A STUDY ON LAMARCKIAN AND BALDWINIAN LEARNING ON NOISY AND NOISELESS LANDSCAPES

Cheng Wai Kheng
University of Nottingham, Malaysia Campus
School of Computer Science
Jalan Broga 43500 Semenyih, Selangor, Malysia.
E-mail: cheng-wai.kheng@nottingham.edu.my

Meng Hiot Lim
Nanyang Technological University
School of Computer Engineering
Block N4, 2b-39, Nanyang Avenue, Singapore 639798
E-mail: emhlim@ntu.edu.sg

Siang Yew Chong
ASAP Research Group
University of Nottingham, Malaysia Campus
School of Computer Science
Jalan Broga 43500 Semenyih, Selangor, Malysia.
E-mail: siang-yew.chong@nottingham.edu.my

KEYWORD
Optimisation, Landscape, Memetic Computing, Baldwinian effect, Lamarckian evolution.

ABSTRACT

Lamarckian evolution and Baldwinian effect are two typical replacement schemes of the individual learning process in Memetic Computing. In this paper, we perform a comprehensive study on the behaviour of Lamarckian evolution and Baldwinian effect in noisy and noiseless continuous optimisation problems. The output of this comprehensive study shows that Lamarckian lifetime learning performs better on noiseless problem, while Baldwinian learning performs better on noisy problems.

INTRODUCTION

Memetic Computing (MC) represents one of the prominent emerging approaches of Computational Intelligence introduced in the last decades. Within the field, one of the most successful methodologies is the memetic algorithm (MA) that has materialised in the form of hybrid evolutionary search.

Typically, MA involves some lifetime learning within the Darwinian Evolution Cycle. MA may involve single or multiple lifetime learning procedures (Ong and Keane 2004). (Krasnogor and Smith 2005). When designing MA, several issues are often considered, from how much computational budget to allocate for lifetime learning at individual and generation level, which individuals of the population that should undergo lifetime learning to what form of lifetime learning methodology to employ (Nguyen, Ong et al. 2009). Among the design issues, one that has received little attention over the years is the type of learning scheme to use, i.e., Lamarckian or Baldwinian learning. In black box optimisation, the underlying problem characteristics are generally unknown beforehand. Hence, it is well established today according to the "No Free Lunch" theorem that no single algorithm, including, i.e., Lamarckian or Baldwinian learning, has clear advantage over another on all classes of problems except random walk (Wolpert and Macready 1997). In fact, the performance of MA is highly dependent on the problem landscape. Information regarding MA and connectivity analysis can be found in (M. N. Le, Y.S. Ong et al. 2009).

The motivation of this paper is to study the effect of Lamarckian and Baldwinian learning in MA for solving noisy and noiseless optimisation problems. We first analyse the behaviour of Lamarckian learning in the context of line search, which leads to the hypothesis that Lamarckian performs poorly in noisy problem, and then validate the hypothesis with simulation results.

The remaining paper is organised as followed. The paper starts with problem definition. Next, brief definitions of Lamarckian and Baldwinian learning and their comparisons are presented. Subsequently, the definition of Multiple Evaluation Method in the context of Baldwinian learning and an analysis of Lamarckian learning in the context of non-linear programming, followed by some empirical results and discussions, and finally ends with conclusion and future work.

LITERATURE REVIEW

A. Problem Definition

Stochastic global search such as evolutionary algorithm (De-Jong 2006) has been used in solving non-linear programming problem for several decades, which takes the form of:

$$\begin{aligned}&\arg\min_{\mathbf{x}} f : \mathbf{x} \to \mathbf{R}, \mathbf{x} \in \mathbf{R}^d \\ &\text{subjected } \mathbf{low} \le \mathbf{x} \le \mathbf{high}\end{aligned} \quad (1)$$

Proceedings 24th European Conference on Modelling and Simulation ©ECMS Andrzej Bargiela, Sayed Azam Ali David Crowley, Eugène J.H. Kerckhoffs (Editors)
ISBN: 978-0-9564944-0-5 / ISBN: 978-0-9564944-1-2 (CD)

where **x** is a parameter vector with size *d* and each variable in the **x** is restricted within the range of **low** and **high**.

In this study, the noise introduced during the optimisation process itself. This situation can arise when vector **x** refers to an output of devices, and the optimisation is real-time. Noise may be introduced during the transmission between the device and the computer, loss of precision when converting from analogue to digital signals, or even the inconsistency of the device itself. Such noises are usually assumed to be normally distributed.

We simulated this noise in standard benchmark functions through perturbations of the input vector **x**. Optimisation problem in Eqn. (1) is then transformed into:

$$\begin{aligned} &\min f(\mathbf{x}+\boldsymbol{\delta}) \\ &\text{subjected } \mathbf{low} \le \mathbf{x} \le \mathbf{high} \\ &\delta^{(i)} \in \gamma, \boldsymbol{\delta} = \begin{bmatrix} \delta^{(1)} \\ \vdots \\ \delta^{(d)} \end{bmatrix} \end{aligned} \tag{2}$$

δ_i is generated randomly according to certain density function γ. In this paper, Gaussian density function is used.

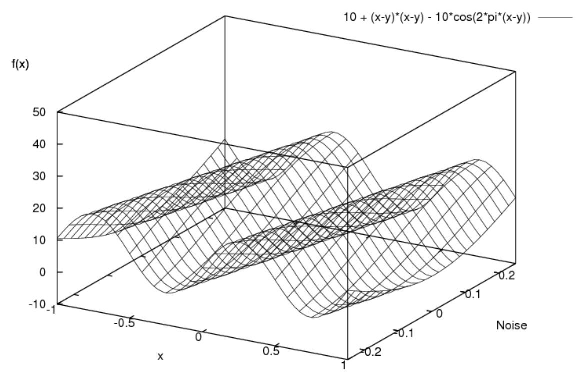

Fig 1: Noisy Rastrigin

Such perturbations transform the underlying problem landscape from static to dynamic. As an evaluation, the same input vector **x** generates different responses from object function *f* when it is to be evaluated several times. In this paper, we refer Eqn. (1) as noiseless problem and Eqn. (2) as noisy problem. Figure 1 demonstrates the noise effect shifting the landscape with noise range between -0.2 to +0.2.

B. Lamarckian and Baldwinian Lifetime Learning

Lamarckian lifetime learning involves the solution (both vector **x** and fitness) to be replaced during the individual learning process whereas Baldwinian replaces only the fitness of the solution instead of vector **x**. Let y be the response of **x**, $y = f(\mathbf{x})$. A solution space is then defined as an unordered pair of solution and its response.

$$\mathrm{S} = \mathrm{XY} = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2)....(\mathbf{x}_n, y_n)\} \tag{3}$$

The individual improvement process defines the neighbouring structure, X^{N}.

$$\begin{aligned} &\mathrm{X}^N = N(\mathbf{x}) = \{\mathbf{x}_1, \mathbf{x}_2 ...., \mathbf{x}_p \mid \\ &\forall_{i \le p} \mathbf{x}_i = \mathbf{x} + \boldsymbol{\delta}_i\} \end{aligned} \tag{4}$$

In general, the neighbourhood can be defined by any random generator with neighbourhood size σ, or special neighbourhood that is bound to specific function.

Subsequently, the replacement policies (with loss of generality, assuming a minimisation problem) proceeds as follows:

Lamarckian replacement policy:

$$\begin{aligned} &L(\mathbf{x}, y) \to (\mathbf{a}, f(\mathbf{a})), \{\mathbf{a}, \mathbf{a'} \in \mathrm{X}^N \mid \\ &f(\mathbf{a}) < f(\mathbf{a'}), \mathbf{a} \ne \mathbf{a'}\} \end{aligned} \tag{5}$$

Baldwinian replacement policy:

$$\begin{aligned} &B(\mathbf{x}, y) \to (\mathbf{x}, f(\mathbf{a})), \{\mathbf{a}, \mathbf{a'} \in \mathrm{X}^N \mid \\ &f(\mathbf{a}) < f(\mathbf{a'}), \mathbf{a} \ne \mathbf{a'}\} \end{aligned} \tag{6}$$

Note that the fitness function is not changed. Baldwinian lifetime learning only alters the function responses.

To date, numerous studies have been conducted on Baldwinian effect and Lamarckian evolution. (Julstrom 1999) studied the effect of applying different frequencies of Baldwinian or Lamarckian on a four-cycle combinatorial optimisation problem. Their empirical results showed that Baldwinian performed worst on average compared to pure Darwinian and Lamarckian. Mayley discussed on the same issue in (Mayley, G. 1996) and commented that if the solution in local improvement cannot be found by genetic operation within certain steps, the Baldwinian effect may mislead the search.

While the above discussed on the drawbacks of Baldwinian learning, Whitley et. al suggested that Baldwinian learning brings about the advantage of escaping from local optima since Lamarckian may possess the drawback of leading to premature convergence (Whitley, Gordon et al. 1994). However, Baldwinian learning generally converged slower and poorer results compared to Lamarckian. Castillo et al studied the same approach in Evolutionary Neural

Networks and concurred Whitley's finding (Castillo, Arenas et al. 2006).

METHODOLOGY

Although there have been many comparisons done between Lamarckian and Baldwinian, they are limited to problem in the absence of noise. The motivation of this paper is to extend the investigation of Baldwinian effect and Lamarckian evolution in both noisy and noiseless landscape.

A. Multiple Evaluation Method as Baldwinian learning

When an underlying optimisation problem is plagued with noise, the use of resampling in the search like the Multiple Evaluation Method (MEM) (Tsutsui and Ghosh 1997) (Tsutsui and Ghosh 1997) may be useful, which takes form as followed:

$$B(\mathbf{x}, y) \rightarrow (\mathbf{x}, \mu), \{\mathbf{x}_i \in \mathrm{X}^N \mid \mu = \frac{1}{p} \sum_{\mathbf{x}_i \in \mathrm{X}^N} f(\mathbf{x}_i)\} \quad (7)$$

Since the process only replaces the fitness with the mean, which in general, qualifies as a form of Baldwinian lifetime learning. Instead of taking the average, there are other forms of MEM, such as taking the worst fitness in the neighbouring solutions. More information regarding MEM can be found in (Y. S. Ong, P. B. Nair et al. 2006)

B. Analysis of Lamarckian in Noisy Optimisation Problem

Deterministic individual learning scheme such as Quasi-Newton, Gradient Descent, etc. that depends on the gradient information as guidance would be misled by the noise, especially when Lamarckian learning is used. When noise is introduced to the landscape, Lamarckian spends the resources trying to adapt to the noise instead of the actual landscape, which would cause the algorithm to converge slower for non convex problem. Nevertheless, Lamarckian should always converge on better results if the global stochastic search manages to bring the solution to the global optimum basin. After all, the basin that consists of the global optimum itself is a convex problem.

On the other hand, by countering the noise with re-sampling, Baldwinian replacement policy gains advantage by having fitness value closer to the actual function. This increases the selection pressure on "good" parents while teaming with fitness proportional selection mechanism, such as stochastic universal sampling.

Without hesitation, the following section analyses how noise could mislead the heuristics and reduce the efficiency of Lamarckian learning in noisy multimodal optimisation problem with a series of simple respond graphs.

A line search is chose as a deterministic search in this paper because it is the core component in most of the gradient-based search, thus, this would be a solid starting point to investigate. With the assumption of non-repeating space travelling, the line search takes form of:

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k \quad (8)$$

$\mathbf{x}_{k+1}$, the next solution to be evaluated, is determined by the current solution $\mathbf{x}_k$, with the increment of constant value α_k according to the searching direction $\mathbf{d}_k$.

$$\mathbf{d}_k = \begin{bmatrix} \frac{f(\mathbf{z}_1)}{\varepsilon} \\ \frac{f(\mathbf{z}_2)}{\varepsilon} \\ \vdots \\ \frac{f(\mathbf{z}_d)}{\varepsilon} \end{bmatrix}. \quad (9)$$

Let $z_m^{(j)}$ indicates the j^{th} dimension of the vector $\mathbf{z}$, and $x_k^{(j)}$ indicates the j^{th} dimension of vector $\mathbf{x}$ at step k. $z_m^{(j)}$ is defined as:

$$z_m^{(j)} = \begin{Bmatrix} x_k^{(j)} + \varepsilon \text{ iff } j = m \\ x_k^{(j)} \text{ otherwise} \end{Bmatrix} \quad (10)$$
$$\text{subjected } 0 < m \le d$$

It is possible that the noise changes the gradient information. As the effect, the searching direction is not determined by the gradient of $\mathbf{x}_k$, instead, it is determined by $\mathbf{x}_k + \boldsymbol{\delta}_k$, where each dimension of $\boldsymbol{\delta}_k$ is drawn independently according to a density function. $z_m^{(j)}$ are then transformed to:

$$z_m^{(j)} = \begin{Bmatrix} x_k^{(j)} + \delta_k^{(j)} + \varepsilon \text{ iff } j = m \\ x_k^{(j)} + \delta_k^{(j)} \text{ otherwise} \end{Bmatrix} \quad (11)$$
$$\text{subjected } 0 < m \le d$$

For the sake of simplicity, we illustrate the misleading searching direction by assuming every respond is drawn with a single δ noise value in a single dimension problem. Figure 2 illustrates the searching direction has been reversed by the noise. Point A and B are the same solution vector where A is evaluated with the absent of noise, and B with the present of noise. Suppose the searching starts with point A. The gradient information is decreasing towards the right. Given the same solution vector, the noise shows the gradient is decreasing towards left.

Second, assuming the noise is large and is divisible by α, it is possible that a line search starts with position that is m steps before the current solution $\mathbf{x}_k$, providing that $\mathbf{x}_{k-m}=\mathbf{x}_k+\boldsymbol{\delta}_k$, which violates the assumption of non-repeating space travelling.

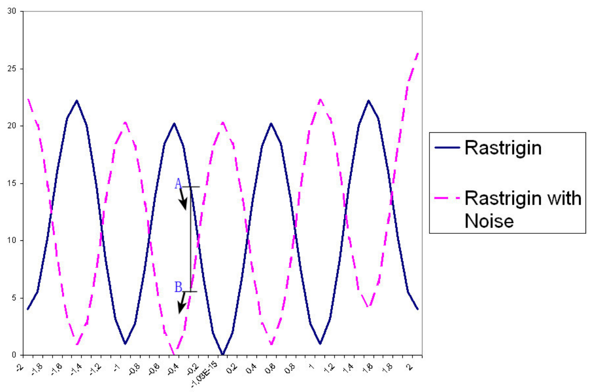

Fig 2: Noise in Searching Direction Determination

Lastly is the live lock effect, which causes the line search to oscillate between the same local optimum that causes by the noise. The term live lock is introduced in concept of concurrency, where two processes are given resources without making any progress. However, this effect happens when the noise is periodic only. In Figure 3, point A is a local maximum that moves decreasingly to reach local minimum point B in step k. In k+1, the respond of B is raised up to point C with the effect of noise. The line search moves decreasingly from point C to point D, which in fact, is the local maximum point A when the noise is removed.

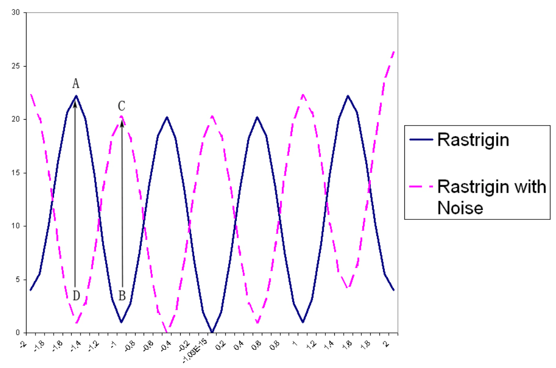

Fig 3: Line search oscillating between local optima, and the same local optimum causes by noise.

EMPIRICAL RESULTS AND DISCUSSIONS

We investigated the above analysis by running simulations on two sets of benchmark functions. First set is the noiseless function. Second set is the noisy functions with 0.001 noise level. The experimental setting is given in Table 1. Table 2 outlines the benchmark functions we use.

The choice of Lamarckian lifetime improvement budget is based on the size of k in Eqn. (8). Broyden-Flecher-Goldfarb-Shanno (BFGS) takes two gradient evaluations in every increment of k. Thus, the budget b formula is given as:

$$b = k[(2\times d)+l] \quad (12)$$

where d is the problem dimension and l is a variable to determine step size α.

Table 1: Experimental Parameters

Global Search: Real Coded Genetic Algorithm
Lamarckian Property: Local Improvement Method: Broyden–Fletcher–Goldfarb–Shanno (BFGS) Lifetime Improvement Frequency: 0.1 Lifetime Improvement Budget (FFE): 130
Baldwinian Property: Local Improvement Method: Multiple Evaluation Method stated in Eqn. (7) Lifetime Improvement Frequency: 1.0 Lifetime Improvement Budget (FFE): 13
Crossover Rate: 1.0
Mutation Rate: 0.03
Mutation Type: Gaussian Mutation with 1.0 Radius
Selection: Stochastic Universal Sampling
Replacement: Ranking
Termination Condition: 300,000 Function Evaluations
Population Size: 50

For the fair comparison, summation of function evaluation on each generation is equal for both approaches. The noise level σ is set to 0.001 * (function upperbound - function lowerbound). Each problem is repeated for 30 independent runs. The mean best over function calls are recorded in Figures 4 and 5.

The simulation results agree with our findings. On one hand, Lamarckian converges on better solutions on average for noiseless problems as shown in Figure 4. On the other hand, Figure 5 shows that Baldwinian is significantly better than Lamarckian on noisy problems. Baldwinian's convergence speed is significantly higher compared to Lamarckian in all test cases on average. However, in the case of Rastrigin, the simulation result shows that Baldwinian converges faster than Lamarckian, but not as optimal as its converged solutions. The convergence speed for Baldwinian is faster than Lamarckian because the re-sampling process increases the chances of selecting good parents to reproduce. However, towards the end of the search, the effect of Baldwinian diminishes as the selection pressure is not as important as it is at the beginning of the search. Since the global search narrows down the searching region in the global optimum basin, a line search is more appropriate in this situation.

Table 2: Benchmark Functions

Name	Range	Equation	Dim (d)
Ackley	[-32, 32]	$F_{Ackley}(\mathbf{x}) = -20 \cdot \exp(-0.2\sqrt{\frac{1}{d}\sum_{i=1}^{d} x_i^2}) - \exp(\frac{1}{d}\sum_{i=1}^{d}\cos(2\pi x_i)) + 20 + e$	30
Hybrid F8F2	[-3, 1]	$F_{HybridF8F2}(\mathbf{x}) = \sum_{i=2}^{n-1}(\frac{z(x_{i-1},x_i)}{4000}) - \cos(z(x_{i-1},x_i)) + 1.0 + (\frac{z(x_n,x_1)}{4000}) - \cos(z(x_n,x_1)) + 1.0$ $z(x,y) = 100(x^2-y)^2 + (x-1)^2$	30
Rastrigin	[-5, 5]	$F_{Rastrigin}(\mathbf{x}) = 10n + \sum_{i=1}^{d} x_i^2 - 10\cdot\cos(2\cdot x_i)$	30
Griewank	[-600, 600]	$F_{Griewank}(\mathbf{x}) = \frac{1}{4000}\cdot\sum_{i=1}^{d}(x_i-100)^2 - \prod_{i=1}^{d}\frac{x_i-100}{\sqrt{i}}) + 1$	30
Schwefel	[-3.142, 3.142]	$F_{Schwefel}(\mathbf{x}) = \sum_{i=1}^{d}(\sum_{j=1}^{i} x_j)^2$	30

CONCLUSION AND FUTURE WORK

In this paper, we analysed the behaviour of Lamarckian evolution and Baldwinian effect in the absence and presence of noise in continuous optimisation problems. The comprehensive study shows that Lamarckian performs well in noiseless problems whereas Baldwinian performs better in noisy problems. The empirical results are consistent with our previous analysis of line search. Results from our studies have the implication on the specific use of replacement policy for problem having specific characteristic (i.e. noisy or noiseless continuous optimisation problem).

This study is at first step towards further understanding of MC for problem solving. Future studies would include investigations of Lamarckian and Baldwinian approach to other problems such as combinatorial optimisation and multi-objective optimisation.

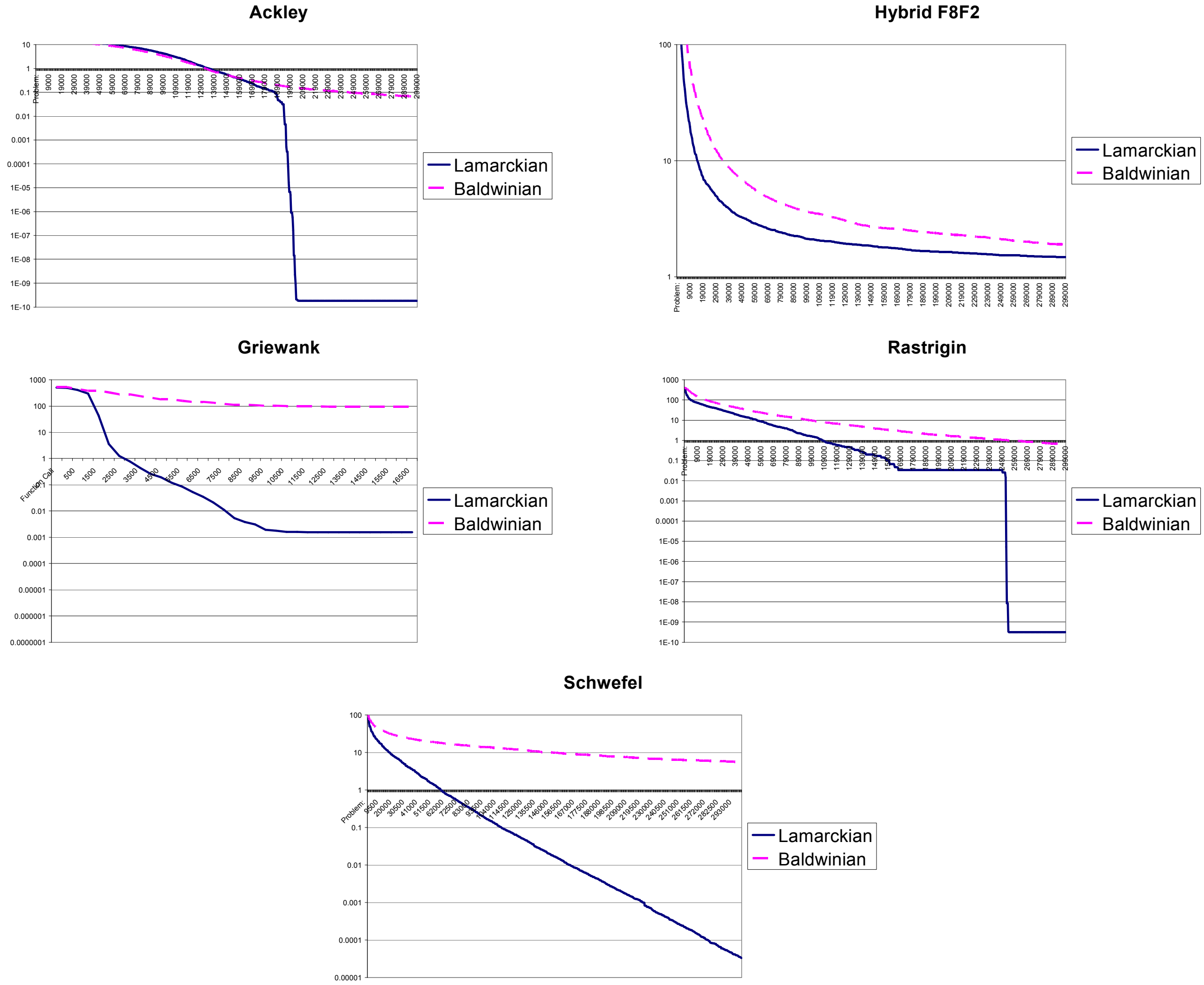

Figure 4: Convergence Speed between Baldwinian and Lamarckian on Noiseless Problem

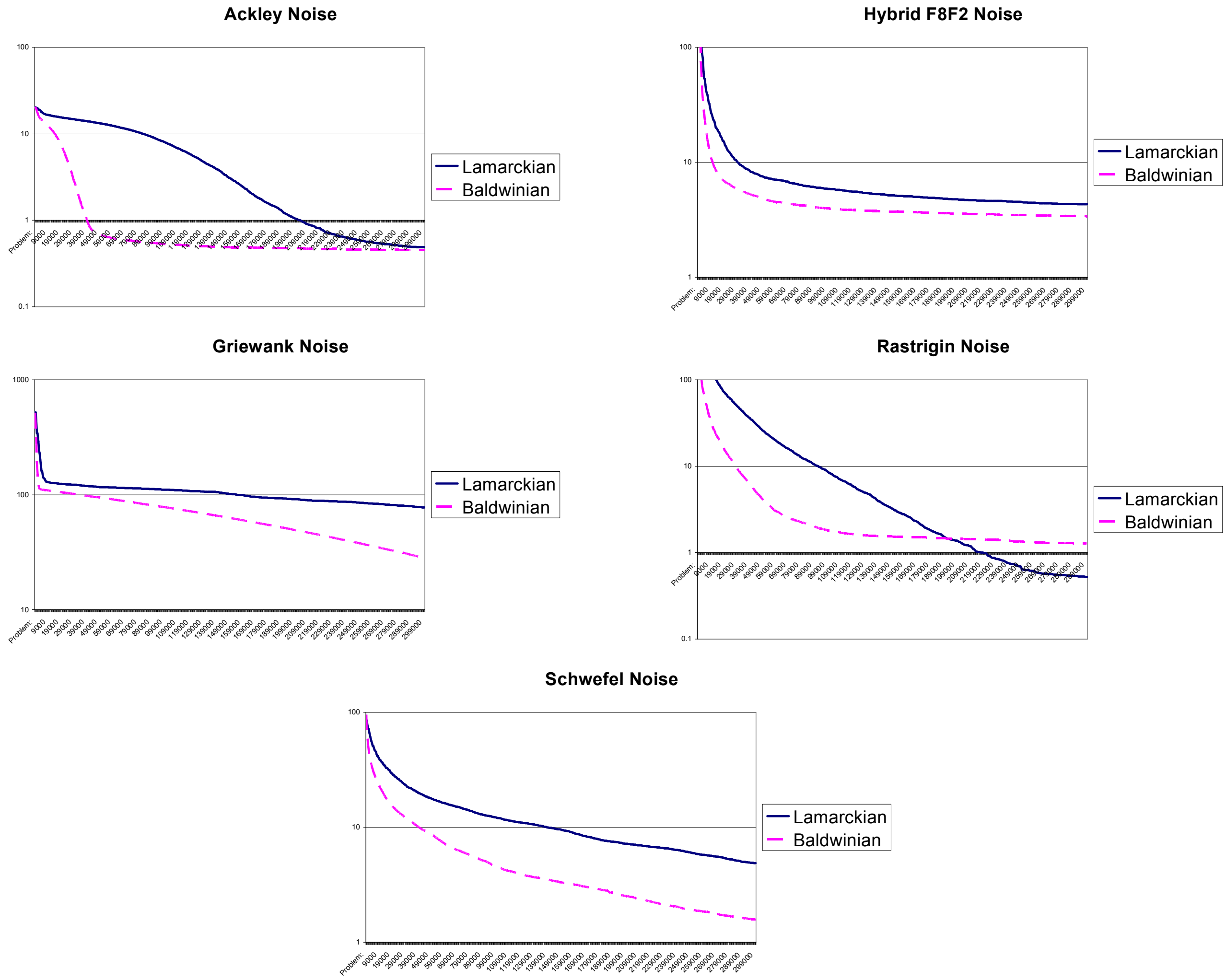

Figure 5: Convergence Speed between Baldwinian and Lamarckian on Noisy Problem with θ=1E-03.

ACKNOWLEDGEMENT

The authors acknowledge the funding support of Singapore Technologies (ST) Engineering Pte Ltd.

REFERENCES

Castillo, P. A., M. G. Arenas, et al. (2006). Lamarckian evolution and the Baldwin Effect in Evolutionary Neural Networks.

De-Jong, K. (2006). Evolutionary Computation: A Unified Approach, Prentice Hall of India Private Limited.

Goldberg, D. E. (1989). Genetic Algorithms in Search, Optimization & Machine Learning, Addison-Wesley.

Julstrom, B. A. (1999). Comparing Darwinian, Baldwinian, and Lamarckian Search in a Genetic Algorithm for the 4-Cycle Problem. Late Breaking Papers at the 1999 Genetic and Evolutionary Computation Conference**:** 134--138.

Krasnogor, N. and J. Smith (2005). "A tutorial for competent MAs: model, taxonomy, and design issues." Evolutionary Computation, IEEE Transactions on **9**(5): 474-488.

Mayley, G. (1996). "Landscapes, Learning Costs and Genetic Assimilation." Evolutionary Computation **4**: 213-234.

Moscato, P. (1989). On Evolution, Search, Optimization, Genetic Algorithms and Martial Arts: Towards MAs, Caltech Concurrent Computation Program.

M. N. Le, Y. S. Ong, Y. Jin & B. Sendhoff. (2009). 'Lamarckian MAs: local optimum and connectivity structure analysis', Memetic Computing Journal, Vol. 1, No. 3, pp. 175-190.

Nguyen, Q. H., Y. S. Ong, et al. (2009). "A Probabilistic Memetic Framework." Evolutionary Computation, IEEE Transactions on **13**(3): 604-623.

Ong, Y. S. and A. J. Keane (2004). "Meta-Lamarckian learning in MAs." Evolutionary Computation, IEEE Transactions on **8**(2): 99-110.

Tsutsui, S. and A. Ghosh (1997). "Genetic algorithms with a robust solution searching scheme." Evolutionary Computation, IEEE Transactions on **1**(3): 201-208.

Whitley, L. D., V. S. Gordon, et al. (1994). Lamarckian evolution, The Baldwin Effect and Function Optimization.

Proceedings of the International Conference on Evolutionary Computation. The Third Conference on Parallel Problem Solving from Nature: Parallel Problem Solving from Nature, Springer-Verlag.

Wolpert, D. H. and W. G. Macready (1997). "No Free Lunch Theorems for Optimization." IEEE Transactions on Evolutionary Computation **1**(1): 67-82.

Y. S. Ong, P. B. Nair and K. Y. Lum (2006). 'Max-Min Surrogate-Assisted Evolutionary Algorithm for Robust Design', IEEE Transactions on Evolutionary Computation, Vol. 10, No. 4, pp. 392-404.

AUTHORS BIBLIOGRAPHY

CHENG WAI KHENG is a Postgraduate Student of School of Computer Science, University of Nottingham, Malaysia Campus. He is a member of Intelligent Decision Support Systems Research Group, University of Nottingham, Malaysia. His current research interests include the application of Hyper/Meta Heuristic, Evolutionary Computation and Optimisation. He received his B.Sc in Computer Science from University of Nottingham, Malaysia in 2007.

MENG HIOT LIM is currently an Associate Professor at the School of Electrical & Electronics Engineering, Nanyang Technological University. He received his B.Sc., M.Sc. and PhD from the University of South Carolina. He holds concurrent appointment as Deputy Director of both the Centre for Financial Engineering, a multi-disciplinary research centre anchored at the Nanyang Business School and the highly regarded M.Sc. in Financial Engineering program. His research interests include memetic computation, computational intelligence, combinatorial and continuous domains optimization, evolvable hardware systems, computational finance and graph theory. He is the Managing Editor-in-Chief of Memetic Computing Journal, the Editor-in-Chief of a Book Series on Adaptation, Learning, and Optimization published by Springer-Verlag and an Associate Editor of IEEE Transactions on SMC – Part C. Dr Lim has also guest-edited several special issues in international journals. His research has been funded by agencies such as the NSTB (now known as A*Star), Temasek Defense Strategic Initiatives and industries which include ST Engineering, ST Yellow Pages, Seiko Instruments, and Boeing Phantom Works. He played major role in international conferences, such serving as INISTA-2009 General Chair, ISIC-09 Publicity Chair, CEC-7 Logistic Chair, etc.; and currently serving as the General Chair of ICCP-2010 in China and SEMCCO-2010 in India.

SIANG YEW CHONG is an Assistant Professor with the School of Computer Science, University of Nottingham, Malaysia Campus, a member of the Automated Scheduling, Optimization and Planning (ASAP) Research Group, University of Nottingham, UK, and an Honorary Research Fellow with the School of Computer Science, University of Birmingham, UK. He was a Research Associate with the Centre of Excellence for Research in Computational Intelligence and Applications (CERCIA), University of Birmingham, UK, in 2007. He is an Associate Editor of IEEE Transaction on Computational Intelligence and AI in Games. His major research interests include evolutionary computation, neural networks, and evolutionary game theory. Dr. Chong received the Ph.D. degree in Computer Science from the University of Birmingham, UK, in 2007, and was awarded the 2009 IEEE Computational Intelligence Society Outstanding Ph.D. Dissertation Award for his work on theoretical frameworks for performance analysis in co-evolutionary learning.