An Algorithm of SOM using Simulated Annealing in the Batch Update Phase for Sequence Analysis

Hiroshi Dozono, Hisao Tokushima, Shigeomi Hara, Yoshio Noguchi
Faculty of Science and Engineering Saga University
Saga JAPAN
hiro@dna.ec.saga-u.ac.jp, tokusima@dna.ec.saga-u.ac.jp,
hara@dna.ec.saga-u.ac.jp, nogu@dna.ec.saga-u.ac.jp

Abstract - *An algorithm of Self-Organizing Maps (SOM) that can extract features of DNA sequences is introduced here. DNA sequences are considered to have characteristic features, depending on regions the sequences are taken from or functions of proteins translated from them. If hidden features of DNA can be extracted from sequence alone, they can be used for predicting regions or functions of unknown sequences. This group developed algorithms that organize sequences of specific lengths using SOM. This algorithm can select a smaller number of sequences from all combinations of nucleotides for a given length. Sequences are organized on 2-dimensional maps according to similarity. A batch SOM algorithm that updates the map using a simulated annealing method to organize adjacent sequences more closely on the map is described. Analysis of DNA sequences in terms of function of the translated proteins was performed and results are presented.*

Key words - Self Organizing Map, Bio-informatics, Sequence analysis, Simulated annealing

1 Introduction

Recently, entire genome sequences for some species have been determined. Many functional studies of genes have been performed using both biological experiments and analysis of similar sequences; however, new methods are required to hasten the determination of functions of all genes in a genome. Furthermore, not all regions (regulatory, promoter, exon, intron, etc.) on genome sequences have been characterized yet. To determine the function of genes or regions on genome sequences, target sequences are compared with known sequences using alignment algorithms. However, the computing power required for calculating alignment between a target sequence and many known sequences is very large. Thus, a method that can efficiently correlate a target sequence with a large number of known sequences would be very useful.
For this purpose, this group developed the SOM algorithm that extracts features of DNA sequences as sets of DNA probes, comprised of short DNA sequences of speci ed length. Hamming distance is used as the similarity measure between sequences, thus sequences with smaller changes in nucleotides in the same position are organized closer on the map. Using this algorithm, a smaller number of probes that represent representative features of the reference sequence are selected on the map.

Application of this algorithm for design of DNA chips has previously been reported [1][2]. DNA chips are powerful tools for sequence analysis that can be used for analysis of gene expression, analysis of Single Nucleotide Polymorphism (SNPs) and sequencing by hybridization. DNA chips comprised of longer probes have higher resolutions for sequencing by hybridization and SNP analysis. However, the number of probes increases exponentially with length of the probe using all combinations of nucleotides of a speci ed length. To determine a 1000 bp DNA sequence, a DNA chip comprising probes longer than 8 kbp is desirable [3], but the size may be larger without any design, as it requires $4^8 = 65536$ probes to be comprehensive. Therefore, design of a DNA chip is important and SOM algorithms have worked very well for this purpose. Adequate probes are selected using this method by considering the number of selected probes, covering rate of the reference sequence and expected rate of SNP detection using probes selected by SOM.

Maps organized by SOM can be applied to sequence analysis. SOM can also be applied to analysis of the relationship between sequence and function [4]. In this report, a modi ed algorithm for SOM, suitable for analysis of DNA sequence based on the sequence data itself is described.

Analysis of DNA sequences by mapping sequences with speci c functions and sequences taken from speci c regions on a map organized by SOM was also performed. Mapped sequences show their own patterns, but the patterns become narrower at continuous regions, depending on the function of the translated genes because adjacent probes on target sequences are mapped in fragments on the map. If target sequences are mapped continuously, sequences with common features (e.g. sequences with similar function, sequences from the same species and sequences taken from similar regions) should be mapped in large clusters consisting of adjacent probes with common features on the target sequences. For this purpose, the probes and their shifted valiations should be mappped closely on the map and SOM algorithm is modi ed so as to map sequences using a similarity measure that takes into account not only changes in nucleotides but also the extent of shifting of probes. From this modi cation, probes that match target sequence adjacently tend to be organized closely on the map. Each region on the map extracts a feature of each domain on the reference sequence, thus the target sequences are mapped adequately, depending on their relationship between each other for the purpose of comparison of themselves.

Furthermore, the method for updating a map was changed from an incremental method to a batch method. Using the batch update method, updates of units can be directed so as to organize adjacent probes closely on the map. Here, a simulated annealing method to direct organization of probes during the update phase is introduced.

Experiments organizing maps were performed using reference sequences obtained from genome databases, categorized by species and gene function. Relationship between sequences used for training was also examined by mapping sequences on an organized map.

2 Feature Mapping of DNA sequence by SOM

In conventional sequence analysis, sequences are mainly analyzed by using 1-dimensional information. For example, motifs of known features are used for nding speci c regions of sequences, alignment of known sequences and target sequences are carried out to identify functions of target sequences, with a hidden Markov model used as the stochastic model

of the 1-dimensional sequence. The relationship between DNA sequences is more readily classi able if mapped appropriately onto a 2-dimensional plane.
For this purpose, this group chose the SOM algorithm as it can organize generic features of DNA sequences by sufficient learning of known DNA sequences on a 2-dimensional plane. An algorithm that trains self organizing maps of probes that are a vector of discrete values for 'A','C','G' and 'T' was also used by this group [2].

3 Batch SOM algorithm for the learning of DNA sequences

Experiments were performed by applying the SOM algorithm for selecting sets of probes that represent features of DNA sequences [1,2]. Number of probes was reduced markedly from a large starting set; however, this reduced number of probes could detect almost all nucleotide changes able to be detected by all combinations of a probe of the same length. However, efficiency of the algorithm was not good in terms of computation time and number of distinct probes organized on the map.
Additionally, if consecutively hybridized probes are arranged adjacently on a map, sequences will be mapped as continuous regions on organized maps and similar sequences will be mapped closely on the map.
For this purpose, two modi cations to the algorithm were made and results reported [3]. At rst, the search method was changed to winner units. In conventional SOM, all units on the map are searched for a winner unit. In this group 's modi ed algorithm, only neighboring units of the preceding winner are searched for during learning of adjacent probes. If the distance between winner in neighboring units and reference sequence exceeds the threshold, the global winner is searched for from all units on the map. This modi cation improves not only computation time but also layout of probes, organizing consecutive probes closely on the map. Secondly, distance measure between the probe and reference sequence was changed. In conventional SOM, hamming distance is used, so that elements in the same position between two vectors are compared. For hybridization, a probe and its shifted probes (e.g. AGTCAT and GTCAT* or *AGTCA) hybridize closely onto the target sequences, so their distance will be small. The distance measure was changed so as to compare two sequences, considering the sequence shift of either one.
In this study, the update strategy was changed from an incremental method to a batch method. Using a batch method, it may be possible to intentionally direct the update of the units on the 2-dimensional map. For this purpose, a simulated annealing method in the update phase was used. A probe associated to a unit is determined according to the reference vectors associated with the unit and its neighboring units. The number of matching nucleotides for each position is counted for each nucleotide 'A','G','T' and 'C' taking into account shifts between the current probe associated with the unit and those associated with neighboring units. The nucleotide at each position of the probe is determined stochastically by a simulated annealing method depending on the number of the match for each nucleotide 'A','G','T' or 'C'. Probes associated to the units on the map are determined gradually as adjacent probes are mapped closely on the map. The detail of this algorithm is as follows.

Batch SOM algorithm using simulated annealing

```
Step 1 Initialization
       Initialize the map of probes of length L using the 1st and 2nd
       principal components as base vectors of the 2 dimensional plane,
       where L is length of the probe associated to each unit. Set BN=0.

Step 2 Batch learning phase

For each reference sequence, repeat the following steps.

Step 2.1 Initialize the buffer of the reference vectors associated to
         each unit and set position of reference sequence P=P'=0

Step 2.2 From all of units on the map, search the winner units(W_U)
         which is associated to the closest probe to the sub-sequence
         of length L which starts from P'.

Step 2.3 If the difference between them is larger than TH1,
         go to Step 2.8, where TH1 is the threshold which decreases with
         increment of BN.

Step 2.4 Add the sub-sequence starts from P' to the buffer associated
         to W_U.

Step 2.5 Update P=P`+1,  IF P >= SEQ_LEN-L go to Step 3,
         where SEQ_LEN is length of reference sequence.

Step 2.6 Search the winner unit W_U from direct neighboring units
         and position P' on the reference sequence, where W_U is
         associated to the closest probe from the sub-sequence
         which starts from P', where P <= P' < P+ L/2

Step 2.7 If the difference between them is larger than TH2, go to Step 2.8,
         where TH2 is the threshold which decreases with the increment
         of BN and Th2>Th1,
         else go to step 2.4.

Step 2.8 Update P=P'=P+1.
         If P>=SEQ_LEN-L then
           If all reference sequences are processed goto step 3
           else select next reference sequence and set P=P'=0
         go to step 2.2

Step 3 Batch update phase using simulated annealing

Step 3.1 Initialize the iteration number N=0 and set DT=log(TI/TT)/NA,
         where TI is initial temperature, TT is terminate temperature
         and NA is number of iteration in simulated annealing.

Step 3.2 Calculate the Unit[x][y].upd[Pos][Nuc] which is the number of
         occurrences of nucleotide Nuc('A','G','T' or 'C') at position
```

```
         Pos(1,2,...,L) in the buffer of Unit[x][y] for each unit.

Step 3.3 Calculate the Unit[x][y].supd[Pos][Nuc] which is sum of upd
         values in the neighborhood of Unit[x][y] for each units as follows.

Step 3.3.1 Set Unit[x][y].supd[Pos][Nuc]=Unit[x][y].upd[Pos][Nuc]
Step 3.3.2 For all neighboring units Unit[xn][yn] in distance D <= Nr(BN) do
            Calculate the best match number of shifts SB between the
            current probe associated to Unit[x][y] and Unit[xn][yn],
            where Nr(BN) is the maximum distance of neighbors at BN.
            For each Pos and Nuc do
             add Unit[xn][yn].upd[Pos+SB][Nuc]/fn(d) to Unit[x][y].supd[pos][Nuc],
             where fn(d) is neighborhood function.

Step 3.4 For each Unit[x][y] and position Pos in the sequence, determine
         a current probe stochastically as follows.

Step 3.4.1 Find the maximum of Unit[x][y].supd[Pos][Nuc] by changing Nuc
           in A,G,T,C and set the value to mupd.

Step 3.4.2 Calculate the probability base value Pr[Nuc] for each Nuc in
           {A,G,T,C} as follows
              prb[Nuc]=exp(-KT*(mupd-supd[Pos][Nuc])/mupd),
           where T=TI/exp(DT*N) is current temperature and K is a
           positive constant.

Step 3.4.3 Determine the nucleotide at position Pos of current probe
           of Unit[x][y] using the probability to select Nuc as
           Pr[Nuc]=prb[Nuc]/(prb[A]+prb[T]+prb[G]+prb[C])

Step 3.4.5 Update N=N+1, and if N<NA go to Step 3

Step 4 Update BN=BN+1 and if BN< MAX_BN go to Step 2,
       else this algorithm stop.
```

After learning the reference sequences, target sequences to be analyzed are mapped to the map organized by SOM. In previous reports, sequences were discretely mapped to each probe on the map by complete matching of nucleotides between target sequences and probes associated with the units. However, during learning of the map, reference sequences are directed to be mapped continuously by this algorithm. Thus, the mapping method was changed from a complete discrete match to an incomplete continuous match using the threshold used in Step 2 of the SOM algorithm.

4 Experimental results

Experiments changing the size of maps, length of probes on the map and the algorithm were performed. As the reference sequence, we used a set of gene sequences taken from metabolic and regulatory pathways in the KEGG database. These sequences were categorized by gene function and species. Experiments were also performed with changed reference sequences, according to species set, function set, speci c chromosome set, etc. Here, results of analyses

using a set of human genes are discussed to examine the performance of the proposed algorithm. Human sequences used (1132 genes) had a total length of approximately 1.7 M base pairs. Training of some 2-dimensional maps was performed by changing size (32x32, 64x64, 128x128) and changing the length of probes (6 bp-8 bp). Algorithms changing the updating method of the map were also compared. SOM1 is a simple incremental SOM algorithm, and SOM2 an incremental SOM algorithm with modi cation of the distance measure and update method for continuous mapping of adjacent sequences reported at WSOM'03 [3].
BSOM1 is a simple batch SOM algorithm using hamming distances for distance measures, and BSOM2 is a batch SOM algorithm using modi ed distance measures considering the shifts in sequences and using simulated annealing at the update phase. Iteration number for learning is set to 2000000 for SOM1 and SOM2, and 60 batch phases of overall sequences for BSOM1 and BSOM2. Parameters for simulated annealing were selected from some experiments and set as K=1.0, TI=0.1, TT=0.001 and NA=200.
Figure 1 shows a trained map of 32x32 probes of 6 bases resulting from applying each algorithm. On each map, similar probes are organized closely on the map. Probes organized by SOM1 produced some regions of identical probes. In contrast, SOM2, BSOM1 and BSOM2 did not make such regions, and SOM2 and BSOM2 organized shifted probes closely on the map.

```
GTGTGA      GTATGA      TTATGA      AGATGA      TGATGA
GTGAGA      GTGAGA      AGATGA      AGATGA      AGAAGA
GGGAGA      AGGAGA      AGAAGA      AGAAGA      CAAAGA
GGGAGA      GGGAGA      GGAAGA      GAAAGA      GAAAGA
GGCAGA      GGCAGG      GGAAGG      GGAAGG      GCAACA
                Simple incremental SOM(SOM1)

ACAGTG      CAGAGC      TCAGCC      GCGCCC      TGACCA
CAGAGT      CCAGAT      ACCAGA      GGCCAG      TGGCCA
TGATTC      AGAAGG      GACCAG      TGGCCG      ATGGTC
GGCACT      CAGGTG      AGATGG      GAGGGC      TGATGG
GGGCAT      ACACCT      CAAAGG      AAATGG      ATGATG
              Modified incremental SOM(SOM2)

CCTTGC      TCTTCC      TTGTCA      TTTTCT      TTGTTT
CTTTCA      GTTTCA      GTGTCA      GTTTCT      GTTTTT
CTTTGA      GTTTGA      GTTTGA      GTTTGT      GGTTGT
CTTGGA      GTTGGA      GGTTGA      GGATGA      GGATGT
GTTGGA      TTTGGA      GGTGGA      GGATTA      GGATGG
                    Simple batch SOM(BSOM1)

CCTTGA      GCCTTG      TGGAGC      GAGCCT      GAGAGC
AGATTG      GCCAGA      AGCCAG      AGAACC      AGAGCC
CTGATT      CCAGAT      TCCAGA      GAGCCA      AAGAGC
ATTGAT      TGATAA      AAAAGA      AAAGAG      AGTTGT
TTGATG      GCCAAA      CAAAAG      AAGAGT      GAGTTG
          Batch SOM with simulated annealing(BSOM2)
```

Figure 1: Magni ed map

Numerical evaluation of the results, consisted of the number of distinct probes organized on the map, number of mapped sub-sequences and number of adjacently mapped sub-sequences on reference sequences. Covering rates of the reference sequences are shown in Table 1. For number of distinct probes, compared with simple incremental SOM, modi ed incremental

Table 1: Numerical evaluations of organized map

Length	6b	6b	6b	6b	7b	7b	7b	7b	8b
Map size	32x32	32x32	32x32	32x32	64x64	64x64	64x64	64x64	128x128
Algorithm	SOM1	SOM2	BSOM1	BSOM2	SOM1	SOM2	BSOM1	BSOM2	BSOM2
Distinct sequences	554	1013	990	972	2163	4059	3886	4011	16448
Mapped sequences	353075	711894	633266	780758	370056	740263	675485	770312	858330
Adjacent sequences	49897	262258	36434	462852	19731	209241	36434	397134	453252
Covering rates	0.86	0.96	0.96	0.93	0.90	0.97	0.97	0.96	0.97

SOM and both batch SOM algorithms organized a larger number of probes, considering map size and usage of the map is over 95 percent for a number of distinct probes and map size. For number of mapped sub-sequences, the thresholds TH1 and TH2 in step 2 of the algorithm are set as L (length of the sequence on the map) and L-1, respectively. The total length of reference sequence is 1647806, so if all of sub-sequences on a reference sequence are mapped, the number becomes 1647806-L. For both cases of 6 and 7 base pair probes, BSOM2 is the best, SOM2 second, BSOM1 third and SOM1 is much worse. For the adjacent sequences, SOM2 and BSOM2, that use a modi cation of distance measure and update method, show apparently better results compared to SOM1 and BSOM1, which use the conventional update method. BSOM2 shows much better results compared to other algorithms. Simulated annealing works very well to direct organization of the map so as to arrange the adjacent probes closely on the map. For covering rates, SOM2, BSOM1 and BSOM2 showed satisfying results covering over 95 percent of reference sequences. Experiments using other sets of sequences show almost the same results, depending on the algorithm. Next, results of sequence analyses using the organized map will be presented. At rst, relationship of categorized gene sequences is shown by mapping sequences on the map. Figure 2 shows mapping results of 3 DNA sequences of genes, No. 3620 taken from amino acid metabolism, No. 4200 from carbohydrate metabolism and No. 770 from energy metabolism on the map organized by BSOM2. Each sequence shows a pattern of continuous dots, because BSOM2 organizes the probes adjacent on each sequence closely on the map. Figure 3 shows mapping results for

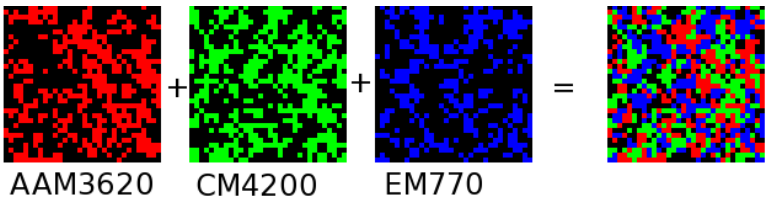

Figure 2: Mapping of gene No.3620+4200+0770 (32x32 units))

all sequences taken from 3 metabolic pathways on the map of 32x32 units associated with 6 base pair probes. The color shows the most signi cant pathway for each probe. The leftmost gure shows results using the map from SOM1, the second gure SOM2, third gure BSOM1 and the rightmost gure BSOM2. From these gures, each pathway shows its own continuous region on the map. Results for SOM2 and BSOM2 are better than for SOM1 and BSOM1, considering that the region for each pathway makes a larger region on the map.

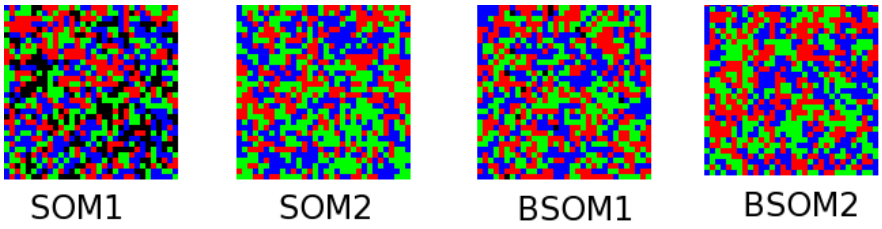

Figure 3: Mapping of amino acid metabolism(red)+carbohydrate metabolism(green)+energy metabolism(blue)(32x32 units)

5 Conclusions

An SOM algorithm using simulated annealing in the batch update phase for sequence analysis was introduced. This group 's SOM algorithm can select a small set of sequences of speci c length that represent characteristic features of the total DNA sequence. Furthermore, some modi cations of the algorithm were made and the method of updating was changed to a batch method. Simulated annealing improves the layout of sequences on the map, organizing adjacent sequences closely on the map, with numerical evaluations and the resulting maps demonstrating the advantages of this algorithm.
An algorithm which can layout more adjacent sequences closely on the map for longer continuous sequences will be important for further development of this model. Using simulated annealing only in the batch update phase, it may be possible to improve performance if it is introduced in the batch learning phase. For this purpose, this group is continuing to develop a system for DNA analysis based on self-organizing maps, with a friendly Graphical User Interface (GUI) that is easy to use for biologists.

References

[1] Hiroshi Dozono, A Design Method of DNA chips for SNP Analysis Using Self Organizing Maps, Advances in Self-Organizing Maps, Springer, 4, 152-159, (2001)

[2] Hiroshi Dozono, A Design Method of DNA chips Using Hierarchical Self Organizing Maps, Proceedings of WSOM'03,(2003)

[3] Dozono,H. and Noguchi,Y., An Application of Genetic Algorithm to DNA Sequencing by Oligonucleotide Hybridization, Proc. of IEEE International Joint Symposia on Intelligence and Systems, pp.92-98,May(1998)

[4] Giuliano,F and et.al. Potentially functional regions of nucleic acids recognized by a Kohonen's self organizing map, Comput. Appl. Biosci. 9, 687-93, (1993)