

## A COEVOLUTIONARY MULTI-OBJECTIVE EVOLUTIONARY ALGORITHM FOR GAME ARTIFICIAL INTELLIGENCE

TSE GUAN TAN  
JASON TEO  
KIM ON CHIN  
RAYNER ALFRED

### ABSTRACT

Recently, the growth of Artificial Intelligence (AI) has provided a set of effective techniques for designing computer-based controllers to perform various tasks autonomously in game area, specifically to produce intelligent optimal game controllers for playing video and computer games. This paper explores the use of the competitive fitness strategy: *K* Random Opponents (KRO) in a multiobjective approach for evolving Artificial Neural Networks (ANNs) that act as controllers for the Ms. Pac-man agent. The Pareto Archived Evolution Strategy (PAES) algorithm is used to generate a Pareto optimal set of ANNs that optimize the conflicting objectives of maximizing game scores and minimizing neural network complexity. Furthermore, an improved version, namely PAESNet\_KRO, is proposed, which incorporates in contrast to its predecessor KRO strategy. The results are compared with PAESNet. From the discussions, it is found that PAESNet\_KRO provides better solutions than PAESNet. The PAESNet\_KRO can evolve a set of nondominated solutions that cover the solutions of PAESNet.

*Keywords:* artificial neural networks, coevolutionary algorithms, evolutionary algorithms, game artificial intelligence, *K* random opponents, Ms. Pac-man, multiobjective evolutionary algorithms, Pareto archived evolution strategy

### INTRODUCTION

In recent years, there has been an increasing interest in bio-inspired computing (Mange & Tomassini, 1998; Sipper, 2002; Teo, 2003; Teuscher et al., 2003; De Castro & Von Zuben, 2005; Floreano & Mattiussi, 2008). It is a broad area encompassing disciplines such as evolutionary algorithms and artificial neural networks that transform biological ideas into computer operations and algorithms. Evolution and learning (Nolfi & Floreano, 1999) in computational intelligence are two mechanisms of bio-inspired algorithms to figure out the best and most effective solutions to problems arising from various science, engineering and financial fields in noisy, dynamic, complex environments. According to Nolfi and Floreano (1999), evolution is defined as “a form of adaptation capable of capturing relatively slow environmental changes that might encompass several generations”, while learning is defined as a process that “allows an individual to adapt to environmental changes that are unpredictable at the generational level”.

01: <i>gen</i> = 0	//Start with an initial time
02: Initial population <i>Pop(gen)</i>	//Randomly initial the population
03: Fitness evaluation <i>Pop(gen)</i>	//Evaluate initial population
04: WHILE Termination = False	//Examination for termination criterion
05: <i>gen</i> = <i>gen</i> + 1	//Increase the generation counter
06:     Parent selection <i>Pop(gen)</i>	//Perform parents selection
07:     Crossover <i>Pop(gen)</i>	//Recombine the “gene” of selected parents
08:     Mutation <i>Pop(gen)</i>	//Perturb the mated population stochastically
09:     Fitness evaluation <i>Pop(gen)</i>	//Evaluation individuals
10:     Survivor selection <i>Pop(gen)</i>	//Select the best individuals
11: END WHILE	

FIGURE 1. Evolutionary algorithms pseudocode

Evolutionary Algorithms (EAs) (Poli & Logan, 1996; Deb, 2001; Eiben & Smith, 2007; Maragathavalli, 2011) are used as a stochastic optimization method to search a set of promising solutions in complex problems, based on the basic principles of biological evolution such as selection, crossover and mutation operations as shown in Figure 1. Coevolutionary Algorithms (CAs) are one of the classes of EAs in which the individual (or population) fitness is depends on the interactions with other individuals (populations). There are two basic methods of CAs in the literature: competitive coevolution and cooperative coevolution (Coello Coello & Sierra, 2004). In competitive coevolution (Rosin & Belew, 1997), individual fitness is evaluated by competing with other individuals to survive in a series of competitions. However, in cooperative coevolution (Potter & De Jong, 1994), the individual fitness is determined by cooperating with other individuals to solve the problems.

Artificial Neural Networks (ANNs) (Haykin, 2009) are a learning paradigm inspired by the operation of the biological nervous systems, which functions analogously to the human brain. Traditionally, ANNs are trained using learning algorithms such as backpropagation (Rumelhart et al., 1986) to determine the optimal connection weights between nodes. However such methods are gradient-based techniques which tend to have two major drawbacks: slow learning speed and easily becoming trapped in local minima (Zhu et al., 2005; Burse et al., 2011) when attempting to optimize the connection weights. There is a large volume of published studies describing the role of EAs in ANNs. Evolutionary approaches have been proposed as an alternative method for optimizing the connection weights to overcome the issues described above. ANNs evolved through this method are thus referred to as Evolutionary ANNs (EANNs). In the literature, research into EANNs generally involves one of three approaches:

1. Evolving the weights of the network (Belew et al., 1990; Fogel et al., 1990).
2. Evolving the architecture (Miller et al., 1989; Kitano, 1990).
3. Evolving both simultaneously (Koza & Rice, 1991; Angeline et al., 1994; Teo & Abbass, 2004).

The primary objective of this study is to investigate the effects of multiobjective competitive coevolution for artificial neural network in dynamic and unpredictable video game environments. One of the well-known Multiobjective Evolutionary Algorithms (MOEAs) called Pareto Archived Evolution Strategy (PAES) is integrated with  $K$  Random Opponents (KRO) competitive fitness strategy in order to evolve both architecture and connection weights (including biases) of ANNs. With this, it hopes to show that it is able to autonomously play the commercial video game known as Ms. Pac-man. This game is an interesting, non-deterministic and challenging test-bed for evaluating machine as well as human intelligence (Lucas, 2005). Therefore it is an ideal benchmark to test and analyze whether computer-based controllers can play the game in an intelligent manner similar to that of a human playing the game.

## METHODS

This section is divided into three subsections to present and describe the PAES, the Pareto Archived Evolution Strategy Neural Network (PAESNet) and the integration of PAESNet with a competitive fitness strategy respectively.

### PARETO ARCHIVED EVOLUTION STRATEGY

Pareto Archived Evolution Strategy or PAES was first introduced by Knowles and Corne (1999), is one of the simplest yet effective MOEAs. The mutation operator plays a major role in this algorithm by altering the genes in each chromosome in the population, such as Cauchy

mutation, Gaussian mutation and so on. Additionally, PAES implements the elitism approach by preserving the best individuals from every generation, and an archive stores all the nondominated solutions along the Pareto front. A crowding method which works by recursively breaking down the objective space into  $d$ -dimensional grids is also introduced for diversity maintenance of the nondominated solutions in the archive. There are three different basic forms of PAES: (1+1)-PAES, (1+ $\lambda$ )-PAES and ( $\mu$ + $\lambda$ )-PAES (Knowles & Corne, 2000). The (1+1)-PAES generates a single offspring from a single parent through a mutation mechanism, and the offspring will then compete with the parent for survival. In the (1+ $\lambda$ )-PAES, a set of  $\lambda$  offspring is created from a single parent and the fittest individual is chosen among the  $\lambda$  offspring and the parent. In the ( $\mu$ + $\lambda$ )-PAES, a set of  $\lambda$  offspring is generated from  $\mu$  parents. The next generation consists of the  $\mu$  best individuals selected from the union of  $\mu$  parents and  $\lambda$  offspring. Overall, the (1+1)-PAES is becoming more popular as compared to other forms because of its simplicity, which has also been applied to serve as a baseline algorithm for handling multiobjective optimization problems.

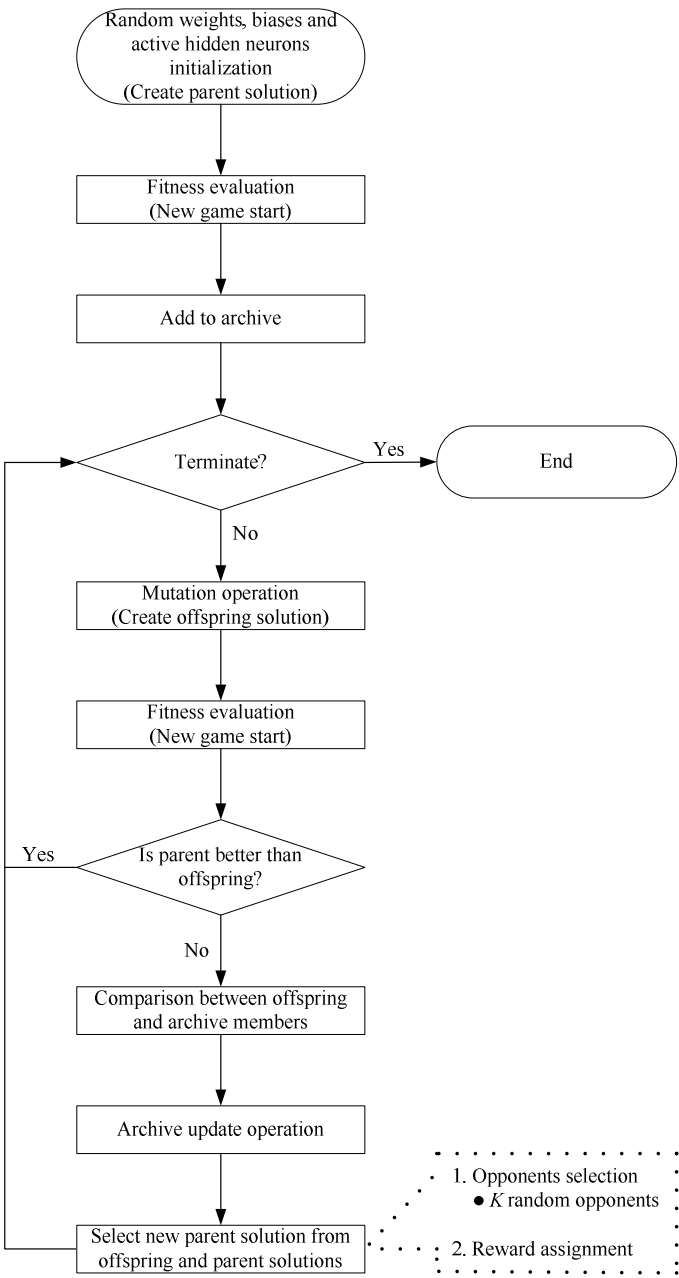


FIGURE 2. The flowchart of PAESNet / PAESNet\_KRO

Pareto Archived Evolution Strategy Neural Network or PAESNet is discussed. In this proposed system, two objectives are involved. The first objective,  $F_1$  is to maximize the game scores of Ms. Pac-man game as shown in Equation 1 whereas the second objective  $F_2$  is to minimize the number of hidden neurons in the feed-forward ANN as shown in Equation 2. The initial value of hidden neurons is set to 20. At the start of the initialization phase, the ANN weights, biases and active hidden neurons in hidden layer are encoded into a chromosome from uniform distribution with range [-1, 1] to act as parent and its fitness is evaluated. Subsequently, polynomial mutation operator is used with distribution index = 20.0 to create an offspring from the parent and its fitness is evaluated. After that, the fitness of the offspring and parent are compared. If the offspring performs better than the parent, then the parent is replaced by the offspring as a new parent for the next evaluation. Otherwise the offspring is eliminated and a new mutated offspring is generated. If the parent and the offspring are incomparable, the offspring is compared with set of previously nondominated individuals in the archive. The proposed algorithms are run 10 times with 5000 evaluations in each. Figure 2 shows the flowchart of PAESNet.

$$F_1 = \sum_{n=1}^N (\text{Ms. Pac - man Scores}) \tag{1}$$

$$F_2 = \sum_{i=1}^M h_i \tag{2}$$

where  $n$  and  $N$  represent the number of lives in a full game,  $M$  and  $h_i$  represent the number of hidden neurons in the feed-forward ANN.

Pareto archived evolution strategy neural network with  $K$  random opponents

In this subsection, one proposed competitive coevolution PAESNet: Pareto Archived Evolution Strategy Neural Network with  $K$  Random Opponents (PAESNet\_KRO) is presented for creating the Ms. Pac-man agent to solve two objective optimization problem. Basically, the framework of the PAESNet\_KRO model is similar to the PAESNet as shown in Figure 2. The main differences of PAESNet\_KRO in comparison to PAESNet are the two additional methods for parent selection process, opponents selection and reward assignment. The opponents selection method will select individuals as the opponents based on the KRO. The fitness of each individual is measured against  $K$  number of random opponents without self-play as shown in Figure 3. With this strategy, this method will randomly select opponents from the archive. The  $K$  is tested with the values of 2 in this study. After the opponents selection process, each individual will compete against the entire set of opponents. During the tournament, the reward value will be calculated for each competition by the reward function as shown in Equation 3. Each reward value will be summed up as the fitness score for the individual using the reward assignment method. The individual with highest fitness score is selected as the next parent and the iteration continues. The predefined maximum number of evaluations serves as the termination criterion of the loop. In this study, the number of runs is set to 10 and each run is tested 5000 evaluations consecutively.

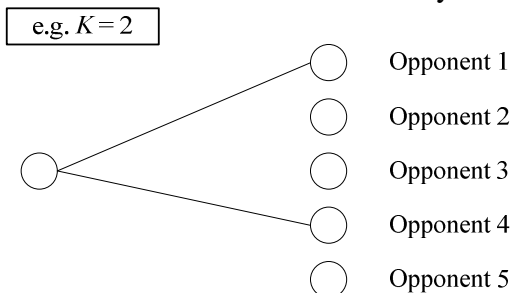


FIGURE 3. KRO strategy

The description of the reward function is as Equation 3.  $I$  represent the participating individual, while  $O$  represents the opponent.  $R$  is the raw fitness value,  $\max(R)$  is the maximum raw fitness value and the  $\min(R)$  is the minimum raw fitness value. The range for values in this function is within  $[-1, 1]$ . If  $Reward(I, O) = 0$ , it corresponds to the competition being a draw.

$$Reward(I, O) = \frac{R(O) - R(I)}{\max(R) - \min(R)} \quad (3)$$

## PERFORMANCE METRIC FOR MOEAS

Coverage ( $C$ ) metric is used for comparing the dominance relationship between two Pareto fronts. As stated in (Zitzler, 2000), the formal definition follows.

- Let  $P_1, P_2 \subseteq P$  be two sets of nondominated solutions.
- The function  $C$  maps the ordered pair  $(P_1, P_2)$  to the interval  $[0, 1]$ :

$$C(P_1, P_2) = \frac{|\{u_2 \in P_2; \exists u_1 \in P_1 : u_1 \underline{\phi} u_2\}|}{|P_2|} \quad (4)$$

$u_1 \underline{\phi} u_2$  if  $u_1$  dominates  $u_2$  or  $u_1$  equal to  $u_2$ . If the value  $C(P_1, P_2) = 1$  means that all the solutions in  $P_2$  are dominated by  $P_1$ . Otherwise, if value  $C(P_1, P_2) = 0$  represents the situation when none of the points in  $P_2$  are dominated by  $P_1$ . In addition, if  $C(P_1, P_2)$  is higher than  $C(P_2, P_1)$ , then  $P_1$  is better than  $P_2$ . Figure 4 shows the graphical presentations for coverage metric. The scale is 0 (no coverage) at the bottom and 1 (total coverage) at the top per rectangle.

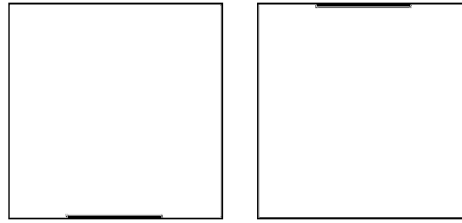


FIGURE 4.  $C(P_1, P_2) = 0$  and  $C(P_1, P_2) = 1$

## EXPERIMENTAL RESULTS AND DISCUSSIONS

Table 1 shows the experimental results of best scores over 5000 evaluations in 10 runs. A paired-samples  $t$ -test was conducted to ascertain whether there was a significant difference between scores on PAESNet and PAESNet\_KRO. There was a significant difference in the scores for PAESNet\_KRO ( $M = 19617$ ,  $SD = 2182.5523$ ) and PAESNet ( $M = 14795$ ,  $SD = 2024.4629$ );  $t(9) = -4.7987$ ,  $p = 0.0010$ ,  $p < 0.05$  (two-tail) as shown in Table 2. These results suggest that coevolutionary approach really does have an effect on the quality of PAESNet\_KRO.

TABLE 1. The best game scores over 5000 evaluations in 10 runs

Run	PAESNet	PAESNet_KRO
1	14930	15930
2	13550	18870
3	14020	21850
4	14920	19540
5	20130	19130
6	15060	17630
7	15020	23680
8	13880	19600
9	12820	21150
10	13620	18790

TABLE 2. *t*-test (paired two sample for means)

	PAESNet	PAESNet_KRO
Mean (M)	14795	19617
Standard Deviation (SD)	2024.4629	2182.5523
Variance	4098450.0000	4763534.4440
df	9	
<b><i>t</i> Stat</b>	<b>-4.7987</b>	
<b><i>P</i>(<math>T \leq t</math>) two-tail</b>	<b>0.0010</b>	
<i>t</i> Critical two-tail	2.2622	

Additionally, the coverage metric is used to compare the significance of the dominance relationship between two sets of nondominated solutions. From the data in Table 3, it is apparent that the nondominated solutions obtained by PAESNet are clearly dominated by the nondominated solutions obtained by PAESNet\_KRO. The global Pareto fronts for the PAESNet\_KRO and PAESNet are shown in Figure 5. The average value of PAESNet dominated by PAESNet\_KRO is 93%. On the other side, the average value of PAESNet\_KRO dominated by PAESNet is only 2%. It is interesting to note that almost all the coverage values of  $C(\text{PAESNet}, \text{PAESNet\_KRO})$  are equal to 0. These results indicate that none solution found by the PAESNet\_KRO is dominated by any solution found by the original PAESNet. While, majority values of  $C(\text{PAESNet\_KRO}, \text{PAESNet})$  are equal to 1 mean that all solutions in PAESNet are dominated by PAESNet\_KRO. Here, boxplots as shown in Figure 6 are used to visualize the distribution of these samples. As can be seen from the chart, the  $C(\text{PAESNet\_KRO}, \text{PAESNet})$  reported significantly more median than the  $C(\text{PAESNet}, \text{PAESNet\_KRO})$ . Overall, the results show that PAESNet\_KRO is capable to solve the multiobjective problem in dynamic game environments and achieve better nondominated solutions. A possible explanation for this might be that KRO strategy is more effectively to select the best nondominated solutions from the archive as the parent in order to create offspring for next generation.

TABLE 3. Coverage of nondominated solution sets resulting of the PAESNet versus PAESNet\_KRO

Run	$C(\text{PAESNet\_KRO}, \text{PAESNet})$	$C(\text{PAESNet}, \text{PAESNet\_KRO})$
1	1.00	0.00
2	1.00	0.00
3	0.80	0.00
4	1.00	0.00
5	0.83	0.00
6	0.86	0.17
7	1.00	0.00
8	1.00	0.00
9	1.00	0.00
10	0.83	0.00
Mean	0.93	0.02

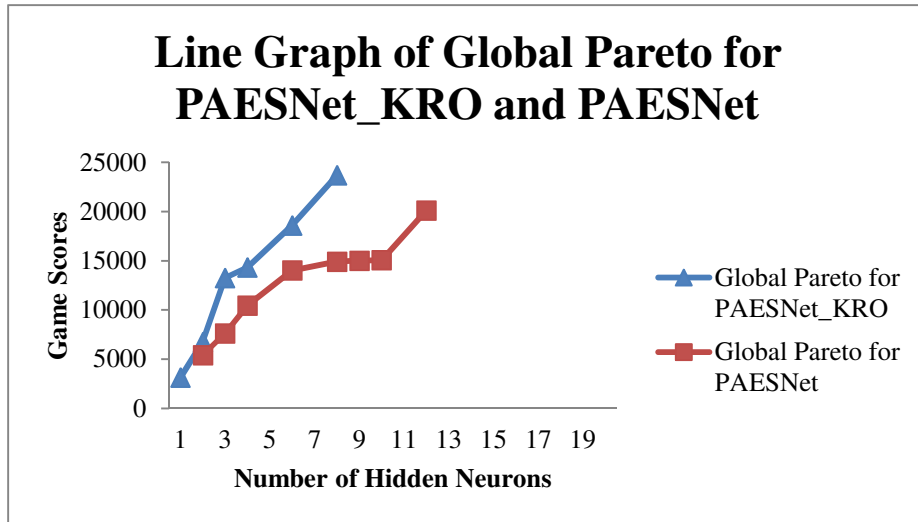


FIGURE 5. Line graph of global Pareto for PAESNet\_KRO and PAESNet

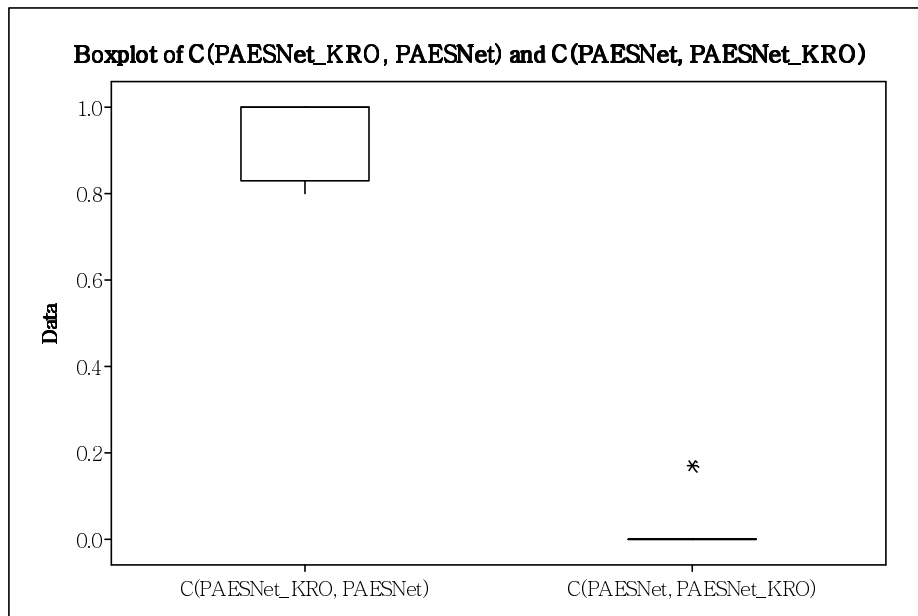


FIGURE 6. Boxplot of  $C(\text{PAESNet\_KRO}, \text{PAESNet})$  and  $C(\text{PAESNet}, \text{PAESNet\_KRO})$

## CONCLUSIONS

PAESNet\_KRO is presented, an improved elitist multiobjective evolutionary algorithm that employs competitive coevolutionary approach compared to its predecessor PAESNet. In this paper, two comparisons of PAESNet\_KRO with PAESNet have been carried out via Ms. Pac-man game domain. The key results of the comparison are (1) PAESNet\_KRO performs better than its predecessor PAESNet in controlling the behaviour of Ms. Pac-man agent to play the game autonomously and (2) the measure coverage indicates clear advantages of PAESNet\_KRO over PAESNet. In conclusion, the coevolutionary method has proven to be effective in improving the performance of multiobjective optimizer.

## REFERENCES

- Angeline, P.J., Saunders, G.M., and Pollack, J.B. 1994. An evolutionary algorithm that constructs recurrent neural networks. *IEEE Transactions on Neural Networks*, 5(1): 54-65.
- Belew, R.K., McInerney, J., and Schraudolph, N.N. 1990. Evolving networks: using the genetic algorithm with connectionist learning. Technical Report CS90-174, Department of Computer Science and Engineering, University of California.
- Burse, K., Manoria, M., and Kirar, V.P.S. 2011. Improved back propagation algorithm to avoid local minima in multiplicative neuron model. *Proceeding of the International Conference on Advances in Information Technology and Mobile Communication*. Nagpur: Springer, 67-73.
- Coello Coello, C.A., and Sierra, M.R. 2004. A study of the parallelization of a coevolutionary multi-objective evolutionary algorithm. *Proceeding of the Third Mexican International Conference on Artificial Intelligence*. Mexico City: Springer, 688-697.
- De Castro, L.N., and Von Zuben, F.J. 2005. *Recent developments in biologically inspired computing*. Pennsylvania: Idea Group Publishing.
- Deb, K. 2001. *Multi-objective optimization using evolutionary algorithms*. New York: John Wiley and Sons.
- Eiben, A.E., and Smith, J.E. 2007. *Introduction to evolutionary computing*. Berlin: Springer.
- Floreano, D., and Mattiussi, C. 2008. *Bio-inspired artificial intelligence: theories, methods, and technologies*. Massachusetts: MIT Press.
- Fogel, D.B., Fogel, L.J., and Porto, V.W. 1990. Evolving neural networks. *Biological Cybernetics*, 63(6): 487-493.
- Haykin, S. 2009. *Neural networks and learning machines*. New Jersey: Prentice Hall.
- Kitano, H. 1990. Designing neural networks using genetic algorithms with graph generation system. *Complex Systems*, 4(4): 461-476.
- Knowles, J.D., and Corne, D.W. 1999. The Pareto archived evolution strategy: a new baseline algorithm for Pareto multiobjective optimization. *Proceeding of the Congress on Evolutionary Computation*. Washington, D.C: IEEE, 98-105.
- Knowles, J.D., and Corne, D.W. 2000. Approximating the nondominated front using the Pareto archived evolution strategy. *Evolutionary Computation*, 8(2): 149-172.
- Koza, J.R., and Rice, J.P. 1991. Genetic generation of both the weights and architecture for a neural network. *Proceeding of the International Joint Conference on Neural Networks*. Seattle: IEEE, 397-404.
- Lucas, S.M. 2005. Evolving a neural network location evaluator to play Ms. Pac-man. *Proceeding of the IEEE Symposium on Computational Intelligence and Games*. Essex: IEEE, 203-210.
- Mange, D., and Tomassini, M. 1998. *Bio-inspired computing machines: towards novel computational architectures*. Lausanne: Presses Polytechniques et Universitaires Romandes.
- Maragathavalli, P. 2011. Search-based software test data generation using evolutionary computation. *International Journal of Computer Science and Information Technology*, 3(1): 213- 223.
- Miller, G.F., Todd, P.M., and Hegde, S.U. 1989. Designing neural networks using genetic algorithms. *Proceeding of the Third International Conference on Genetic Algorithms*. Virginia: Morgan Kaufmann, 379-384.
- Nolfi, S., and Floreano, D. 1999. Learning and evolution. *Autonomous Robots*, 7(1): 89-113.
- Poli, R., and Logan, B.S. 1996. On the relations between search and evolutionary algorithms. Technical Report CSRP-9607, School of Computer Science, University of Birmingham.
- Potter, M.A., and De Jong, K.A. 1994. A cooperative coevolutionary approach to function optimization. *Proceeding of the Third Conference on Parallel Problem Solving from Nature*. Jerusalem: Springer, 249-257.
- Rosin, C.D., and Belew, R.K. 1997. New methods for competitive coevolution. *Evolutionary Computation*, 5(1): 1-29.
- Rumelhart, D.E., Hinton, G.E., and Williams, R.J. 1986. Learning internal representations by error propagation. In Rumelhart, D.E., and McClelland, J.L. (eds.) *Parallel distributed processing*, 318-362. Massachusetts: MIT Press.
- Sipper, M. 2002. *Machine nature: the coming age of bio-inspired computing*. New York: McGraw-Hill.



- Teo, J. 2003. Pareto multi-objective evolution of legged embodied organisms. PhD thesis, University of New South Wales.
- Teo, J., and Abbass, H.A. 2004. Automatic generation of controllers for embodied legged organisms: a Pareto evolutionary multi-objective approach. *Evolutionary Computation*, 12(3): 355-394.
- Teuscher, C., Mange, D., Stauffer, A., and Tempesti, G. 2003. Bio-inspired computing tissues: towards machines that evolve, grow and learn. *Biosystems*, 68(2-3): 235-244.
- Zhu, Q.Y., Qin, A.K., Suganthan, P.N., and Huang, G.B. 2005. Evolutionary extreme learning machine. *Pattern Recognition*, 38(10): 1759-1763.
- Zitzler, E., Deb, K., and Thiele, L. 2000. Comparison of multiobjective evolutionary algorithms: empirical results. *Evolutionary Computation*, 8(2): 173-195.

## BIOGRAPHY

1. Tan Tse Guan was born in Kuala Lumpur, Malaysia. He received the bachelor of computer science (software engineering) and master degree in artificial intelligence from Universiti Malaysia Sabah, Malaysia, in 2006 and 2008. He is working toward the PhD degree in computer science at the Universiti Malaysia Sabah, Malaysia.
2. Jason Teo is a senior lecturer in computer science at the School of Engineering and Information Technology and the deputy director of the Centre for Artificial Intelligence, Universiti Malaysia Sabah. He received his doctorate in information technology from the University of New South Wales, Australia, researching Pareto artificial evolution of virtual legged organisms. He has over 70 publications in the areas of artificial life, evolutionary robotics, evolutionary computing and swarm intelligence. His current research interests focus on the theory and applications of evolutionary multi-objective optimization algorithms in co-evolution, robotics and metaheuristics.
3. Chin Kim On research interests included gaming AI, evolutionary computing, evolutionary robotics, neural networks, image processing, semantics technology, agent technology, evolutionary data mining and biometric security system with mainly focused on fingerprint and voice recognition. He has more than 60 articles in the forms of journals, book chapters and conference proceedings. He is a member of IEEE and IAENG.
4. Rayner Alfred leads and defines projects around knowledge discovery and information retrieval at Universiti Malaysia Sabah. Rayner's works address current advanced intelligent techniques to model and optimise the complex and dynamic processes of knowledge discovery and information retrieval for structured/unstructured data. He holds a PhD degree in Computer Science from York University (UK), a Master degree in Computer Science from Western Michigan University (USA) and a Computer Science degree from Polytechnic University of Brooklyn, New York (USA). He has authored and co-authored more than 75 journals/book chapters, conference papers, and served on the organising committees of numerous international conferences and workshops.

Tse Guan Tan,

Jason Teo,

Kim On Chin,

Rayner Alfred.

Evolutionary Computing Laboratory,

School of Engineering and Information Technology,

Universiti Malaysia Sabah,

88400 Kota Kinabalu, Sabah, Malaysia.

tseguantan@gmail.com, jtwteo@ums.edu.my, kimonchin@ums.edu.my, alfred@ums.edu.my