Binary SOM Based on Significance of Inputs and its Application to Reproduction of GA

Ryosuke Kubota[†], **Keiichi Horio**[‡] **and Takeshi Yamakawa**[‡][*]
[†]Graduate School of Computer Science and Systems Engineering,
Kyushu Institute of Technology
680-4 Kawazu, Iizuka-shi, Fukuoka, 820-8502, Japan
kubota-ryosuke@edu.brain.kyutech.ac.jp
[‡]Graduate School of Life Science and Systems Engineering,
Kyushu Institute of Technology
2-4 Hibikino, Wakamatsu-ku, Kitakyushu-shi, Fukuoka, 808-0196, Japan
{horio,yamakawa}@brain.kyutech.ac.jp

Abstract - *SOM achieves a probability density distribution approximation of input vectors using weight vectors. Binary SOM in which each element of the input vectors and the weight vectors is represented by "binary" integer is same to the SOM with respect to the approximation. In this paper, we propose a new updating equation of the binary weight vectors based on significance of binary input vectors. In the proposed updating equation, the binary weight vectors with low significance factors are adjusted to the binary input vectors with high significance factors. This updating equation achieves a probability density distribution approximation of the significant binary input vectors. The modified Binary SOM employing the proposed updating equation is applied to GA as a reproduction operator in order to maintain a genetic diversity of population.*

Key words - self-organizing map, significance of input vector, genetic algorithm, reproduction, fitness value

1 Introduction

Self-organizing maps (SOMs)[1] can achieve a precise probability density distribution approximation of a set of input vectors using weight vectors. In addition, Binary SOM in which each element of the input and the weight vectors is represented by "binary" integer can do the same approximation[2]. Hence, the SOMs with their related extensions are applied to many kinds of applications[3]. In some applications, the significancy of input vectors needs to be considered. In such cases, ordinary SOMs are ineffective, because the significance factors of all input vectors are evaluated equally. Here, we define significance factor as a measure of profit, utility and goodness of problem application fitness. To approximate the probability density distribution of the significant input vectors in a set of all inputs appropriately, the

[*]The author is also with the Fuzzy Logic Systems Institute (FLSI), Kawazu, Iizuka-shi 820-0067 Japan.

weight vectors are updated by considering the significance of the input vectors. Here, we define significant input vector as the input vector with high significance factor.

In this paper, we propose a new updating equation by adding new coefficients with respect to the significance factors of binary input vectors. In the proposed updating equation, the binary weight vectors with the low significance factors are updated toward the binary input vectors with the high significance factors. This updating equation achieves the appropriately probability density distribution approximation of the input vectors with high significance factors in a set of all inputs.

The modified Binary SOM by employing the proposed updating equation is applied to genetic algorithm (GA)[4] as a reproduction operator, to maintain a high genetic diversity of population (set of chromosomes). The traditional reproduction operators based on roulette wheel selection or tournament selection may lose the genetic diversity of the population in the early stage[5], because it can not generate new chromosomes which are different from the present chromosomes. In the GA, the genetic diversity represents the number of searching points in the present generation, and the high genetic diversity is preferable. In the proposed reproduction operator using the modified Binary SOM, the input vectors and their significance correspond to the chromosomes and fitness values in the GA, respectively. The set of new chromosomes of next generation is decided by learning of the modified Binary SOM. This learning facilitates preservation of the high genetic diversity and the effective search, because the proposed updating equation can achieve the probability density distribution approximation of the high significant vectors in a set of all inputs. The effectiveness of the reproduction using the modified Binary SOM is verified by applying it to a combinatorial optimization problem.

2 Binary SOM Based on Significance of Binary Inputs

2.1 Ordinary Binary SOM

In the ordinary SOM, real values are used for weight vectors. The learning process of the SOM using real-valued weight vectors is achieved as follows.

Step1 The real-valued input vector $\boldsymbol{x} = (x_1, x_2, \cdots, x_M)$ is applied to the input layer, where M represents number of elements.

Step2 The Euclidean distance $E(\boldsymbol{x}, \boldsymbol{w}_j)$ between the input vector $\boldsymbol{x}$ and the weight vector $\boldsymbol{w}_j = (w_{j1}, w_{j2}, \cdots, w_{jM})$ of the j-th unit in the competitive layer is calculated by:

$$E(\boldsymbol{x}, \boldsymbol{w}_j) = \sqrt{\sum_{i=1}^{M}(x_i - w_{ji})^2}. \tag{1}$$

Step3 The unit which has the minimum Euclidean distance is defined as the winner unit c by:

$$c = \arg\min_j\{E(\boldsymbol{x}, \boldsymbol{w}_j)\}. \tag{2}$$

Step4 The weight vectors of the winner unit and its neighboring units are updated by:

$$\boldsymbol{w}_j(t+1) = \boldsymbol{w}_j(t) + \alpha_r(t)(\boldsymbol{x} - \boldsymbol{w}_j(t)), \tag{3}$$

where $\alpha_r(t)$ is defined as learning ratio, and decreases monotonously.

In ordinary Binary SOM, each element of the weight vectors $\boldsymbol{w}$ and the input vector $\boldsymbol{x}$ which is applied to the network is represented as binary integer “0” and “1”. When the distance between the input vector $\boldsymbol{x}$ and the weight vector $\boldsymbol{w}_j$ of the j-th unit in the competitive layer is calculated, Eq. (1) is replaced with:

$$H(\boldsymbol{x}, \boldsymbol{w}_j) = \mathit{bitcount}\{(\bar{x_i} \wedge w_{ji}) \vee (x_i \wedge \bar{w_{ji}}) \mid i = 1, \cdots, M\}. \tag{4}$$

Eq. (1) is called the Hamming distance. Concomitantly, Eq. (2) is also replaced with:

$$c = \arg\min_j \{H(\boldsymbol{x}, \boldsymbol{w}_j)\}. \tag{5}$$

Moreover in updating of weight vectors of the ordinary Binary SOM, exclusive-OR (XOR) of each element of $\boldsymbol{x}$ and $\boldsymbol{w}_j(t)$ is computed. If XOR of x_i and $w_{ji}(t)$ is “1”, $w_{ji}(t)$ is inverted. The number of inverting bits is defined as a learning ratio, $\alpha_b(t)$. $\alpha_b(t)$ gradually decreases as learning progresses.

The learning process of the SOM facilitates the probability density distribution approximation of the real-valued and the binary input vectors.

2.2 Modified Updating Based on Significance of Binary Inputs

In some applications, the significancy of input vectors needs to be considered. In such cases, ordinary SOMs are ineffective, because the significance factors of all input vectors are evaluated equally. The significance factor represents a measure of profit, utility and goodness of problem application fitness. In the ordinary Binary SOM (including the ordinary SOM), when the significance of all binary input vectors are same, the probability density distribution of the binary input vectors are approximated. However, the ordinary Binary SOM can not approximate the probability density distribution of the binary input vectors appropriately, when the binary input vectors have their significance factors, because significance factors of all binary input vectors are evaluated equally. To approximate the probability density distribution of the input vectors with high significance in a set of all binary inputs appropriately, the weight vectors must be updated by considering the significance of the binary input vectors.

In this paper, we propose a new updating equation by adding new coefficients with respect to the significance factors of binary input vectors. The proposed updating equation is defined by:

$$\boldsymbol{w}_j(t+1) = \boldsymbol{w}_j(t) + \lfloor \alpha_s \cdot I_{\boldsymbol{x}} \cdot h(I_{\boldsymbol{x}}, d_j) \cdot (1 - I_{\boldsymbol{w}_j(t)}) \rfloor (\boldsymbol{x} - \boldsymbol{w}_j(t)), \tag{6}$$

where α_s, $I_{\boldsymbol{x}}$, $I_{\boldsymbol{w}_j(t)}$ and $\lfloor\ \rfloor$ are learning ratio, significance factors of the binary input vectors, those of the binary weight vectors and Gauss' notation, respectively. The significance factor is normalized from 0 to 1. d_j is the distance between the j-th unit and the winner unit in the competitive layer. $h(I_{\boldsymbol{x}}, d_j)$ is a neighboring coefficient and is represented by:

$$h(I_{\boldsymbol{x}}, d_j) = \exp\left(-\frac{{d_j}^2}{2{I_{\boldsymbol{x}}}^2}\right). \tag{7}$$

In Eq. (6), $\lfloor \alpha_s \cdot I_{\boldsymbol{x}} \cdot h(I_{\boldsymbol{x}}, d_j) \cdot (1 - I_{\boldsymbol{w}_j(t)}) \rfloor$ is corresponding to the learning ratio $\alpha_b(t)$ which represents the number of inverting bits. Specifically, a lot of elements of the binary weight vector are adjusted to those of the binary input vector, when $I_{\boldsymbol{x}}$ is large and $I_{\boldsymbol{w}_j(t)}$ is small. Adversely, almost all the elements of the binary weight vector are not adjusted to those of the binary input vector, when $I_{\boldsymbol{x}}$ is small and $I_{\boldsymbol{w}_j(t)}$ is large. Eq. (7) means that the neighboring area is wide, when $I_{\boldsymbol{x}}$ is large. This gradual neighborhood function is utilized to avoid the convergence of weight vectors in narrow area. Eq. (6) facilitates that the binary weight vectors with low significance factors are adjusted to the binary input vectors with high significance factors. The learning by employing this updating equation can achieve the approximation of the probability density distribution of the binary input vectors with high significance factors in a set of all inputs.

3 Reproduction Operator using Modified Binary SOM

GA is a search algorithm based on the mechanics of natural selection and natural genetics. They combine "survival of the fittest" among string structures and randomized information exchange to form a search algorithm with some of the innovative flair of human search. This mechanic are surprisingly simple, involving nothing more complex than copying strings (chromosomes) and swapping partial strings.

The searching process of GA is achieved by iterating reproduction, crossover and mutation operators. Reproduction is a process in which chromosomes are copied according to their fitness function values f. Copying chromosomes according to their fitness values means that chromosomes with a higher value have a higher probability of contributing one or more offsprings in the next generation. However, occasionally the traditional reproduction operators may become overzealous and lose genetic diversity of population, because the chromosomes of the next generation are generated by copying those of the present generation. In other words, the traditional reproduction operators can not generate new chromosomes which are different from the present chromosomes. The lost of genetic diversity leads a lack of some potentially useful genetic material. Fig. 1(a) shows the present population before reproduction. Fig. 1(b) shows the population after the traditional reproduction. Notice that the genetic diversity of the population decreases. In Fig. 1(b), it appears that a population size has got smaller because this phenomenon results from the fact that some chromosomes take on

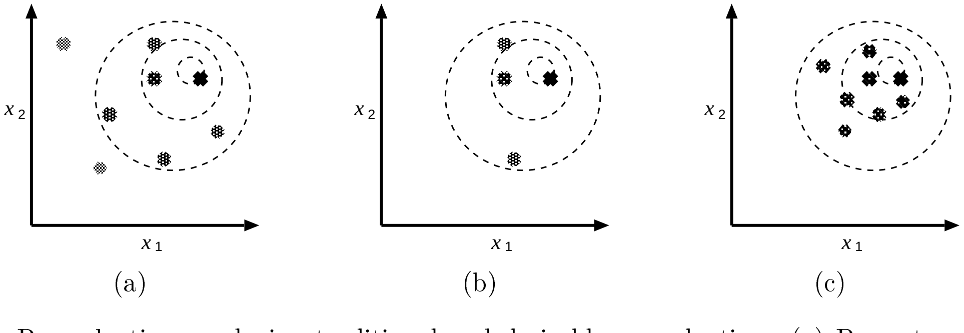


Figure 1: Reproduction employing traditional and desirable reproduction. (a) Present population. (b) Population after traditional reproduction. (c) Population after desirable reproduction. Each axis and dashed circles represent the element of chromosomes and contours of fitness value, respectively. Area in the circle with smaller radius means that fitness values is higher.

identical string. However, the population size is exactly the same from beginning to end of iteration. The chromosomes are the searching points in the present generation, and they are also base points of crossover in the next generation. Thus, the decrease of genetic diversity leads to ineffectiveness of the search. The mutation with high probability can contain the decrease of genetic diversity. However, it is not effective because the fitness values of new chromosomes which are produced by mutation should not be high. To achieve the effective search, the reproduction operator approximating the probability density distribution of the chromosomes with high fitness values as shown in Fig. 1(c) is preferable.
In this paper, to verify the validity of the proposed updating equation, the modified Binary SOM with proposed updating equation is applied to the reproduction operator of the GA. In the modified Binary SOM, the binary input vectors and their significance correspond to the chromosomes and their fitness values of the GA, respectively. The procedure of the GA including this operator is summarized as follows.

Step0 All chromosomes are initialized using random values (bits).

Step1 Chromosomes of the next generation are selected from those of the present generation using proposed reproduction operator.

Step1-0 Weight vectors are initialized by learning of an ordinary Binary SOM, in which randomly generated vectors are used as input vectors. Chromosomes in the present population are kept as a set of input vectors.

Step1-1 One chromosome is selected from the present population. It corresponds to the input vector and applied to the input layer.

Step1-2 The Hamming distance $H(\boldsymbol{x}, \boldsymbol{w}_j)$ is calculated, and the unit which has the minimum distance is defined as the winner unit.

Step1-3 The weight vectors of all units in the competitive layer are updated by Eq. (6).

Step1-4 Step1-1 to Step1-3 are repeated until that all chromosomes are selected. The weight vectors after updating in the Step1-3 are used as the chromosomes after the reproduction.

Step2 New chromosomes are produced from the randomly-selected parents by using the crossover and mutation operators. The crossover and mutation operators are traditional ones in GAs.

Step3 Step1 to Step5 are repeated until the given problem are solved.

This partial update in Step1-3 can generate new chromosomes which are different from the present chromosomes based on “the distribution of the probability density of the chromosomes with high fitness values” in the GA. The population with high fitness values can maintain the high genetic diversity and achieve an effective search.

4 Simulations

To verify the effectiveness of the proposed method, a normal knapsack problem (one of combinatorial optimization problems) is solved. In the normal knapsack problem, the objective is to maximize the total commercial value of the objects placed in a sack under constraint of maximum physical weight. The knapsack problem of 20 subjects is defined by:

$$\textbf{Problem:}\ \text{Maximize} \sum_{k=1}^{20} v_k x_k, \text{ where } x_k \in \{0,1\}, \text{ subject to } \sum_{k=1}^{20} v_k u_k \leq L.$$

where x_k is "1" or "0", if the k-th subject is put into the sack or not, respectively. v_k and u_k represent the commercial value and physical weight of the k-th subject, respectively. L stands for the maximum physical weight characterized by the sack. The commercial value and physical weight of each subject in this simulation are shown in Table 1.

The parameter values for the GA are (1)population size $N = 9$, (2)crossover probability $P_c = 0.3$, (3)mutation probability $P_m = 0.01$ and (4)learning ratio $\alpha_s = 10$. One-point crossover and uniform mutation are adopted. In this simulation, the GA employing the proposed reproduction operator is compared with that of employing the roulette wheel selection and tournament selection.

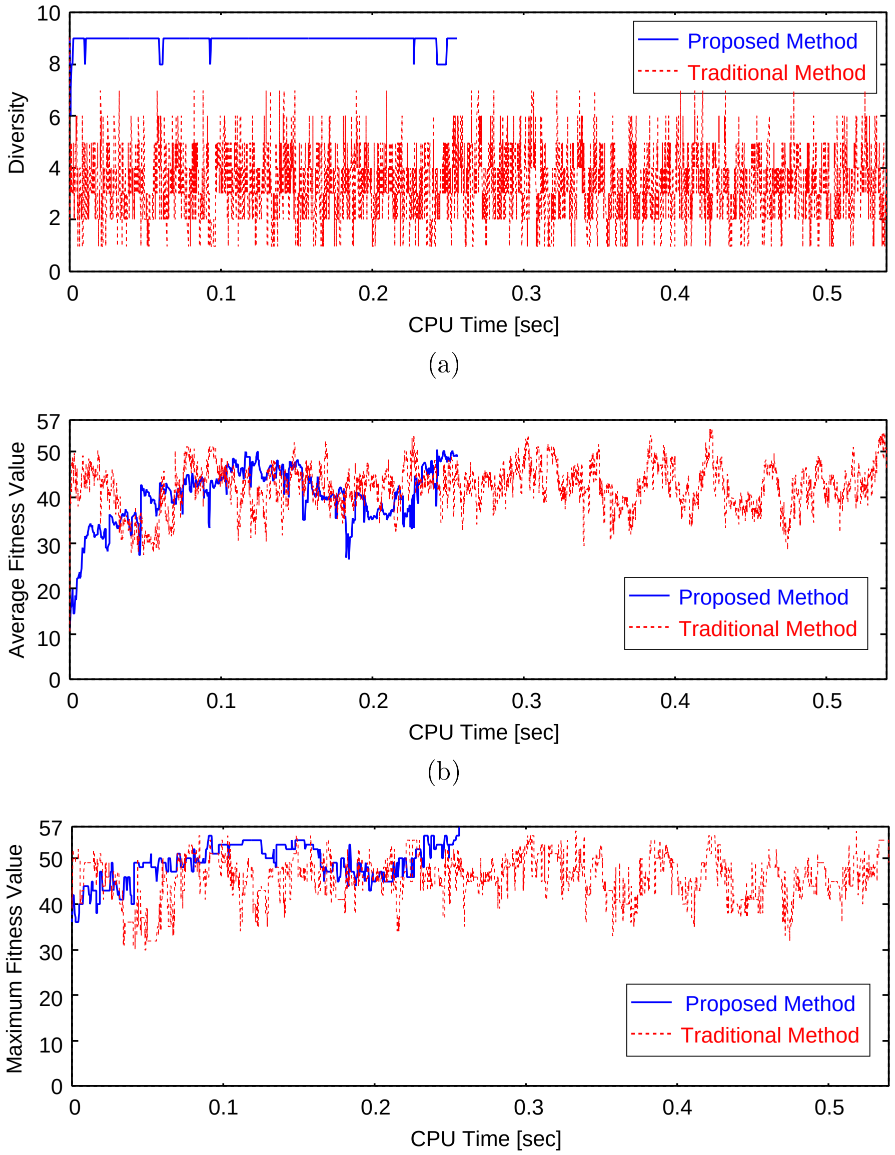
Table 2 shows the comparison of the number of the generations and CPU time. Ten runs are executed and the average value is calculated to suppress dependencies on all stochastic effects. The results show the proposed reproduction operator enables the better search than the roulette wheel selection and tournament selection with respect to the number of the generations and CPU time. Fig. 2 shows the transition of the fitness value and genetic diversity using proposed reproduction operator and roulette wheel selection operator. Fig. 2(a) shows the genetic diversity, which represents the number of the searching points in one generation, where it does not decrease even if the generation progresses in the proposed reproduction. The transition of the average fitness values of both GA are almost same as shown in Fig. 2(b). These results show that the proposed reproduction operator approximates the probability density distribution of the chromosomes with high fitness values in more detail comparing with the roulette wheel selection. The population with high genetic diversity and high average fitness values guarantees the high performance of the crossover operator. This increases the fitness value of the population as shown in Fig. 2(c). Hence, the GA employing the proposed reproduction operator can achieve the effective search comparing with the traditional reproduction operators.

Table 1: Commercial value and physical weight of each subject.

Subject No.	1	2	3	4	5	6	7	8	9	10
Commercial value	8	3	9	2	3	8	2	9	2	2
Physical weight	5	8	2	3	6	1	3	9	5	7
Subject No.	11	12	13	14	15	16	17	18	19	20
Commercial value	3	4	5	2	5	7	2	3	4	8
Physical weight	4	6	4	3	8	9	2	4	1	12

Table 2: Comparison of the number of the generations and CPU Time.

	# of generations	CPU Time
Proposed Method	673	0.269 *sec.*
Roulette Wheel Selection	5444	0.544 *sec.*
Tournament Selection	11087	1.109 *sec.*



(a)

(b)

(c)

Figure 2: Transition of fitness values and genetic diversity using proposed reproduction and roulette wheel selection. (a) Genetic diversity. (b) Average fitness value of the population. (c) Maximum fitness value of the population. These results show that the proposed reproduction operator maintains the high genetic diversity of the population and facilitates the effective search.

5 Conclusions

In this paper, we proposed a new updating equation by adding new coefficients with respect to the significance factors of the binary input vectors. The proposed updating equation achieves the probability density distribution approximation of the high significant vectors in a set of all binary inputs. Besides, to maintain the genetic diversity of the population, the modified Binary SOM employing the proposed updating equation are applied to the GA as the reproduction operator. The proposed reproduction operator realized the preserving the high genetic diversity and the faster search comparing with the traditional reproduction operators. This proposed reproduction overcame the drawbacks of the traditional ones by realizing the generation of the chromosomes with different strings of high fitness values. The results of the test problems verified the validity and the effectiveness of proposed updating equation and modified learning of the Binary SOM.

Acknowledgment

This work was supported (in part) by the 21st Century Center of Excellence Program, "World of Brain Computing Interwoven out of Animals and Robot (PI: T. Yamakawa)" granted in 2003 to Department of Brain Science and Engineering, (Graduate School of Life Science and Systems Engineering), Kyushu Institute of Technology by Japan Ministry of Education, Culture, Sports, Science and Technology and was also supported by Grant-in-Aid for Scientific Research (A), 2002, 14205038.

References

[1] T. Kohonen (1982), Self-organized formation of topologically correct feature maps, *Biological Cybernetics*, **vol. 43**, pp. 59-69.

[2] T. Yamakawa, K. Horio and T. Hiratsuka (2002), Advanced Self-Organizing Maps using binary weight vector and its digital hardware design, *Proc. of The 9th International Conference on Neural Information Processing*, **vol. 3**, pp. 1330-1335.

[3] S. Kaski, J. Kangas and T. Kohonen (1998), Bibliography of Self-Organizing Map (SOM) Papers:1981-1997, *Neural Computing Surveys 1*, pp. 102-350.

[4] D. E. Goldberg (1989), *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley Publishing Company Inc..

[5] I. Ono and S. Kobayashi(1997), A real-coded genetic algorithm for function optimization using unimodal normal distribution crossover, *Proc. 7-th International Conference on Genetic Algorithms*, pp. 246-253.