2003 International Symposium on Advanced Intelligent Systems
September 25-28, 2003, Jeju, Korea

A Study on Modeling of Search Space with GA Sampling

Yoshifumi Banno
Graduate School of Engineering
Mie Univ.

Miho Ohsaki
Faculty of Information
Shizuoka Univ.

Tomohiro Yoshikawa
Faculty of Engineering
Mie Univ.

Tsuyoshi Shinogi
Faculty of Engineering
Mie Univ.

Shinji Tsuruoka
Faculty of Engineering
Mie Univ.

Abstract

To model a numerical problem space under the limitation of available data, we need to extract sparse but key points from the space and to efficiently approximate the space with them. This study proposes a sampling method based on the search process of genetic algorithm and a space modeling method based on least-squares approximation using the summation of Gaussian functions. We conducted simulations to evaluate them for several kinds of problem spaces: DeJong's, Schaffer's, and our original one. We then compared the performance between our sampling method and sampling at regular intervals and that between our modeling method and modeling using a polynomial. The results showed that the error between a problem space and its model was the smallest for the combination of our sampling and modeling methods for many problem spaces when the number of samples was considerably small.

1 Introduction

The concern with such learning systems based on the interaction between human and computer has been growing for the last several years as autonomous symbiotic robot [5, 6], Web personalization [7], artistic design support [13], and physical impairment compensation [13]. Although these systems need the feedback of user's subjective instruction and/or evaluation on their behavior, that makes the user fatigued and takes much time. The quantity of subjective data is then severely limited comparing with that of objective data obtained with monitoring sensors. Consequently, it is difficult for an interactive learning system to successfully learn its behavior based on such small subjective data.

There are some studies to compensate for the lack of learning data using the model of a problem space based on response surface method [9] or neural network : adding the outputs from a model to the real outputs from a problem space to increase learning data [3, 10, 14] and visualizing a model to make it possible for humans to grasp and to support the learning process [4, 8]. However, most them are specific to applications and do not systematically examine the accuracy of modeling for several kinds of problem spaces. In addition, there are only a few studies that focus on how to actively pick up important data points for modeling from a problem space [4].

Therefore, we discuss efficient sampling and modeling methods under the limitation of available data and examine how effective they are for what kind of problem spaces. Note, however, that we define the limiting conditions for the problem space and its model in this study, since sampling and modeling methods strongly depend on them.

Conditions of a Problem Space: **(1)** The problem space is nearly static and consists of numerical axes. **(2)** We can previously obtain the information on the axes. While, we cannot do so on the space landscape. **(3)** The sampling place on the problem space is not restricted; we can sample optional data points. While, the number of sampling times is severely restricted due to sampling cost.

Conditions of a Model: **(1)** The model simulates the input-output characteristics of the problem space and is used to compensate for the lack of data for an interactive learning system. **(2)** The model does not need to be the genuine model of the problem space and does not need high approximation accuracy if it has the benefit of the compensation for the lack of data.

Under these limiting conditions, we propose a sampling method based on the search process of Genetic Algorithm (GA), *GA-based Sampling*, and discuss a modeling method based on response surface method [9] using the summation of Gaussian functions, *Gaussian Summation Approximation* and Polynomial. We think that these methods will contribute to the applications mentioned at the beginning of this chapter [5, 6, 7, 13] when they are improved enough and established.

In this paper, we explain our proposed methods, GA-based sampling in Chapter 2 and examine their performance in simulations comparing with sampling at regular intervals and approximation using a polynomial in Chapter 3. Finally, we conclude this paper and note the future work in Chapter 4.

2 GA-based Sampling

The simplest sampling is to divide a problem space at regular intervals and pick up the cross points of the dividing lines. We expediently call it *Mesh Sampling*. Mesh Sampling has a high possibility to miss important data points to grasp the landscape of a problem space such as the apex or the corner points of a large peak (See the right side of Figure 1).

You may come up with a sampling method based on experimental design [2] or information gain [12]. It is efficient to reduce the number of sampling iteration times at a same coordinate on a problem space. However, we made a premise that the number of data is so severely restricted that we cannot get enough data points even at different coordinates. Therefore, this method is not fundamentally available in such a case.

We then propose GA-based Sampling shown in the left side of Figure 1 because of the following features of

GA: rapid initial convergence and probabilistic multi-search. On the former feature, conventional studies on Interactive Evolutionary Computation (IEC) have shown that individuals rapidly arrive in the neighborhood of an optimal point even if the total number of individuals is small [13]. That means GA has the ability to sample data points near by the apex of the largest peak in a problem space. On the latter feature, the diversity of individuals, namely the data point spread on a problem space, is kept through a search process due to random initialization, crossover, and mutation. We then thought that GA can sample the corner points of the largest peak while it can do the points around local peaks.

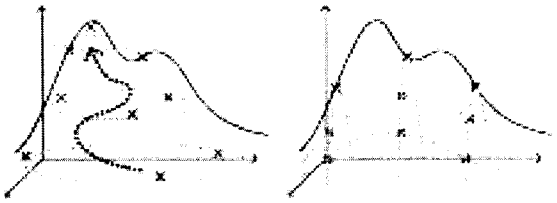

Figure 1: GA-based Sampling(left) and Mesh Sampling(right)

3 Evaluation Experiment

We cannot clarify the absolute performance of our sampling and modeling methods if we do not use them for concrete applications. However, we should systematically examine their effectiveness and generality for various problem spaces as the first step of this research. We then conduct simulations for artificial problem spaces to evaluate their relative performance comparing with other simple methods.

3.1 Comparison Object

In this simulation experiment, we compare GA-based Sampling with Mesh Sampling and Gaussian Summation Approximation with response surface method using a polynomial.

3.2 Experimental Conditions

The experiment consists of three procedures: **(1)** sampling data points from a problem space and modeling it, **(2)** calculating the mean square error, which is the sampling and modeling performance measure, between the model and the problem space for all data points on the problem space, and **(3)** comparing the errors among all combinations of sampling and modeling methods after conducting **(2)(3)**. We go through these procedures for several problem spaces and examine for what kinds of problem spaces GA-based Sampling and Gaussian Summation Approximation are effective.

The combinations of sampling and modeling methods were "Mesh and polynomial", "Mesh and Gaussian", "GA-based and polynomial", and "GA-based and Gaussian", respectively. We used popular benchmark search spaces and our original one: DeJong's functions, from F_1 to F_4 [1], Schaffer's one, F_1[11], and the summation of three Gaussian functions. It takes an infinite time to calculate the mean square error for DeJong's functions, F_3 and F_4 due to their vastness. We modified them to solve this problem as shown in Table 1.

We should determine the conditions of GA operation for GA-based Sampling. If we target a real-world application, it is possible to properly determine GA operation conditions based on the domain knowledge on the application. However, this simulation experiment uses artificial problem spaces, and we do not have any domain knowledge on them. We then previously conducted GA search several times, regarded the result as pseudo domain knowledge, and determined GA operation conditions based on it for each problem space.

Here we discuss how to determine the number of total sample data in simulations. It may be proper to use relative measure, namely the ratio of sample data size to problem space size. However, the number of sample data is absolutely determined in an interactive learning system due to the iteration limit of human instruction or evaluation. We then adopted several values within an absolute range of the number of sample data [13] and observed modeling performance trend to the absolute number of sample data and the relative one.

3.3 Results and Discussion

Figure 2 shows some problem spaces, DeJong's F_1, DeJong's F_2, and Schaffer's F_3, and Table 2 shows the experimental results for them. The results for DeJong's F_3 and F_4, and that for our original problem space were similar to that for DeJong's F_1 and that for Schaffer's F_1, respectively. We then do not mention the details for them. The results for Schaffer's F_1 in Table 2 do not include the results with polynomial approximation, because a 5-degree polynomial function cannot approximate such a complex space due to its nature.

The numerical values in Table 2 are not the magnitude mean square errors between a model and a problem space, but their proportions to the output range of a problem space shown in Table 1. We conducted one of statistical tests, one-side t-test, at 1% significant level on the error proportion difference for all the combinations of sampling and modeling methods.

Sampling Performance

The error of GA-based Sampling was significantly lower than or same as that of Mesh Sampling for all conditions. There were some cases in which the former was about one-fifth of the latter. The error difference between GA-based Sampling and Mesh Sampling was large especially for smaller sample data size.

Mesh Sampling sometimes missed important data points for modeling when the number of data was severely restricted. While, GA-based Sampling could sufficiently extract them due to its fast convergence and probabilistic search. We then conclude that GA-based Sampling was effective under the severe limitation of available data.

The error difference decreased with the increase of the number of data. Although GA-based Sampling was better than Mesh Sampling in this experiment, their positions might become reversed if the sample data were increased as shown in Figure 3. Because Mesh Sampling becomes able to extract data points sensitively and evenly, and GA Sampling becomes unable to do so due to over-convergence.

It depends on not only the size of a problem space but also its landscape complexity when the performance reversal appears. However, we think that it is possible to use the ratio of sample data size to problem space size as the rough measure of reversal and will make a

Table 1: Equations of Problem Spaces

Function	Definition	Input Range & Step	Space Size	Output Range
DeJong F_1	$-\sum_{i=1}^{3} x_i^2$	$-5.11 \leq x_i \leq 5.12$ $\Delta x_i = 0.01$	1.0×10^9	$-78.64 \leq y \leq 0$
DeJong F_2	$-\{100(x_1^2 - x_2)^2 + (1 - x_1)^2\}$	$-2.047 \leq x_i \leq 2.048$ $\Delta x_i = 0.001$	1.7×10^6	$-3899 \leq y \leq 0$
DeJong F_3	$-\sum_{i=1}^{5} \lfloor x_i \rfloor$	$-5.11 \leq x_i \leq 5.12$ $\Delta x_i = 0.01$	1.0×10^{15}	$-25 \leq y \leq 30$
DeJong F_4	$-\{\sum_{i=1}^{3} i x_i^4$ $+\frac{2}{151} \times gauss(0,1)\}$	$-1.27 \leq x_i \leq 1.28$ $\Delta x_i = 0.01$	1.7×10^7	$-16.11 \leq y \leq 0.01$
Schaffer F_1	$0.5 + \frac{(\sin\sqrt{\sum_{i=1}^{2} x_i^2})^2 - 0.5}{1.0+0.0001(\sum_{i=1}^{2} x_i^2)^2}$	$-81.91 \leq x_i \leq 81.92$ $\Delta x_i = 0.01$	2.7×10^8	$0 \leq y \leq 1$
Original	$\sum_{i=1}^{3} h_i \exp\left\{-\frac{{}^t(x-\mu_i)A_i^{-1}(x-\mu_i)}{2}\right\}$	$-8.191 \leq x_i \leq 8.192$ $\Delta x_i = 0.001$	2.7×10^8	$0 \leq y \leq 88.92$

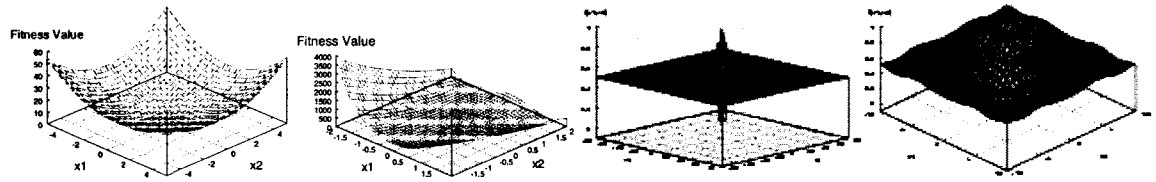

Figure 2: Examples of the Problem Spaces : DeJong's F_1, DeJong's F_2, Schaffer's F_1, and the closeup of Schaffer's F_1

guideline with this measure on what conditions we can apply GA-based Sampling to.

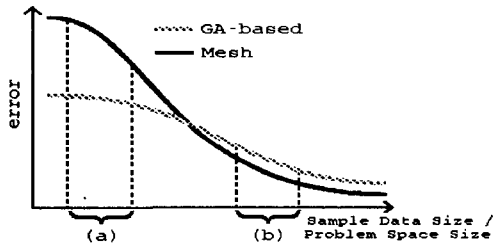

Figure 3: Relation between the Performance of GA-based Sampling and that of Mesh Sampling

Modeling Performance

The error of Gaussian Summation Approximation was lower than that of polynomial approximation for DeJong's F_2 for five function summation. In addition, it was fundamentally impossible for polynomial approximation to model Schaffer's F_1, which Gaussian Summation Approximation could model. On the other hand, it showed the countertrend for DeJong's F_1, F_3, and F_4.

That may be caused by the complexity difference of problem space landscape. The landscape of DeJong's F_2 and Schaffer's F_1 was so complex that a polynomial with low flexibility could not approximate them. While, the summation of five Gaussian functions had higher flexibility than that of a polynomial. The landscape of DeJong's F_1, F_3, and F_4 were comparatively simple and did not need high function flexibility. Such cases were against Gaussian Summation Approximation with many function parameters to be adjusted.

It may be better to use polynomial approximation if we previously know that the landscape of a problem space is simple. However, many problem spaces in real-world applications have complex landscape, and there are many cases in which we cannot obtain the information on the landscape. Therefore, we conclude that it is better to use Gaussian Summation Approximation consisting of five functions that is applicable to complex problem spaces.

Here, we conclude the discussions above. We confirmed that GA-based Sampling was more effective than Mesh Sampling for considerably small data and that Gaussian Summation Approximation was more effective than polynomial approximation for complex problem spaces. Therefore, we think that the combination of GA-based Sampling and Gaussian Summation Approximation is recommendable for a complex problem space under the severe limitation of available data. Although the computational time of this combination was longer than the others, we can say that such time difference is comparatively slight to the human instruction or evaluation time in operating an interactive learning system.

4 Conclusions and Future Work

This study focused on modeling a problem space under the severe limitation of available data. We proposed a sampling method based on GA search process, GA-based Sampling, and a modeling method based on response surface method using the summation of Gaussian functions, Gaussian Summation Approximation.

We conducted a simulation experiment to evaluate our methods for several problem spaces comparing with

Table 2: Experimental Results for DeJong's F_1 (upper), DeJong's F_2 (middle), and Schaffer's F_1 (lower). Sig. means the results of one-side t-test. ** means that GA-based Sampling was better than Mesh Sampling at 1% significant level. – means that there was no significant difference between them.

DeJong's F_1

Function	Polynomial			1 Gaussian Functions			3 Gaussian Functions			5 Gaussian Functions		
Num. of Data	Sampling		Sig.	Sampling		Sig.	Sampling		Sig.	Sampling		Sig.
	Mesh	GA		Mesh	GA		Mesh	GA		Mesh	GA	
64	7.50	0.00	**	6.54	2.85	**	15.87	4.74	**	17.33	3.99	**
125	0.00	0.00	–	5.27	2.77	–	3.72	3.17	–	15.20	2.55	**
216	0.00	0.00	–	4.60	2.77	–	2.62	2.65	–	3.86	2.25	**

DeJong's F_2

Function	Polynomial			1 Gaussian Functions			3 Gaussian Functions			5 Gaussian Functions		
Num. of Data	Sampling		Sig.	Sampling		Sig.	Sampling		Sig.	Sampling		Sig.
	Mesh	GA		Mesh	GA		Mesh	GA		Mesh	GA	
49	6.25	7.18	–	9.99	8.04	**	5.90	4.53	–	10.11	4.04	**
100	6.01	6.53	–	9.23	7.99	–	3.92	4.10	–	2.47	3.45	–
169	5.90	6.36	–	8.87	7.95	–	3.65	3.75	–	1.75	2.76	–

Schaffer's F_1

Function	1 Gaussian Functions			3 Gaussian Functions			5 Gaussian Functions		
Num. of Data	Sampling		Sig.	Sampling		Sig.	Sampling		Sig.
	Mesh	GA		Mesh	GA		Mesh	GA	
49	2.27	2.96	–	2.30	2.48	–	10.01	2.75	–
100	2.37	2.73	–	0.76	1.66	–	0.61	1.92	–
169	2.29	2.73	–	0.93	1.32	–	0.97	1.36	–

sampling at regular intervals and approximation using a polynomial. As the results, the error between a problem space and its model obtained with the combination of our sampling and modeling methods was smaller than that of the other combinations for a complex problem space when the number of data was severely restricted. We then confirmed the effectiveness of our methods.

We are going to continue experiments for more complex problem spaces and to clarify the relation between the modeling performance and the ratio of sample data size to problem space size. In addition, we are considering a new GA-based Sampling using gradient information and the application of our methods to problem space visualization.

References

[1] K. A. DeJong. *An Analysis of the Behavior of a Class of Genetic Adaptive Systems.* PhD thesis, Department of Computer and Communication Sciences, University of Michigan, 1975.

[2] R. A. Fisher. *The Design of Experiments (8 ed.).* Oliver and Boyd, New York, USA, 1966.

[3] K. Ohnishi H. Takagi, T. Ingu. Accelerating a ga convergence by fitting a single-peak funct ion. *Journal of Japan Society for Fuzzy Theory and Intelligent In formatics*, 15(2):219–229, Apr. 2003.

[4] N. Hayashida and H. Takagi. Acceleration of ec convergence with landscape visualization and human intervention. *Applied Soft Computing*, 1(4F):245–256, 2002.

[5] F. Iida, H. Ayai, and F. Hara. Behavior learning of face robot using human natural instruction. In *Proc. of IEEE Int'l Workshop on Robot and Human Communication (ROMAN'01)*, pages 171–176, Bordeaux/Paris, France. Sept. 2001.

[6] D. Katagami and S. Yamada. Interactive evolutionary robotics from different viewpoints of observation. In *Proc. of IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems (IROS-2002)*, pages 1108–1113. Switzerland. Oct. 2002.

[7] A. Kobsa, J. Koenemann, and W. Pohl. Personalizaed hypermedia presenation techniques for improving online customer relationships. *The Knowledge Engineering Review*, 16(2):111–155, 2001.

[8] J. H. Lee and S. B. Cho. Analysis of direct manipulation in interactive evolutionary computation on fitness landscape. In *Proc. of IEEE Congress on Evolutionary Computation (CEC'02)*, pages 460–465, Honolulu, Hawaii, USA, May 2002.

[9] R. H. Myers and D. C. Montgomery. *Response Surface Methodology: Process and Product Optimization Using Designed Experiments (2nd ed.).* Wiley, New York, USA, 2002.

[10] M. Ohsaki and H. Takagi. Improvement of presenting interface by predicting the evaluation order to reduce the burden of human interactive ec operators. In *Proc. of IEEE Int'l Conf. on System, Man, and Cybernetics (SMC'98)*, pages 1284–1289, San Diego, California, USA, Oct. 1998.

[11] J. D. Schaffer, R. A. Caruana, and L. J. R. Das. A study of control parameters affecting online performance of genetic algorithms for function optimization. In *Proc. of Int'l Conf. on Genetic Algorithms (ICGA'89)*, pages 51–59, Fairfax, Virginia, USA, June 1989.

[12] C. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, pages 379–423 and 623–656, 1948.

[13] H. Takagi. Interactive evolutionary computation: Fusion of the capacities of ec optimization and human evaluation. *Proc. of IEEE*, 89(9):1275–1296, 2001.

[14] M. Yamamoto, T. Hashiyama, and S. Okuma. Reducing computational time on using fitness estimationevolution under the real environment. In *Proc. of IEEE Int'l Conf. on Industrial Electornics, Control, and Instrumentation (IECON'00)*, pages SS47–ECC–3. Nagoya, Japan, Oct. 2000.