Recombination Guidance for Numerical Genetic Programming

Hitoshi IBA[1] Taisuke SATO[2] Hugo de GARIS[3]
iba@etl.go.jp sato@cs.titech.ac.jp degaris@hip.atr.co.jp
1)Machine Inference Section, Electrotechnical Laboratory.
2)Dept. of Computer Science, Tokyo Institute of Technology.
3)ATR Human Information Processing Research Lab.

ABSTRACT

In our earlier papers, we introduced our adaptive program called "STROGANOFF' (i.e. STructured Representation On Genetic Algorithms for NOn-linear Function Fitting), which integrated a multiple regression analysis method and a GA-based search strategy. The effectiveness of STROGANOFF was demonstrated by solving several system identification problems. This paper proposes an "adaptive recombination" mechanism for STROGANOFF. Our intention is to exploit already built structures by "adaptive recombination", in which GP recombination is guided by a certain measure. The effectiveness of our approach is shown by the experiment in predicting a chaotic time series. Thereafter we describe real-world applications of STROGANOFF to computer vision.

1. Introduction

In traditional GP (i.e. Genetic Programming), recombination can cause frequent disruption of building-blocks or mutation can cause abrupt changes in the semantics. To overcome these difficulties, in our earlier paper [Iba93, 94, 95a] we introduced our adaptive program called "STROGANOFF' (i.e. STructured Representation On Genetic Algorithms for NOn-linear Function Fitting)[1], which integrated a multiple regression analysis method and a GA-based search strategy. The effectiveness of STROGANOFF was demonstrated by solving several system identification problems. A tree structure used in STROGANOFF is called a "GMDH" tree, in which terminal nodes are the input (dependent) variables, and non-terminal nodes represent simple polynomial relationships between two descendant (lower) nodes. STROGANOFF consists of two adaptive processes; a) Evolving structured representations using a traditional genetic algorithm, b) Fitting parameters of the nodes with a multiple regression analysis. The advantages of STROGANOFF are summarized as follows:-

1. Analog (i.e. polynomial) expressions complemented the digital (symbolic) semantics. Therefore the representational problem of standard GP does not arise for STROGANOFF.
2. MDL-based fitness evaluation works well for tree structures in STROGANOFF, which controls GP-based tree search.

This paper studies another advantage, i.e. the adaptive recombination mechanism of STROGANOFF. That is,

3. Multiple-regressions tuned the node coefficients so as to guide GP recombination effectively.

Our intention is to exploit already built structures by "adaptive recombination", in which GP recombination is guided by a certain measure. By "adaptive recombination", we mean controlling the application of recombination operators (i.e. mutation or crossover) over the generations, for the sake of improvement of search efficiency, which has been studied by many researchers in string-based GA [Schaffer87,Bäck91,92,93]. The effectiveness of our approach is shown by the experiment in predicting a chaotic time series. Thereafter we describe real-world applications of STROGANOFF to computer vision.

2. MDL-based guidance for a numerical GP

We propose an adaptive recombination for a numerical GP, based on our system-identification approach. For this purpose, we use an MDL (Minimum Description Length)-based fitness evaluation.

[1]STROGANOFF is mainly aimed at solving numerical problems, such as patter recognitions or time series predictions. Thus, we call STROGANOFF a numerical GP.

0-7803-2759-4/95/$4.00

The MDL fitness definition for a tree is defined as follows [Tenorio90]:-

$$MDL = 0.5Nlog S_N^2 + 0.5klog N \quad (1)$$

where N is the number of data pairs, S_N^2 is the mean square error, i.e.

$$S_N^2 = \frac{1}{N}\sum_{i=1}^{N} | \overline{y_i} - y_i |^2 \quad (2)$$

and k is the number of parameters of the tree (see [Iba93] for details). An MDL value involves a tradeoff between certain details of the tree, and the errors. In general, the smaller the MDL value, the better the fitness. Therefore, in order to allow adaptive recombination, we calculate MDL values for all subtrees in a GMDH tree. When applying GP operations, we use these MDL values to decide which subtree will be chosen (i.e. to which subtree a GP operator is applied). We show the effectiveness of this approach with experiments which predict Mackey-Glass time series.

Mackey-Glass time series is generated by integrating the following delay differential equation and is used as a standard benchmark for prediction algorithms.

$$\frac{dx(t)}{dt} = \frac{ax(t-\tau)}{1 + x^{10}(t-\tau)} - bx(t), \quad (3)$$

with a=0.2, b= 0.1 and τ=17, the trajectory is chaotic and lies on an approximately 2.1-dimensional strange attractor. Following Hartman [Hartman91], we experimented in predicting $x(t+85)$ given $x(t)$, $x(t-6)$, $x(t-12)$ and $x(t-18)$ as inputs, and used 500 training and 500 testing patterns.

The STROGANOFF parameters used are shown in Table 1. Fig.1 illustrates an exemplar GMDH tree for this problem, in which the error of fitness ratios (i.e. mean square error, MSE) and MDL values are shown for all subtrees. As can be seen from the figure, the MSE values monotonically decrease towards the root node in a given tree. Thus the root node has the lowest (i.e. best) MSE value. However, the MDL values do not monotonically change. Fig.2 shows the averages of subtree depths whose MDL values are best (i.e. lowest) and worst (i.e. highest). These averages were calculated for 10^5 randomly generated trees. The horizontal axis is the maximum depth of the random trees. The nodes of the worst MDL values are relatively lower, whereas the nodes of the best MDL values are positioned higher. The subtree whose MDL value is lowest is expected to give the best performance of all subtrees. Therefore, it can work as a building-block for crossover operations.

Fig.3 shows the results of experiments in predicting the Mackey-Glass equation. This figure shows

Population Size	120
Crossover Prob.	60%
Mutation Prob.	3.3%
Selection Method	Tournament
Target Variable	$x(t+85)$
Terminal Nodes	$\{x(t),\ x(t-6),$ $x(t-12),\ x(t-18)\}$
Non-terminal Nodes	$\{a_0 + a_1x_1 + a_2x_2$ $+a_3x_1x_2 + a_4x_1^2 + a_5x_2^2\}$
# of Training Data	500
# of Testing Data	500

Table 1 STROGANOFF Parameters (Predicting the Mackey-Glass equation)

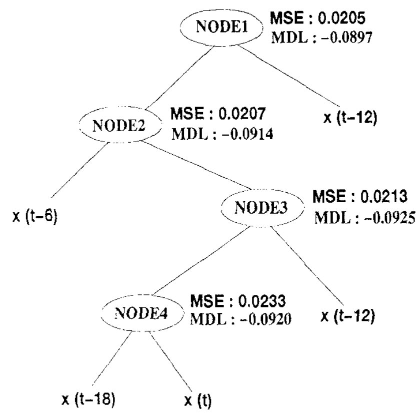

Fig.1:An Exemplar Tree

the best MSE's by generation. For the sake of comparison, STROGANOFF was run under the following conditions:-

1. MDL-based crossover guidance (solid lines). When applying crossover operators to two parents P_1 and P_2, execute the following steps.
   (a) Let W_1 and W_2 be the subtrees with the worst MDL values of P_1 and P_2.
   (b) Let B_1 and B_2 be the subtrees with the best MDL values of P_1 and P_2.
   (c) A new child C_1 is a copy of P_1, in which W_1 is replaced by B_2.
   (d) A new child C_2 is a copy of P_2, in which W_2 is replaced by B_1.
2. Usual crossover (dotted lines). Crossover points are randomly chosen.

Stars and triangles represent MSE values for training data and test data respectively. This figure clearly shows the improvement of search by means of crossover guidance not only for training and but

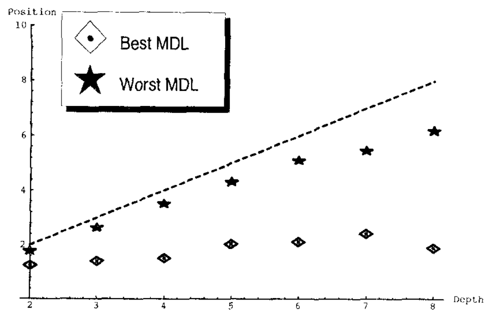

Fig.2:Average Depths of Node of Best and Worst MDL values

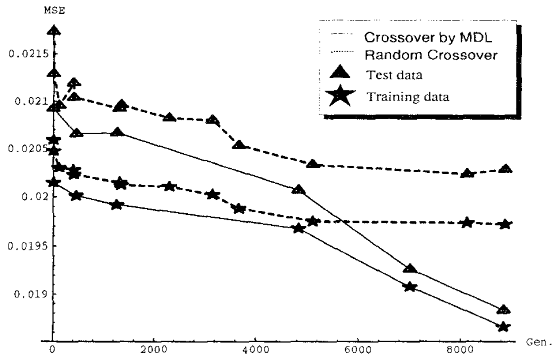

Fig.3:Prediction of Mackey-Glass Equation

also for test data. Thus we have confirmed the effectiveness of crossover guidance based on MDL values.

With the above experimental results, we propose a type of adaptive recombination based on MDL values. For this purpose, in applying crossover or mutation operators, we follow the rules described below:-

1. Apply a mutation operator to a subtree whose MDL value is larger.
2. Apply a crossover operator to a subtree whose MDL value is larger.
3. When 2 is done, get a subtree whose MDL value is smaller from another parent.

3. Applications to Computer Vision

To show the effectiveness of our approach, we have made real-world applications of STROGANOFF to computer vision. Because of the limitation of space, we briefly describe two experiments. The other applications, such as recognition of simple textures and more complex patterns, have been reported and discussed in [Iba94b].

3.1. Predicting a Face Image

This experiment is to predict the center pixel of a 3×3 region given the values of the other 8 pixels. The reason for predicting the center is that STROGANOFF will form a causal nonlinear two-dimensional filter. This experiment is related to the previous research [Sanger91], in which four 128×128 8 bit images of human faces under similar lighting conditions were used and radial basis functions were trained and tested for approximation.

We used the parameters shown in Table 2. The terminal variables are eight pixel values (normalized range between 0 and 1) around the center pixel (output variable, i.e. y) (see Fig.4). Fig.5(a) shows a face image used for training data (100×100 pixels). Fig.5(b), (d) and (f) show images predicted from 8 surrounding pixels at generations of 0, 10 and 58 respectively. Fig.5(c), (e) and (g) are those errors. As can be seen, the prediction is becoming more precise over the generations. Since the reconstructed images tend to be indistinguishable to human observers, we think the performance of our STROGANOFF is relatively high compared to traditional vision approaches.

Population Size	120
Crossover Prob.	60%
Mutation Prob.	3.3%
Selection Method	Tournament
Target Variable	Center Pixel of 3×3 Region
Terminal Nodes	$\{x_1, x_2, \cdots, x_8\}$
Non-terminal Nodes	$\{a_0 + a_1 z_1 + a_2 z_2$ $+ a_3 z_1 z_2 + a_4 z_1^2 + a_5 z_2^2\}$
# of Training Data	100×100 pixels (256 gray scale)

Table 2 STROGANOFF Parameters (Predicting central pixels)

3.2. Classification of Window Patterns

The second experiment involves recognition of more complex patterns. We chose the STROGANOFF parameters shown in Table 3. Terminal nodes are filtering operators commonly used in computer vision. For instance, x_3 returns a pixel value of the resultant image filtered by the Sobel operator [Duda73]. We used a "Hursley House" picture of 512×512 pixels (see Fig.6(a)). Fig.6(b) and (c) show positive data (i.e. front door image, 25×60 pixels) and negative data (130×60 pixels). The goal of this problem is to classify window and door patterns. The reason for classifying these patterns is that STROGANOFF will form a causal nonlinear two-dimensional filter, which might be regarded as

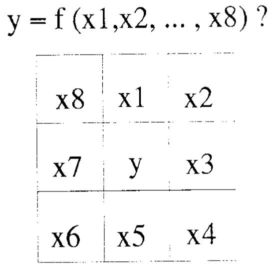

Fig.4:3×3 Region

Fig.5(a):Source

Fig.5(b):Prediction (Generation 0)

Fig.5(c):Error (Generation 0)

Fig.5(d):Prediction (Generation 10)

Fig.5(e):Error (Generation 10)

Fig.5(f):Prediction (Generation 58)

Fig.5(g):Error (Generation 58)

a "window function" or a "door function". This is because doors and windows have many characteristics (i.e. geometric invariants) in common. For instance, as can be seen in Fig.6(a), small black squares are regularly lined for window areas as well as the front door area. Fig.6(d) shows the result of the experiment. The black pixels are classified as positive examples (i.e. windows and doors). This figure was post-processed through a smoothing filter for the purpose of noise reduction. The acquired tree recognized 19 window areas out of about 34 windows and doors. Although further post-processing is required for more precise recognition, we can regard the acquired tree as a sort of "window function". Therefore, we think this approach leads to the realization of a learning vision system, which combines various vision operators adaptively for the sake of pattern segmentation.

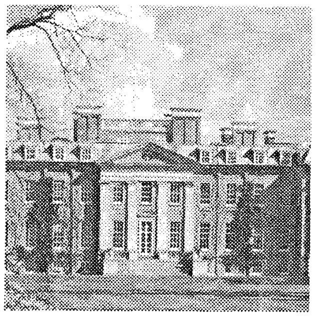

Fig.6(a):Hursley House Image(Source)

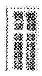

Fig.6(b):Positive Example

Fig.6(c):Negative Example

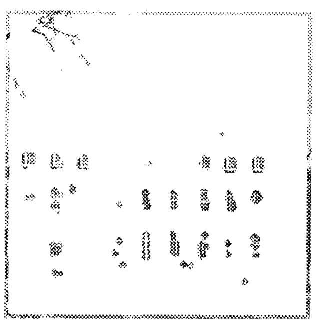

Fig.6(d):Result

Population Size		120
Crossover Prob.		60%
Mutation Prob.		3.3%
Selection Method		Tournament
Target Variable	0	Negative Example
	1	Positive Example
Terminal Nodes	x_0	Pixel Value (256 gray scale)
	x_1	First Differential, 3 × 3 region
	x_2	Laplacian (total differential), 3 × 3 region
	x_3	Sobel (total differential), 3 × 3 region
	x_4	Maximum of Pixel Values, 5 × 5 region
	x_5	Minimum of Pixel Values, 5 × 5 region
	x_6	Average of Pixel Values, 5 × 5 region
	x_7	Std. of Pixel Value, 5 × 5 region
Non-terminal Nodes		$\{a_0 + a_1 z_1 + a_2 z_2 + a_3 z_1 z_2 + a_4 z_1^2 + a_5 z_2^2\}$
# of Training Data		Negative Example (30 × 30 pixels) Positive Example (30 × 30 pixels)
# of Testing Data		256 × 256 pixels (256 gray scale)

Table 3 STROGANOFF Parameters (Classification)

4. Discussions

4.1. Recombination Guidance for GP

This paper proposed recombination guidance mechanism for a numerical GP and showed the effectiveness by experiments. We think that GP is to be studied in the same point of view of the previous study on GA-search improvement. Our goal was to exploit already built structures by "adaptive recombination", in which GP recombination was guided by a measure called MDL.

Our proposed concept is similar to the block fitness function proposed by [Rosca94]. Rosca used this measure to discover a useful building block by his system AR-GP. When AR-GP solved a parity problem, he defined the block fitness function to be the same as the usual fitness function (i.e. hits), except that hits were evaluated on a subset of the set of fitness cases, determined by fixing the values of variables not used in the block to arbitrary values. This is equivalent to taking simpler (shorter) parity examples.

We believe that our approach can also realize an efficient search strategy for traditional GP (i.e.

symbolic problems), provided that a certain form of MDL-like value is well-defined for GP trees. Our recent results have shown the effectiveness of this method in various symbolic problems [Iba95b]. We are currently working on this topic further.

4.2. Applications of STROGANOFF to Computer Vision

We have shown the effectiveness of STROGANOFF with applications to computer vision. We have not yet conducted comparative studies of STROGANOFF with other techniques. Since comparison with other computer vision methods is a key to elucidate the performance of STROGANOFF, we are currently working on that topic. The important decision factors with the above applications are as follows:-

1. the choice of negative examples
2. the choice of filters (terminal variables)
3. pre- and post-processing

Negative examples are usually chosen from the background of the images. For instance, in the third example, we chose the image of the ground for the negative examples. But there are many alternatives for the negative examples. In the extreme case, no negative examples are given in the training phase. It is well known that the classification performance is dependent not only on positive data but also on negative data. Therefore, the choice of negative data from the training picture is very important. We are now researching the influence of negative examples on STROGANOFF performance.

We used 7 filters in the set of terminal variables (see Table 3). These are well-known filters in the computer vision literature. The choice of filters is essential for classification. For instance, when classifying pixels belonging to the sky area from a picture, we found that the absolute coordinate values (x, y) of the picture is most important. This is because the sky usually occupies the upper part of the picture, i.e. the vertical coordinates are higher. We believe the essential variables are selected through the evolution of STROGANOFF trees.

Pre- and post-processing is usually required for image processing. For instance, in the third experiment, the final figure was post-processed through a smoothing filter. However, processes are applied heuristically at the moment. Therefore, we are now trying to make an adaptive system, which integrates pre-processing, GP-based processing, and post-processing.

5. Conclusion

This paper introduced an "adaptive recombination" mechanism for numerical GP. Our goal was to exploit already built structures by "adaptive recombination", in which GP recombination is guided by a certain measure. The effectiveness of our approach has been shown by the experiment in predicting a chaotic time series, and real-world applications of STROGANOFF to computer vision.

References

[Bäck91] Bäck, T., Self-Adaptation in Genetic Algorithms, In *Proc. of the First European Conference on Artificial Life*, MIT Press, 1991

[Bäck92] Bäck, T., The Interaction of Mutation Rate, Selection, and Self-Adaptation within a Genetic Algorithm, In *Parallel Problem Solving from Nature 2*, North-Holland, 1992

[Bäck93] Bäck, T., Optimal Mutation Rates in Genetic Search, In *Proc. of 5th ICGA*, 1993

[Duda *et al.*73] Duda, R.O. and Hart, P.E., Pattern Classification and Scene Analysis, Wiley, 1973

[Hartman *et al.*91] Hartman, E. and Keeler, J.D., Predicting the Future: Advantages of Semilocal Units, in *Neural Computation*, vol.3, 1991

[Iba *et al.*93] Iba, H., Kurita, T., deGaris, H. and Sato, T. System Identification using Structured Genetic Algorithms, in *Proc. of 5th International Conference on Genetic Algorithms*, 1993

[Iba *et al.*94a] Iba, H., deGaris, H. and Sato, T. Genetic Programming using a Minimum Description Length Principle, in *Advances in Genetic Programming*, (ed. Kenneth E. Kinnear, Jr.), MIT Press, 1994

[Iba *et al.*94b] Iba, H. and Sato, T. Adaptive Recombination for Genetic Programming and its Application to Computer Vision, ETL-TR-94-20, 1994

[Iba *et al.*95a] Iba, H., deGaris, H. and Sato, T. Temporal Data Processing Using Genetic Programming, in *Proc. of 6th International Conference on Genetic Algorithms*, 1995

[Iba *et al.*95b] Iba, H. and deGaris, H. Genetic Programming Extended by Recombination Guidance, ETL-TR-95-15, also in GP workshop (ICGA95), 1995

[Rosca94] Rosca,J., Hierarchical Self-Organization in Genetic Programming, in *Proc. of the 11st International Conference on Machine Learning*, 1994

[Schaffer *et al.*87] Schaffer, J. D. and Morishima, A., An Adaptive Crossover Distribution Mechanism for Genetic Algorithms, in *Proc. of 2nd International Conference on Genetic Algorithms*, 1987

[Sanger91] Sanger, T.D., A Tree-Structured Adaptive Network for Function Approximation in High-Dimensional Spaces, in *IEEE Tr. on Neural Networks*, vol.2, no.2, March 1991