Combining Landscape Approximation and Local Search in Global Optimization

Ko-Hsin Liang, Xin Yao and Charles Newton
Computational Intelligence Group, School of Computer Science
University College, The University of New South Wales
Australian Defence Force Academy, Canberra, ACT, Australia 2600
Email: {liangk, xin, csn}@cs.adfa.oz.au

Abstract- Local search techniques have been applied in variant global optimization methods. The effect of local search to the function landscape can make multimodal problems easier to solve. For evolutionary algorithms, the usage of the step size control concept normally will result in failure by the individual to escape from the local optima during the final stage. In this paper, we propose an algorithm combining landscape approximation and local search (LALS) which is designed to tackle those difficult multimodal problems. We demonstrate that LALS can solve problems with very rough landscapes and also that LALS has very good global reliability.

1 Introduction

Evolutionary algorithms (EA) have been applied to many optimization problems successfully in recent years. Genetic algorithms (GA) [1], evolution strategies (ES) [2] and evolutionary programming (EP) [3] are three major fields in EA. One of the essential applications of EA is global optimization on numerical problems [4, 5, 6, 7]. A common feature of EA on solving numerical functions is the usage of the step size concept. EP and ES both use self-adaptation to adjust the step size and improve the search progress. The self-adaptation schemes normally have the trend of tuning the objective variables with smaller step sizes during the latter evolutionary stage. In a multimodal problem, if the individual stays in a local optimum at this time and the landscape between the final local optimum and the global one is really rough or very hilly, it is inevitable that the final solution will become trapped and the probability to jump over the hurdle will be very small.

To overcome this difficulty, local search techniques can be used to indirectly change the roughness of the fitness landscape. In Figure 1, the effect of local search on the one-dimensional (1D) landscape of generalized Rastrigin's function is displayed. The difficult problem becomes easier to solve after applying the local search method. It has previously been used in genetic algorithms for function optimization [8, 9]. In this paper we use the Lamarckian evolution to form a population such that each individual is located in different local minima. Using the landscape bottom information, we can approximate the combined shape of the function landscape. Therefore a better offspring can be generated using interpolation approximation methods.

In this paper we develop a novel method using landscape approximation and local search (LALS) with EA to solve multimodal problems. The empirical results of 18 benchmark problems show that LALS has very good global reliability, which means the finding of the global optimum is almost guaranteed. However, local search is expensive. In a few case LALS consumes more computation time than the other method used in this study for comparison.

The rest of this paper is organized as follows. Section 2 describes the benefits of applying local search techniques in global optimization. The details of the LALS algorithm are also given here. The empirical results and discussion are presented in Section 3. Finally, Section 4 concludes with some remarks and future research directions.

2 Benefits from Local Search

The local search techniques have been applied in variant global optimization methods. The simplest way to benefit from a local search procedure is known as Multistart. This method applies local search from each randomly generated point. However, local searches are the most time consuming parts in the algorithm. Obviously, the inefficiency is caused by using many local searches to find the same minimum several times. To improve the inefficiency, clustering methods [10] and the Multi Level Single Linkage [11] are designed. They use variant clustering techniques to link points to different groups. Within each group local search is only conducted on one point and the found local optimum is assumed to be the representative of that group. These methods provide impressive results to the benchmark test function in Dixon and Szegö (1978) [12]. However, when solving a problem with a large number of minima, these methods may not be suitable [11, 13].

In recent years, evolutionary algorithms become increasingly robust and easy to use as a global optimization

0-7803-5536-9/99/$10.00 ©1999 IEEE

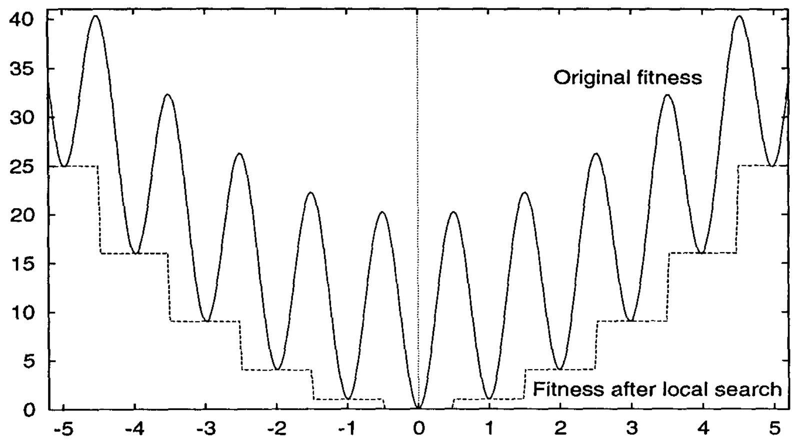

Figure 1: The effect of local search on the 1D landscape of generalized Rastrigin's function.

method. Applying local search in the GA has substantially improved their performance for some multimodal problems [8, 9]. In terms of evolution, the local search can be thought of as a consequence of individual learning during the individual's life time. Combining learning and evolution has the effect of changing the fitness landscape. The fitness of an individual will be changed after applying a learning process (i.e., local search). However, the individual's genetic codes (parameters) may or may not be altered depending on the way that learning and evolution interact. They are known as *Lamarckian evolution* and the *Baldwin effect* [14, 15], respectively. In Hart and Belew [9], the performance of the Baldwin effect and Lamarckian evolution are compared. It should be noticed that the usage rates of local search to the GA have different impacts on the evolutionary process. To design an efficient search algorithm, the application of the local search should be carefully considered.

2.1 Landscape Approximation with Local Search Algorithm (LALS)

In this paper, we propose an evolutionary search algorithm which uses Lamarckian evolution to initialize the population. Each individual can be thought of as a point located at different basins of the fitness landscape. With the landscape information, we apply the quadratic approximate to estimate the approximate location of the next offspring. Hopefully, after learning, this offspring will have a higher probability to reach a better minimum. Figure 2 demonstrates the 1D landscape approximation with local search result on Ackley's function. P_1, P_2, P_3 are three local minimal points, P_4 is the estimated point and is mapped to the fitness point P_4'. After proceeding a local search from P_4', the potential of finding better optimum can be strongly expected.

The landscape approximation with local search is implemented as follows:

1. Initialize μ individuals at random, each individual performs a local search.
2. REPEAT
   (a) Generate 3 points P_1, P_2, P_3 by global discrete recombination.
   (b) Perform a quadratic approximation using P_1, P_2, P_3 to produce a point P_4.
   (c) Proceed a local search from P_4 and update P_4 with the search result.
   (d) Place P_1, P_2, P_3, P_4 in the population and do a $(\mu + 4)$ truncation selection.
3. UNTIL termination criteria are met.

The local search method used in this paper is the Local Evolutionary Search with Random Memorizing (LESRM) from Voigt and Lange (1998) [16]. LESRM generates promising new search directions from prestored solutions to the current point. A generated good solution is stored in a sequential memory which is randomly accessed later. Very impressive results of solving some unimodal functions are shown in [16].

For global discrete recombination, each new individual's objective variable is decided at random and copied from one of the parents:

$$x_{i,j} = x_{\chi,j}, \forall j \in \{1, \cdots, n\},$$

where $x_{i,j}$ denotes the j-th component of the vectors x_i, $\forall i \in \{1, 2, 3\}$, n is the dimensionality. χ denotes a uniform distributed random integer in $\{1, \cdots, \mu\}$, and μ is the population size.

We approximate the position of P_4 using the quadratic interpolation method as follow.

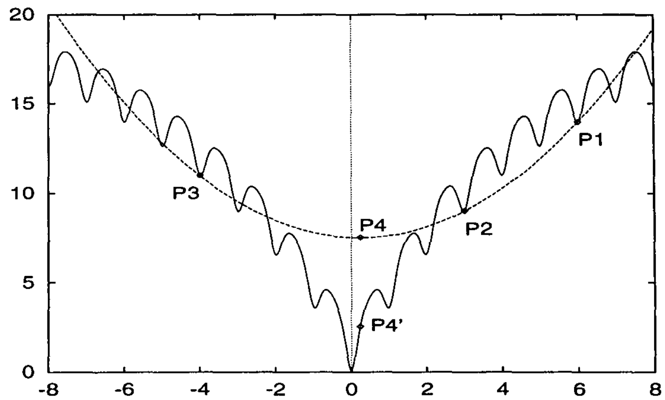

Figure 2: The 1D landscape approximation with local search result on Ackley's function

$$x_{4,j} = \frac{1}{2} \cdot \frac{(x_{2,j}^2 - x_{3,j}^2)f(x_1) + (x_{3,j}^2 - x_{1,j}^2)f(x_2) + (x_{1,j}^2 - x_{2,j}^2)f(x_3)}{(x_{2,j} - x_{3,j})f(x_1) + (x_{3,j} - x_{1,j})f(x_2) + (x_{1,j} - x_{2,j})f(x_3)}$$

The algorithm terminates when the maximum number of function evaluations used is reached or the differences of the fitness values of P_1, P_2, P_3 are less than 1e-6.

3 Experiments

The LALS algorithm is designed to solve multimodal problems. We use all the multimodal functions tested in [7] and two Schaffer's functions [17] in our experimental studies. Table 1 lists all the test functions. We use the same problem numbering as in [7] followed by the two Schaffer's functions. Each problem was run 50 times, the population size was 30, and the maximum number of function evaluations was 500000.

To evaluate the performance of LALS, we also list the results of Improved Fast EP(IFEP) [18] in the tables in the next two sections. In IFEP each parent generates two offspring, one with Gaussian and the other with Cauchy mutation. It is a very simple and effective algorithm for function optimization. The parameter setup follows the suggestions from [18]. It is difficult to rate one algorithms against another since the solution precision, function evaluations, and ratios of global optimum differ in such a way that neither algorithm demonstrate superiority. These three indices are used for comparison in our studies.

3.1 Multimodal Functions With Many Local Minima(f_8–f_{13})

Functions f_8–f_{13} are multimodal functions with many local minima. The number of local minima increases exponentially as the function dimension increases [2, 19]. These functions appear to be very "rugged" and difficult to optimize. Table 2 lists experimental results for IFEP and LALS. LALS has better solution quality and global hits on all functions except f_{13}. However, more function evaluations used were necessary on f_{10} and f_{12}. For f_{13}, we increased the limitation of the maximal function evaluations to 700000. A better mean fitness 1.82e-11 was obtained with full global hit and 614252 evaluations on average.

Although the better results of f_8 and f_9 are found using LALS, the global hits are not as high as for other functions. We run the experiments again with population sizes 50 and 60. The results are shown in table 3. Both f_8 and f_9 obtained improved mean fitness at the cost of increased computation time. The global hits of f_9 increase as expected, but not for f_8. Figure 3 illustrates why f_8 may be difficult for LALS. The main reason lies in our simple one-dimensional quadratic approximation which may generate a point that is a "local maximum" as the curve is convex. However, this can be fixed by moving away from the calculated P_4 if the curve is convex.

For most approximations, the estimated points will fall close in the central area like arc($a_1a_2a_4$). Only a few arcs like arc($a_1a_2a_3$) and arc($a_3a_5a_6$) will generate points over the boundaries. In this case we reset the points to the boundary values which may help to locate the global minimum. However, the two best minima are located far apart. It is difficult for f_8 to use the quadratic interpolation method and have a large jump from one side to another.

3.2 Multimodal Functions With a Few Local Minima(f_{14}–f_{25})

The final results of IFEP and LALS on functions f_{14}–f_{25} are summarized in Table 4. It is clear that LALS has better global reliability than IFEP. For f_{16}, f_{17}, f_{18}, f_{19}, f_{25}, LALS performs the same as IFEP, but spends more function evaluations on all but f_{16}. For the rest of the problems, LALS has a better ratio of finding global

Table 1: The 18 test functions used in our experimental studies, where n is the dimension of the function, f_{min} is the minimum value of the function, and $S \subseteq R^n$.

Test function	n	S	f_{min}
$f_8(x) = \sum_{i=1}^{n} -x_i \sin(\sqrt{\lvert x_i \rvert})$	30	$[-500, 500]^n$	-12569.5
$f_9(x) = \sum_{i=1}^{n}[x_i^2 - 10\cos(2\pi x_i) + 10)]$	30	$[-5.12, 5.12]^n$	0
$f_{10}(x) = -20\exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}\right) - \exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos 2\pi x_i\right) +20 + e$	30	$[-32, 32]^n$	0
$f_{11}(x) = \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	30	$[-600, 600]^n$	0
$f_{12}(x) = \frac{\pi}{n}\left\{10\sin^2(\pi y_1) + \sum_{i=1}^{n-1}(y_i - 1)^2[1 + 10\sin^2(\pi y_{i+1})] +(y_n - 1)^2\right\} + \sum_{i=1}^{n} u(x_i, 10, 100, 4)$, $y_i = 1 + \frac{1}{4}(x_i + 1)$, $u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m, & x_i > a, \\ 0, & -a \le x_i \le a, \\ k(-x_i - a)^m, & x_i < -a. \end{cases}$	30	$[-50, 50]^n$	0
$f_{13}(x) = 0.1\left\{\sin^2(3\pi x_1) + \sum_{i=1}^{n-1}(x_i - 1)^2[1 + sin^2(3\pi x_{i+1})] +(x_n - 1)^2[1 + \sin^2(2\pi x_n)]\right\} + \sum_{i=1}^{n} u(x_i, 5, 100, 4)$	30	$[-50, 50]^n$	0
$f_{14}(x) = \left[\frac{1}{500} + \sum_{j=1}^{25}\frac{1}{j + \sum_{i=1}^{2}(x_i - a_{ij})^6}\right]^{-1}$	2	$[-65.536, 65.536]^n$	1
$f_{15}(x) = \sum_{i=1}^{11}\left[a_i - \frac{x_1(b_i^2 + b_i x_2)}{b_i^2 + b_i x_3 + x_4}\right]^2$	4	$[-5, 5]^n$	0.0003075
$f_{16}(x) = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1 x_2 - 4x_2^2 + 4x_2^4$	2	$[-5, 5]^n$	-1.0316285
$f_{17}(x) = \left(x_2 - \frac{5.1}{4\pi^2}x_1^2 + \frac{5}{\pi}x_1 - 6\right)^2 + 10\left(1 - \frac{1}{8\pi}\right)\cos x_1 + 10$	2	$[-5, 10] \times [0, 15]$	0.398
$f_{18}(x) = [1 + (x_1 + x_2 + 1)^2(19 - 14x_1 + 3x_1^2 - 14x_2 +6x_1x_2 + 3x_2^2)] \times [30 + (2x_1 - 3x_2)^2(18 - 32x_1 +12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)]$	2	$[-2, 2]^n$	3
$f_{19} = -\sum_{i=1}^{4} c_i \exp\left[-\sum_{j=1}^{4} a_{ij}(x_j - p_{ij})^2\right]$	4	$[0, 1]^n$	-3.86
$f_{20} = -\sum_{i=1}^{4} c_i \exp\left[-\sum_{j=1}^{6} a_{ij}(x_j - p_{ij})^2\right]$	6	$[0, 1]^n$	-3.32
$f_{21} = -\sum_{i=1}^{5}[(x - a_i)^T(x - a_i) + c_i]^{-1}$	4	$[0, 10]^n$	$-1/c_1$
$f_{22} = -\sum_{i=1}^{7}[(x - a_i)^T(x - a_i) + c_i]^{-1}$	4	$[0, 10]^n$	$-1/c_1$
$f_{23} = -\sum_{i=1}^{10}[(x - a_i)^T(x - a_i) + c_i]^{-1}$ where $c_1 = 0.1$	4	$[0, 10]^n$	$-1/c_1$
$f_{24} = 0.5 + \frac{\sin^2\sqrt{x_1^2 + x_2^2} - 0.5}{[1.0 + 0.001(x_1^2 + x_2^2)]^2}$	2	$[-100, 100]^n$	0
$f_{25} = (x_1^2 + x_2^2)^{0.25}[\sin^2(50(x_1^2 + x_2^2)^{0.1}) + 1.0]$	2	$[-100, 100]^n$	0

Table 2: The results of IFEP and LALS on f_8–f_{13}. "Eval." is the number of evaluations.

Prob	IFEP					LALS			
	Eval.	Mean	Std Dev	Found at	Global hit	Eval.	Mean	Std Dev	Global hit
8	900000	-10640.18	431.21	757500	0/50	199244	-11834.65	298.78	1/50
9	500000	2.89	1.98	500000	1/50	306280	2.67	1.58	5/50
10	150000	6.33e-4	3.86e-4	150000	50/50	370686	2.08e-12	2.11e-12	50/50
11	200000	1.27e-1	1.90e-1	176500	3/50	173592	1.87e-10	1.31e-9	50/50
12	150000	2.49e-2	5.70e-2	150000	40/50	476245	5.84e-9	2.16e-8	50/50
13	300000	4.42e-8	2.65e-8	150000	50/50	504212	1.19e-2	3.01e-2	33/50

Table 3: The results of LALS on f_8–f_9 with population size 50 and 60.

Prob.	Pop. size	Eval.	Mean	Std Dev	Global hit
8	50	327434	-12296.68	119.27	1/50
8	60	391634	-12358.64	166.33	12/50
9	50	508021	2.98e-1	4.97e-1	36/50
9	60	610163	2.19e-1	4.12e-1	39/50

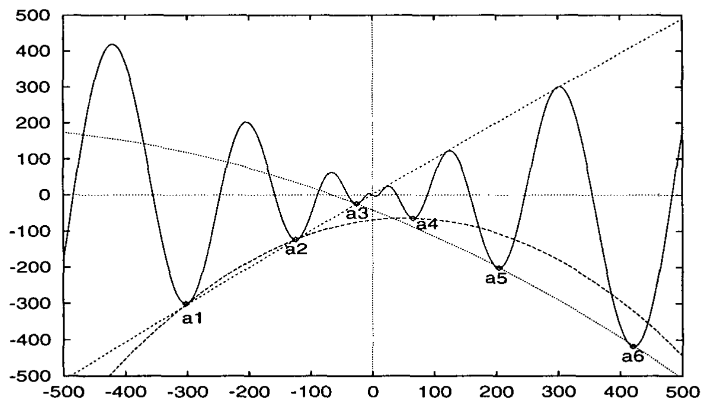

Figure 3: The 1D landscape approximation examples on Schwefel's function (f_8)

Table 4: The results of IFEP and LALS on f_{14}–f_{25}. "Eval." is the number of evaluations.

Prob	IFEP					LALS			
	Eval.	Mean	Std Dev	Found at	Global hit	Eval.	Mean	Std Dev	Global hit
14	10000	1.56	0.96	10000	33/50	3052	1.04	0.20	48/50
15	400000	4.17e-4	2.98e-4	235500	44/50	111748	3.0749e-4	2.45e-10	50/50
16	10000	-1.031628	2.70e-8	3500	50/50	2817	-1.031628	2.70e-8	50/50
17	10000	0.3979	8.26e-9	4500	50/50	5496	0.3979	8.26e-9	50/50
18	10000	3	0	4500	50/50	4676	3	0	50/50
19	10000	-3.86	0	6000	50/50	6852	-3.86	0	50/50
20	20000	-3.26	5.90e-2	17500	24/50	17504	-3.32	2.35e-7	50/50
21	10000	-7.02	2.74	10000	21/50	13790	-10.1532	0	50/50
22	10000	-8.42	2.96	9500	34/50	13354	-10.4029	2.16e-7	50/50
23	10000	-9.09	2.87	7500	39/50	14312	-10.5364	4.58e-7	50/50
24	10000	8.86e-3	2.67e-3	8500	4/50	10754	3.89e-4	1.90e-3	48/50
25	10000	2.47e-3	1.27e-3	10000	50/50	15614	1.07e-4	7.51e-4	50/50

optima. However, LALS spends more computation efforts except f_{14} and f_{15}. It is worth noticing that the big difference of global hits happens on f_{24}. Figure 4 displays the 1D landscape of Schaffer's function f_{24}. For traditional step size control EA, it is difficult for individuals to escape out of local minima during the later evolutionary stage. The way LALS is designed, it does ignore the high barriers among the global minimum and local minima. For f_{14} and f_{24}, LALS can improve the global hits to 100% with population size 40.

3.3 Discussion

According to the experimental results, the population size is a very important factor to affect LALS's performance. Larger population sizes generate better diversity and more information of the fitness landscape which provides higher probability to find the global minimum. However, the basic premium will increase at the beginning because each individual needs to perform a local search. In table 5, by using a lesser population size on simpler functions (f_{16}–f_{23}) one can also obtain the same quality of solution, but spend less function evaluations than IFEP. It is difficult to determine the population size while applying LALS to practical problems which have little knowledge about the landscape.

One disadvantage that can be observed is that LALS does not perform efficiently on some multimodal functions with many local minima. We believe that the inefficiency is caused by either the inadequacy of the quadratic interpolation method or duplicate usages of the local search method with close start points. This is something we would like to find out in our future research.

4 Conclusion

This paper proposes a new evolutionary algorithm for global optimization (i.e., LALS). Extensive empirical studies on 18 benchmark multimodal problems were carried out to evaluate the performance of LALS. For most of problems, LALS can find the global minima with 100% reliability. The population size is an important parameter to control the LALS efficiency. For problems with convex bottom structure, the quadratic interpolation method may not be a good approach. However, it still can improve the solution quality with large population size.

The future work of this research includes theoretical analysis of LALS and the improvement of its efficiency. The local search method(LESRM) we use is a very fast and robust algorithm when applied in a stand alone mode. However, it may be better to combine another local search technique such as Powell method [20] into LALS to provide better efficiency. The quadratic approximation currently is done on a one-dimensional base. It would be better done by approximating the original landscape on the n-dimensional base.

Bibliography

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

[2] H.-P. Schwefel, *Evolution and Optimum Seeking.* New York: Wiley, 1995.

[3] D. Fogel, *Evolutionary Computation: Toward a New Philosophy of Machine Intelligence.* New York: IEEE Press, 1995.

[4] T. Bäck and H.-P. Schwefel, "An overview of evolutionary algorithms for parameter optimization," *Evolutionary Computation*, vol. 1, no. 1, pp. 1–23, 1993.

[5] T. Bäck, *Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms.* New York: Oxford University, 1996.

[6] H. Mühlenbein and D. Schlierkamp-Voosen, "Predictive models for the breeder genetic algorithm I. Continuous parameter optimization," *Evolutionary Computation*, vol. 1, no. 1, pp. 25–49, 1993.

[7] X. Yao and Y. Liu, "Fast evolutionary programming," in *Evolutionary Programming V: Proc. of the Fifth Annual Conference on Evolutionary Programming* (L. Fogel, P. Angeline, and T. Bäck, eds.), (Cambridge, MA), pp. 451–460, MIT Press, 1996.

[8] D. Whitley, V. Gordon, and K. Mathias, "Lamarkian evolution, the baldwin effect and function optimization," in *Parallel Problem Solving from Nature-PPSN III* (Y. Davidor, H.-P. Schwefel, and R. Männer, eds.), vol. 866 of *Lecture Notes in Computer Science*, (Berlin Heidelberg), pp. 6–15, Springer-Verlag, 1994.

[9] W. Hart and R. Belew, "Optimization with genetic algorithm hybrids that use local search," in *Adaptive Individuals in Evolving Populations: Models and Algorithms* (R. Belew and M. Mitchell, eds.), vol. 26 of *SFI Studies in the Sciences of Complexity*, (Reading, MA), pp. 483–496, Addison-Wesley, 1996.

[10] A. Törn, "A search-clustering approach to global optimization," in *Towards Global OPtimization 2* (L. Dixon and G. Szegö, eds.), (Amsterdam), pp. 49–62, North-Holland, 1978.

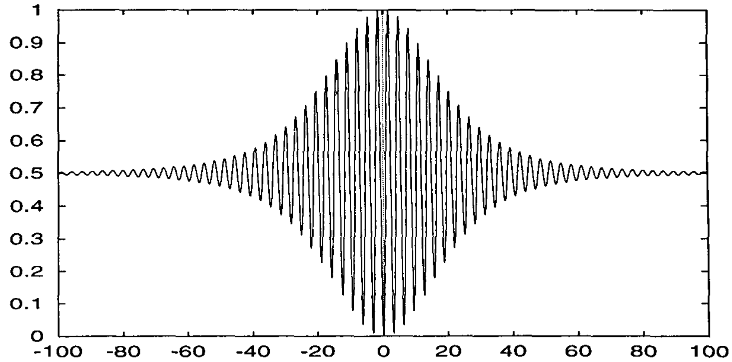

Figure 4: The 1D landscape of Schaffer's function(f_{24}).

Table 5: The results of LALS on f_{16}–f_{23} with population size 10.

Prob.	Pop. size	Eval.	Mean	Std Dev	Global hit
16	10	1063	-1.031628	2.70e-8	50/50
17	10	2278	0.3979	8.26e-9	50/50
18	10	1665	3	0	50/50
19	10	2458	-3.86	0	50/50
20	10	6576	-3.32	5.67e-7	50/50
21	10	4990	-10.1532	1.29e-6	50/50
22	10	5455	-10.4029	2.16e-7	50/50
23	10	5554	-10.5364	4.58e-7	50/50

[11] A. Rinnooy Kan and G. Timmer, "Stochastic global optimization methods part II: Multi level methods," *Mathematical Programming*, vol. 39, pp. 57–78, 1987.

[12] L. Dixon and G. Szegö, "The global optimization problem: An introduction," in *Towards Global OPtimization 2* (L. Dixon and G. Szegö, eds.), (Amsterdam), pp. 1–15, North-Holland, 1978.

[13] M. Ali, C. Storey, and A. Törn, "Application of stochastic global optimization algorithms to practical problems," *Journal of Optimization and Application*, vol. 95, pp. 545–563, 1997.

[14] J. Baldwin, "A new factor in evolution," *American Naturalist*, vol. 30, pp. 441–451, 1896.

[15] G. Hinton and S. Nolan, "How learning can guide evolution," *Complex Systems*, vol. 1, pp. 495–502, 1987.

[16] H.-M. Voigt and J. Lange, "Local evolutionary search enhancement by random memerizing," in *Proceedings of the 1998 IEEE International Conference on Evolutionary Computation(ICEC'98)*, (Piscataway, NJ), pp. 547–552, IEEE Press, 1998.

[17] J. Schaffer, R. Caruana, L. Eshelman, and R. Das, "A study of control parameters affecting online performance of genetic algorithms fot function optimization," in *Proceedings of the third International Conference on Genetic Algorithms(ICGA'89)* (J. Schaffer, ed.), (San Mateo, CA), pp. 51–60, Morgan Kaufmann, 1989.

[18] X. Yao, G. Lin, and Y. Liu, "An analysis of evolutionary algorithms based on neighbourhood and step sizes," in *Evolutionary Programming VI: Proc. of the Sixth Annual Conference on Evolutionary Programming* (P. Angeline, R. Reynolds, J. McDonnell, and R. Eberhart, eds.), vol. 1213 of *Lecture Notes in Computer Science*, (New York), pp. 297–307, Springer, 1997.

[19] A. Törn and A. Žilinskas, *Global Optimization*, vol. 350 of *Lecture Notes in Computer Science*. New York: pringer-Verlag, 1989.

[20] M. Powell, "An efficient method for finding the minimum of a function of several variables without calculating derivatives," *The Computer Journal*, vol. 7, pp. 155–162, 1964.