mutation rate and probability and experimented with those values.

In this paper, the strong mutation rate and probability are dynamically controlled using the measure of convergence as shown in the lines 11 $\sim$ 18. We first calculate the measure of convergence $m_c(t)$ and control the strong mutation rate ξ and strong mutation probability $p^{'}_m$. If the state of current generation is a converged state, then the strong mutation rate and probability are increased as $\xi = \xi + \delta\xi$ and $p^{'}_m = p^{'}_m + \delta p^{'}_m$. Otherwise, they are decreased as $\xi = \xi - \delta\xi/\eta$ and $p^{'}_m = p^{'}_m - \delta p^{'}_m/\eta$, where η is a control scale factor. As the control scale factor is increased, the strong mutation effect will be large.

As described in previous section, we used two parameters for the measure of convergence. First is the average fitness and second is the failed ratio of evolution. More specifically, if the average fitness of current generation is less than that of previous generation, we regard as the converged state. Similarly, if a specific percent of offsprings are failed to evolved from their parents, we also regard as the converged state. Failed evolution means that the fitness f_o of an offspring o is less than or equal to the fitness of parents $f_p = (f_i + f_j)/2$, where f_i and f_j are the fitness of parents.

3. Experimental Results

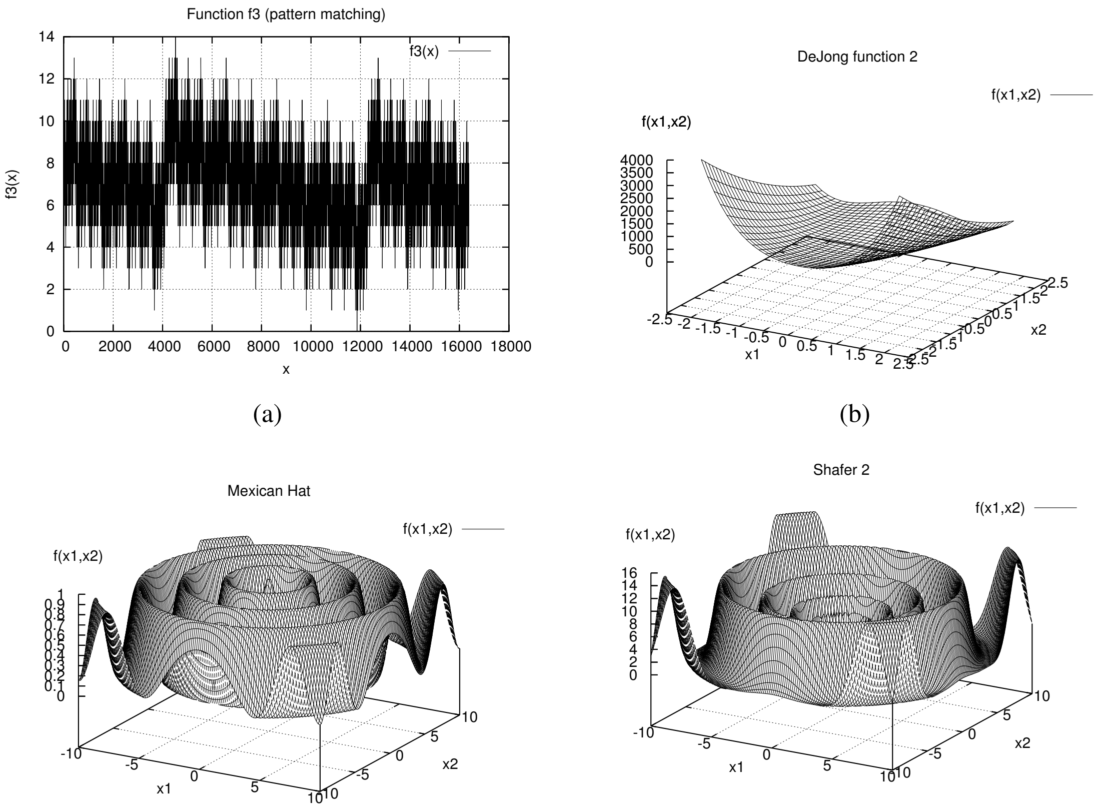
We experimented our algorithm with four function optimization problems. The four functions are given in Equation 1.

$$
\begin{aligned}
f_1 &= \textstyle\sum_{j=1}^{h} m_j \begin{cases} m_j = 1 & \text{if } T_j = I_j \\ m_j = 0 & \text{if } T_j \neq I_j \end{cases} \\
f_2 &= 100(x_1^2 - x_2)^2 + (1 - x_1)^2, \\
&\quad \text{where } -2.048 \leq x_i \leq 2.048 \\
f_3 &= 0.5 - \frac{sin(\sqrt{x_1^2+x_2^2})sin(\sqrt{x_1^2+x_2^2}) - 0.5}{(1.0+0.001(x_1^2+x_2^2))(1.0+0.001(x_1^2+x_2^2))}, \\
&\quad \text{where } -10 \leq x_i \leq -10 \\
f_4 &= (x_1^2 + x_2^2)^{0.25} sin^2(50(x_1^2 + x_2^2)^{0.1} + 1)
\end{aligned}
$$

Functions $f_1 \sim f_4$ are a bit pattern matching function, DeJong function 2, a Mexican hat function, and a Schafer function 2, respectively. Fig. 1 shows the input-output relations of four functions.

Function f_1 is a relatively simple, one dimensional bit pattern matching problem. Function f_2 has one global optimum area and one relatively large local optimum area. As a very complex multi-modal problem, function f_3 called a Mexican hat function is quite difficult to find the global optimum located at $(0, 0)$ because it has too many local optimum area around the global optimum. There are four global optimum areas in Schafer function 2 at four corners. Too many local optimum areas in the middle of the function prevent the individuals from approaching to the global optimum.

We tested with typical parameters as shown in Table 1. If one of the individuals finds the global optimum, then the generation number at that time is recorded. We measured the performances of all methods with an averaged value of 10 runs on the same parameter set. Table 2 showed one of experimental results using the average fitness as a measure of convergence and $\eta = 8$. For simplicity, we describe only average values without standard deviation values. In the table, QGA means original queen-bee GA and PGA denotes the queen-bee GA with proposed adaptive control method. As you can see in the table, the performances of original queen-bee GA are greatly changed according to the parameters of ξ and $p^{'}_m$. In proposed method, we don't need to choose the two parameters because they are adaptively controlled by our proposed method. We marked the best result of QGA with the asterisk (*). The performances of PGA are similar to the best result of QGA. Note that our method showed better results than the QGA in the most complex



(a) (b) (c) (d)

Fig. 1. Experimental functions (a) f_1 (b) f_2 (c) f_3 (d) f_4

Table 1. Parameters for experiments

Parameters	Values
Selection method	roulette wheel
Crossover probability (p_c)	0.6
Mutation probability (p_m)	0.05
Population size	10
Individual length	24 bits
Strong mutation rate	0, 0.2, 0.4, 0.6
Strong mutation probability	0.6, 0.8, 1.0
Control scale factor (η)	1, 2, 4, 8
Increment of strong mutation rate ($\delta\xi$)	0.005
Increment of strong mutation probability ($\delta p_m^{'}$)	0.025
Initial strong mutation rate for adaptive control	0
Initial strong mutation probability for adaptive control	0.05

Table 2. Experimental results of QGA and PGA

		f_1		f_2		f_3		f_4	
ξ	p'_m	QGA	PGA	QGA	PGA	QGA	PGA	QGA	PGA
0	-	(*) 58.4		1840432.7		17779.8		29603.1	
0.2	0.6	84.2		732.4		65066.5		26486.6	
	0.8	97.5		349.2		37867.0		37916.2	
	1.0	77.5		(*) 298.0		(*)3767.3		18000.0	
0.4	0.6	1115.3	475.1	1560.8	1774.5	355116.1	1454.8	175004.1	31216.7
	0.8	654.9		2354.5		295721.0		130466.9	
	1.0	351.0		962.6		4170.3		(*) 16472.4	
0.6	0.6	40039.6		50147.8		1368656.2		197116.1	
	0.8	56686.6		24768.4		549066.7		226828.3	
	1.0	6933.1		10347.2		7701.1		38822.7	

Table 3. Experimental results according to the control scale factor η

	average fitness				failed ratio			
η	f_1	f_2	f_3	f_4	f_1	f_2	f_3	f_4
1	(*) 51.6	(*)640.9	8155.5	57067.1	1040.7	1032.3	3570.1	22558.9
2	1270.0	1498.6	2765.4	20250.3	819.6	1209.5	5839.4	(*) 18094.6
4	611.3	1370.8	2047.7	(*) 14711.2	951.7	1075.8	3525.7	21428.3
8	475.1	1774.5	(*)1454.8	31216.7	(*) 661.8	(*)839.5	(*)2366.5	36429.4

and difficult function f_3. This indicates that our method is very useful and practical because we don't need the empirical selection of the parameters.

In order to show the performances according to the $\eta = 1,\ 2,\ 4,\ 8$, we experimented our method using the two measure of convergence: average fitness and failed ratio, respectively. Table 3 shows the performances of two measures of convergence with four control scale factors, $\eta = 1,\ 2,\ 4,\ 8$. The performances of large η are generally better than those of small η, but it is not always. The performances using average fitness as a measure of convergence is better than those using the failed ratio at most functions. However, the deviation of performances at average fitness is larger than that at failed ratio. Note that as the η is getting more larger, the performances of f_3 is more and more better at the average fitness.

As a final experiment, we measured the performances according to the five parameter sets: $(\delta\xi,\ \delta p'_m) = \{(0.005,\ 0.025),\ (0.01,\ 0.05),\ (0.02,\ 0.1),\ (0.04,\ 0.2),\ \ (0.08, 0.4)\}$ for average fitness and $r_e = \{0.1,\ 0.3,\ 0.5,\ 0.7,\ 0.9\}$ for the failed ratio of evolution. Table 4 shows the experimental results. The underlines of the numbers indicate the best results under various parameters on a same control scale factor and the asterisk of the numbers implies that the best results on whole scale factors. As you can see, all best results are located in the average fitness cases even if the difference of best results between average fitness and failed ratio is not so much. In the case of $r_e = 0.9$, the most performances are bad because it rarely occurs the 90 percents of individuals are failed to evolve. This results in preventing from increasing of strong mutation rate and probability. Finally, the strong mutation rate and probability will be remained very small values and this causes individuals not to get out of the local optimum areas.

4. Conclusion

In this paper, we introduced an adaptive control method of strong mutation rate and probability. This dynamic control of strong mutation rate and probability enables us not to do the empirical selection through too many experiments. In order to adaptively control the strong mutation rate and probability, we introduced the measure of convergence using the average fitness and failed ratio. With this measure of convergence if the GA falls into a converged state, then the strong mutation rate and probability are increased by the predetermined amount and otherwise, they are decreased. This helps the individuals which fall into local optimum areas get out of the areas and results in increasing the performances of the GA. From extensive experiments we could find that our method showed relatively similar results to the best one of empirical selection and sometimes better than the empirical selection. This addresses that our method is very effective and practical in that we can easily apply the queen-bee GA to the practical problems without

Table 4. Experimental results of various parameter sets

η	$(\delta\xi, \delta p_m^{'})$	average fitness				failed ratio				r_e
		f_1	f_2	f_3	f_4	f_1	f_2	f_3	f_4	
1	(0.005, 0.025)	(*)51.6	640.9	8155.5	57067.1	814.8	1232.5	4131.6	20457.3	0.1
	(0.01, 0.05)	59.3	995.9	13194.6	34773.5	814.8	1266.4	5823.7	32987.1	0.3
	(0.02, 0.1)	199.8	666.4	21234.7	36183.9	1040.7	1032.3	3570.1	22558.9	0.5
	(0.04, 0.2)	153.7	323.7	32043.0	33657.1	63.4	6498.0	13922.7	35206.2	0.7
	(0.08, 0.4)	68.9	(*)223.1	19524.0	35447.2	59.8	2073162.9	10034.2	40240.6	0.9
2	(0.005, 0.025)	1270.0	1498.6	2765.4	20250.3	814.8	1232.5	4131.6	20457.3	0.1
	(0.01, 0.05)	1239.6	2109.2	4904.3	12452.7	814.8	1266.4	2781.8	16935.2	0.3
	(0.02, 0.1)	811.7	1322.9	4214.0	19608.8	819.6	1209.5	5839.4	18094.6	0.5
	(0.04, 0.2)	991.8	1870.4	13220.4	25402.8	795.3	1602.8	3484.1	22910.3	0.7
	(0.08, 0.4)	384.1	2125.5	43269.8	25527.9	59.8	1955973.6	7792.9	104099.4	0.9
4	(0.005, 0.025)	611.3	1370.8	2047.7	14711.2	814.8	1232.5	4131.6	20457.3	0.1
	(0.01, 0.05)	1076.1	1105.1	2863.2	12854.1	814.8	1244.2	3055.4	25307.2	0.3
	(0.02, 0.1)	1305.1	2808.3	4136.7	21946.4	951.7	1075.8	3525.7	21428.3	0.5
	(0.04, 0.2)	1286.6	2739.9	3919.3	25129.6	405.5	2040.5	1455.5	25240.3	0.7
	(0.08, 0.4)	1199.7	1848.4	14753.6	36254.0	56.3	2452459.9	13974.4	60047.3	0.9
8	(0.005, 0.025)	475.1	1774.5	(*)1454.8	31216.7	814.8	1232.5	4131.6	20457.3	0.1
	(0.01, 0.05)	548.5	1486.7	3602.2	(*)9960.4	814.8	1232.5	4236.8	15524.7	0.3
	(0.02, 0.1)	718.0	1386.5	2222.7	25215.2	661.8	839.5	2366.5	36429.4	0.5
	(0.04, 0.2)	837.9	1213.1	3993.9	23641.6	407.2	1669.6	5035.4	21198.4	0.7
	(0.08, 0.4)	995.8	2880.8	3769.3	21538.1	56.3	2172136.9	15899.0	86514.5	0.9

worrying about the parameter selection from many trial and error experiments.

References

[1] D. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*. Reading, MA: Addison–Wesley, 1989.

[2] M. Srinivas and L. M. Patnaik, "Genetic Algorithms: A Survey," *IEEE Computer Magazine*, pp. 17–26, June 1994.

[3] D. Beasley, D. R. Bull, and R. R. Martin, "An Overview of Genetic Algorithms: Part 1, Fundamentals," Technical Report obtained from http://home.ifi.uio.no/∼jimtoer/GA_Overview1.pdf.

[4] R. Yang and I. Douglas, "Simple Genetic Algorithm with Local Tuning: Efficient Global Optimizing Technique," *Journal of Optimization Theory and Applications*, vol. 98, pp. 449–465, Aug. 1998.

[5] S. M. Libelli and P. Alba, "Adaptive mutation in genetic algorithms," *Soft Computing*, vol. 4, pp. 76–80, 2000.

[6] E. Alba and B. Dorronsoro, "The exploration/exploitation tradeoff in dynamic cellular genetic algorithms," *IEEE Transactions on Evolutionary Computation*, vol. 9, pp. 126–142, Apr. 2005.

[7] V. K. Koumousis and C. Katsaras, "A saw-tooth genetic algorithm combining the effects of variable population size and reinitialization to enhance performance," *IEEE Transactions on Evolutionary Computation*, vol. 10, pp. 19–28, Feb. 2006.

[8] A. E. Eiben, Z. Michalewicz, m. Schoenauer, and J. E. Smith, "Parameter Control in Evolutionary Algorithms," *Studies in Computational Intelligence*, vol. 54, pp. 19–46, 2007.

[9] Z. Jinhua, Z. Jian, D. Haifeng, and W. Sun'an, "Self-organizing genetic algorithm based tuning of PID controllers," *Information Sciences*, vol. 179, pp. 1007–1018, 2009.

[10] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, pp. 182–197, Apr. 2002.

[11] R. Marler and J. Arora, "Survey of multi-objective optimization methods for engineering," *Structural*

and Multidisciplinary Optimization, vol. 26, pp. 369–395, Apr. 2004.

[12] K. Anagnostopoulos and G. Mamanis, "Multiobjective evolutionary algorithms for complex portfolio optimization problems," *Computational Management Science*, vol. 8, pp. 259–279, Aug. 2011.

[13] J. C. Potts, T. D. Giddes, and S. B. Yadav, "The development and evaluation of an improved genetic algorithm based on migration and artificial selection," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 24, no. 1, pp. 73–86, 1994.

[14] J. Andre, P. Siarry, and T. Dognon, "An improvement of the standard genetic algorithm fighting premature convergence in continuous optimization," *Advances in engineering software*, vol. 32, no. 1, pp. 49–60, 2001.

[15] C. Xudong, Q. Jingen, N. Guangzheng, Y. Shiyou, and Z. Mingliu, "An Improved Genetic Algorithm for Global Optimization of Electromagnetic Problems," *IEEE Transactions on Magnetics*, vol. 37, pp. 3579–3583, Sept. 2001.

[16] S. H. Jung, "Queen-bee evolution for genetic algorithms," *Electronics Letters*, vol. 39, pp. 575–576, Mar. 2003.

Sung Hoon Jung
He received his B.S.E.E. degree from Hanyang University, Korea, in 1988 and M.S. and Ph.D. degrees from KAIST, in 1991 and 1995, respectively. He joined the Department of Information and Communication Engineering at the Hansung University in 1996, where he is a professor. His research interests are in the fields of intelligent systems, and in particular of application of neural networks, fuzzy logic, evolutionary computation algorithms to intelligent systems such as intelligent characters of computer games, computer generated forces on wargame, and so on. Recently, he started working on new research fields such as systems biology, bio-inspired engineering, brain engineering, and evolutionary complex systems. He is a member of the Korean Institute of Intelligent Systems (KIIS) and Institute of Electronics Engineers of Korea (IEEK).

E-mail : shjung@hansung.ac.kr