TR-0800

Formulation of PROTEIN SEQUENCE ANALYSIS using Knowledge

by
M. Hirosawa, M. Hoshida & M. Ishikawa

September, 1992

ICOT
Mita Kokusai Bldg. 21F
4-28 Mita 1-Chome
Minato-ku Tokyo 108 Japan

(03)3456-3191 ~ 5
Telex ICOT J32964

Institute for New Generation Computer Technology

Formulation of PROTEIN SEQUENCE ANALYSIS using Knowledge

Makoto Hirosawa, Masaki Hoshida, Masato Ishikawa
ICOT

Protein sequence analysis is used to predict the structure of protein and to draw the phylogenic tree of creatures. Multiple alignment is one of the essential technique for analyzing the protein sequence. And various algorithms for multiple alignment have been developed. But, because such algorithms try to optimize some evaluation function for multiple alignment, the multiple alignment they produce are not necessarily biologically optima.

We interviewed experts on multiple alignment and extracted the alignment rules and biological knowledge which they use to make multiple alignment. Based on analysis of these result, we formulated and constructed multiple alignment system.

知識を用いた蛋白質の配列解析の試み

広沢誠，星田昌紀，石川幹人
(財) 新世代コンピュータ技術開発機構

蛋白質の機能解析や生物の系統樹を書くために、蛋白質をマルチプルアライメント（配列解析）を作成することは重要な技術である。このため従来から、様々なマルチプルアライメントのアルゴリズムが開発されてきた。しかしながら、これらのアルゴリズムは計算機的に設定された評価値を最適化するものであり、生物学的に妥当なマルチプルアライメントとは必ずしも一致しない。

我々は、マルチプルアライメントの専門家にインタビューし、彼らがマルチプルアライメントを作成する際に意識、無意識的に使用しているアライメントルール、生物学的知識を抽出した。そして、それらの解析結果に基づき、マルチプルアライメントシステムを試作した。

1 Introduction

Multiple alignment, which is used for similarity analysis of protein, is an important technique for drawing the phylogenic trees of creatures and for predicting the function and structure of proteins. By simultaneously aligning the sequences of similar proteins, we can identify the regions of protein sequences that may have important functions when these sequences are folded into proteins. Also, by simultaneously aligning the same proteins (hemoglobin, for example) from different species of creatures, we can analyze the similarity between sequences belonging to creatures and draw phylogenic trees of creatures.

Because biological expertise is necessary for multiple alignment, biolgists have up to recently produced multiple alignment by hand. However with the increasing rate of determination of protein sequences, the number of multiple alignments that biologists must handle has also increased remarkably. And each multiple alignment has become more difficult because the number of sequences that must be aligned and the length of their sequences have increased. This situation has become more burdensome on biologists Therefore, computer for use in multiple alignment are now indispensable. And researches have been made to facilitate multiple alignment by computer.

1.1 Review of Multiple Alignment

So far, various multiple alignment algorithms have been developed. These algorithms try to optimize computationally defined evaluation function and produce computaionally optimal or semi-optimal alignment. The evalation function is based on a similarity index between amino acids. The Dayhoff score index [Dayhoff 1978] is one of the similarity indices generally used.

Needleman and Wunsch [Needleman and Wunsch 1970] introduced Dynamic Programming(DP) into multiple alignment. With n way DP, the computationally optimal alignment can be produced theoretically. However, the problem with DP is the incredible length of time it takes to compute. N-way DP takes computational time in the order of the n-th power of the sequence length. If we align sequences which are short, by resricting solution space we can align the sequences within the manageable time[Carrillo and Lipman 1988]. However, if we apply the program to practical problem, the program takes extremely long time.

To keep this expansion of the computational time manageable, various multiple alignment algorithms have been developed that can produce semi-optimal alignment within a limited time. As algorithms with 2 way DP, tree based algorithms [Johnson and Dolittle 1986] [Barton 1990] and the iterative improving algorithm[Berger and Manson 1991][Ishikawa *et al.* 1992] were developed. As an iterative improving alignment system without 2 way DP, an alignment by simulated annealing was developed[Ishikawa *et al.* 1991].

Recently, with the increased power of the computer, 2 way DP as well as 3 way DP has become available [Murata 1985] and alignment system based on 3-way DP has been developed [Hirosawa *et al.* 1991].

1.2 Need for Multiple alignment system with Refiner

Today, biologists use these programs to produce a temporary alignment. However, biologists must have to refine it to produce the final alignment . They must refine the alignment into a biological meaningful alignment by themselves because the alignments that these programs produce are just computational optimal or semi optimal alignments instead of biologically optimal alignments.

As stated before, the number of alignments that biologists must handle is also increasing, and the difficulty of each alignment is increasing. Therefore, biologists now feel the need for computer assistance even for

refining temporary alignments. And with protein sequence data accumulated in databases, researchers other than biology experts now have the chance to make biological discoveries merely by analyzing sequence data. Automatic refinement of alignments using biological know-how and knowledge is also necessary for them.

For knowledge to be used in the refinement phase, the heuristics that biologists rely on are rather ambiguous. However, knowledge on the biological meaningful portion of sequences has been accumulated in databases, one of which is Prosite [Bairoch 1991], and another information has been published in various papers.

We have developed an alignment system composed of two modules, namely, *the aligner* and *the intelligent refiner*[Hirosawa 1992]. The aligner produces a computationally semi-optimal alignment. Then, the intelligent refiner refines the product of the aligner to produce the biologically optimal alignment.

To design an intelligent refiner, we interviewed experts on multiple alignment. We base the framework of the intelligent refiner on analysis of the way they align sequences and the knowledge they use for alignment.

In this paper, we explain the alignment system with the intelligent refiner. In the second section, we explain why the alignment with intelligent refiner is important. In the third section, the alignment system with the intelligent refiner is introduced. Finally, discussion is delivered.

2 Why is an intelligent refiner necessary?

Subsection 2.1 shows that computationally optimal alignment does not always correspond to biologically optimal alignment. And subsection 2.2 indicates the possibility that we can make biologically optimal alignments from computationally optimal alignments by using biological knowledge.

2.1 Biological reliability of computationally optimal alignment

It is hard to define biologically optimal alignment, because this depends on the sequences to be aligned. Here, we select some sequences as an example, and we define biologically optimal alignment of the sequences. Then, we investigate the biological reliability of the computationally optimal alignment.

Definition of biologically optimal alignment

To define the biologically optimal alignment, we will use are six sequences of protein called endonuclease from Retro-virus and its relatives. This is shown in Figure 1, and one of its biological optimal alignments is shown in Figure 2.

```
17.6    : ILDFHEKLLHPGIQKTTKLFGETYYFPNSQLLIQNIINECSICNLAKTEHRNTDMPTKTT
M-MuLV  : LLDFLHQLTHLSFSKMKALLERSHSPYYMLNRDRTLKNITETCKACAQVNASKSAVKQGTR
HTLV    : LTDALLITPVLQLSPAELHSFTHCGQTALTLQGATTTEASNILRSCHACRGGNPQHQMPRGHI
RSV     : VADSQATFQAYPLREAKDLHTALHIGPRALSKACNISMQQAREVVQTCPHCNSAPALEAGVN
MMTV    : ISDPIHEATQAHTLHHLNAHTLRLLYKITREQARDIVKACKQCVVATPVPHLGVN
SMRV    : ILTALESAQESHALHHQNAAALRFQFHITREQAREIVKLCPNCPDWGSAPQLGVN
```

Figure 1 Sequences to be aligned

This set of sequences is from endonuclease of retrovirus and its relatives

```
17.6   : -------ILD--F-------HEKLLHPGIQKTTK-LF--GET-YY-FPNSQLLIQNIINECSICNL-AKT-EHR--N-TDMPTKTT
M-MuLV : -------LLD-FL-------HQ-LTHLSFSKM-KALLERSHSPYYMLNRDRTL-KNITETCKACAQ-VNA-SKS--A-VKQGTR--
HTLV   : LTDALL-ITP-VLQLSPAELHS-FTHCGQTAL-T-LQ--------GATTTEA--SNILRSCHACRG-GNPQHQMPRGHI-------
RSV    : VADSQATFQAYPLR-EAKDLHT-ALHIGPRAL-S-KA-------CNISMQQA--REVVQTCPHC---NSA-PALEAG-VN------
MMTV   : -------ISD-PIH-EATQAHT-LHHLNAHTL-R-LL-------YKITREQA--RDIVKACKQCVV-ATPVPHL--G-VN------
SMRV   : -------ILT-ALE-SAQESHA-LHHQNAAAL-R-FQ-------FHITREQA--REIVKLCPNCPDWGSA-PQL--G-VN------
                          *   *                       *  *
```

Figure 2: One of biologically optimal alignment

The alignment identifies the reverse pattern of the prototype zinc finger

Columns whose elements are identical are marked by "*". These columns are called conserved columns. There are four conserved columns "H"s and "C"s. Some patterns of conserved columns are known to have biological meaning and are called motifs. The motif consisting of conserved patterns of "HXXXH" and "CXXC" is a reverse prototype motif of the zinc finger. "X" in the pattern means any amino acid. Protein with the zinc finger is one of those known to have the capability to bind to DNA/RNA sequences. We define the biologically optimal alignmenta as the alignments that identify the zinc finger pattern.

The homology score (the index that measures the similarity between sequences) of sequences in the alignment is very low at 26 percent. It indicates that these sequences are difficult to align.

Biological reliability of computationally optimal alignment

We investigated the biological reliability of alignments produced by 3-way DP, because the computationally optimal alignment of three sequences is obtained by 3-way DP.

We selected three sequences from these six sequences and aligned the three sequences by 3-way DP. Since there can be 20 triplets of sequences, a corresponding number of alignment were produced. Then, we investigated whether the alignment corresponds to the biologically optimal alignment.

Of the twenty alignments, only six alignments identified the zinc finger. It indicates that biological opimal alignments don't neccessary correspond to the biological meaningful alignment.

2.2 Possibility of Inference of biologically optimal alignment from computationally optimal alignment

As shown in 2.1, the computationally optimal alignment doesn't necessarily correspond to biological optimal alignment. However, biologists and even computer scientists with biological knowledge can make biologically optimal alignment by refining the computationally optimal or near-optimal alignment.

We will show you an example in which we can make the biologically optimal alignment by refinement if we use biological knowledge. The example alignment to be refined is produced 3 way DP and is computationally optimal alignment. The sequences to be aligned are the sequences whose biological optimal alignment is defined in 2.1. The biologically optimal alignment must identify the zinc finger pattern.

Example Alignment

Example alignment (Figure 3)can be refined into a biologically more optimal alignment if we have biological knowledge. In the part of the alignment that corresponds to "HXXH", "H" is not properly aligned but is mis-bridged. Here, mis-bridged means that "H"s corresponding to the first letter in "HXXH"(in the third

sequence) are aligned with "H"s corresponding to the last letters "HXXH"(in the first and second sequences)

If we fix the mis-bridged "H" and refine the alignment, we cannot identify the zinc finger. However, if we have experience in aligning the sequences of some protein that contains the zinc finger, we can guess the possibility of a mis-bridged "H" and therefore, refine the alignment into an alignment in which the zinc finger is identified (Figure 4).

```
17.6    : -------ILDF--HE-KLL-HPGIQKTTKLF--GET-YY-FPNSQLLIQNIINECSICNLAKTEHRNTDMPTKTT
M-MULV  : -------LLDF-LHQ---LTHLSFSKMKALLERSHSPYYMLNRDRTL-KNITETCKACAQVNASKSAVKQGTR--
RSV     : VADSQATFQAYPLREAKDL-HTALHIGPRAL--SKA-CN-ISMQQA--REVVQTCPHCNSAPALEAGVN------
   (Evaluation value = 161)
```

Figure 3 Example Alignment 2

```
17.6    : -------ILDF--------HEKLLHPGIQKTTKLF--GET-YY-FPNSQLLIQNIINECSICNLAKTEHRNTDMPTKTT
M-MULV  : -------LLDF-------LHQ-LTHLSFSKMKALLERSHSPYYMLNRDRTL-KNITETCKACAQVNASKSAVKQGTR--
RSV     : VADSQATFQAYPLREAKDLHT ALHIGPRAL--SKA-----CN-ISMQQA--REVVQTCPHCNSAPALEAGVN------
   (Evaluation value = 156)
```

Figure 4 Biologically more optimal alignment of Example 2

The above example indicates that we can infer biologically optimal alignment from biologically optimal alignment. It is also found that if we use only the evaluation value of the alignment as a measure of the alignments, we cannot make biologically optimal alignment (In the example, the evaluation value of the alignment is reduced from 161 to 156 when we refine the alignment).

3 Multiple alignment system with Intelligent Refiner

3.1 Framework of the alignment system with Intelligent Refiner

Our system consists of two modules, *the aligner* and *the intelligent refiner* (Figure 5). The aligner produces a computationally optimal or near-optimal alignment. The alignment is made without using biological knowledge.

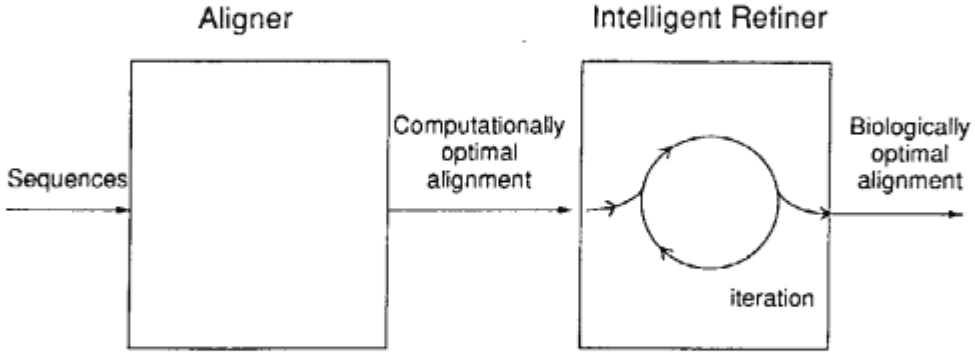

Figure 5 Multiple alignment system with intelligent refiner

By analyzing the alignment produced by aligner and by consulting biological knowledge, the intelligent refiner can roughly understand where the conserved column regions are and where another conserved column region may be found in the alignment. The intelligent refiner modifies the alignment in order to increase biologically meaningful conserved column region. The modified alignment in one cycle becomes the input to the next iteration. In each iteration, the intelligent refiner can understand more precisely where the conserved regions are. Thus, it can gradually identify the conserved column regions to produce the biologically near-optimal alignment.

Because we can use any alignment system that can produce a computationally optimal or near-optimal alignment as the aligner module, we don't describe the aligner. In the following subsection, only the intelligent refiner is explained.

3.2 Intelligent refiner

We designed the intelligent refiner by analyzing the knowledge used by experts on multiple alignment. The program of intelligent refiner is written in Prolog and KL1. The structure and function of the intelligent aligner is explained below.

3.2.1 Framework of intelligent refiner

The overall framework of the intelligent refiner is shown in Figure 6. The intelligent refiner is composed of *a control module*, *refinement rule base* and *biological knowledge base*. The control module iteratively modifies the alignment by calling refinement rules in the refinement rule base to produce a biologically optimal alignment. Refinement rules consult with biological knowledge base when neccessary.

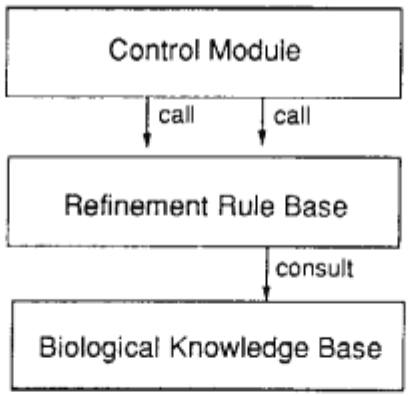

Figure 6 Framework of Intelligent Refiner

Biological knowledge base

In our system, biological knowledge is written in Prolog syntax. In the biological knowledge base, the biological knowledge we extracted from the database, Prosite [Bairoch 1991] and so on, are contained. Prosite is the database in which knowledge on motifs and related knowledge is written in natural language.

Besides the knowledge we store, biologists can easily input their own knowledge into the biological database, because Prolog has a syntax similar to natural language.

We show a portion of the biological knowledge in Figure 7. The syntax and meanings are explained below. `motif` is a predicate that tells us the meaning of the motif in the third argument. The expression of the motif

is similar to that employed in Prosite. When its first argument is `name`, the second argument means the name of the motif, for example, zinc finger[(1)]. When its first argument is `protein`, the second argument means the name of the protein that contains the motif, for example, kinase[(2),(3),(4)].

The predicate `upper_concept` expresses the hierarchical relationship of protein. For example, serine_threonine kinase is one class of kinase[(5)], and tyrosine kinase is another class of kinase[(6)].

Expression (11) is a rule which means that even if we don't know the motif of a protein, we can infer it if we know the motif of the protein which is the upper concept of the protein we are focusing on.

```
motif(name, zinc_finger(reverse), ''H-X(3,5)-H-X(10,25)-C-X(3,5)-C''). (1)
motif(protein, kinase,''[LIV]-G-X-G-[FY]-[SG]-X-[LIV]''). (2)
motif(protein, kinase(tyrosine),''[LIVMFYC]-X-[HY]-X-D-[LIVMFY]-K-X(2)-N-[LIVMFC](3)''). (3)
motif(protein, kinase(serine,threonine),
                        ''[LIVMFYC]-X-[HY]-X-D-[LIVMFY]-RSTA-X(2)-N-[LIVMFC](3)''). (4)
upper_concept(kinase(serine,threonine), kinase). (5)
upper_concept(kinase(tyrosine), kinase). (6)
motif(protein,Protein,Motif)  :-
          upper_concept(Protein,UpperProtein), motif(protein,UpperProtein,Motif). (7)
```

Figure 7 Biological knowledge base

3.2.2 Refinement rule base

We extracted more than fifteen rules from alignment experts. Of these, ten rules are currently used in the intelligent refiner.

The rules are expressed using the IF-THEN rule. Here we show representative five rules are shown in Figure 8. The rules will be explained using examples later. In the rules, several routines are called. But, since their functions are clear from the context, we will not explain the routines further.

Rule 1 (see Figure 9)

IF An half conserved column c_i , in which more than the specified percentage(e.g. 80 %) of whose elements are identical amino acids (x_s), is found. **AND**

In the sequence that doesn't have x_s in the column, x_s are sought within specified distance from c_i (checked by `search routine 1`).

THEN The modified alignment is produced in the constraint that the found x_s is aligned in the column c_i (done by `modification routine`). When plural alternatives are generated, the modified alignment whose evaluation value is the highest is selected.

Rule 2 (see Figure 10)

IF Sequences in the alignment are grouped in to two groups, g_i and g_j according to similarities between the sequences (checked by `grouping routine`). **AND**

Patterns of conserved columns (p_i and p_j) in each group of alignment (a_i and a_j) are found **AND**

Common sub-pattern (p_{ij}) is found in the both patterns (p_i and p_j) (done by `search routine 2`)

THEN a_i and a_j are aligned in the constraint that p_{ij} in a_i and p_{ij} in a_j are aligned (done by `modification routine`).

Figure 8(to be continued) Representative rules in refinement rule base

Rule 3 (see Figure 11)

IF Discovered conserved column pattern p contains some part of an motif m_i stored in the biological knowledge base (done by `motif_check routine`).

THEN the `Motif_finding routine` is called to identify the other part of the motif.

Rule 4 (see Figure 11)

IF An motif (that `motif_finding routine` trys to identify) in the biological knowledge base have two amino acids, x_i and x_j, of same kind of amino acid x **AND**

The motif has no other conserved amino acids between x_i and x_j **AND**

There are a half conserved column of c_i of x and a conserved column of c_j of x in the alignment (done by `motif_finding routine`).

THEN `Modification routine` is called to produce alignment in the constraint that $x_{j,t}$ (belonging to the the sequence s_t that doesn't have x in the half conserved column c_i) is aligned in the half conserved column c_j.

Rule 5

IF A motif (m_i) is identified in the alignment **AND**

The protein that has the motif (m_i) have another motif m_j (checked by `knowledge_consulting routine`)

THEN the `motif_finding routine` is called to identify the motif m_i

Figure 8 Representative rules in refinement rule base

Example application of Refinement rules

Five refinement rules are explained using exmaples.

Rule 1 and Rule 2

Figure 9 is an example of application of Rule 1. By `search routine 1` a half conserved column of "A" (there is an exception in the first sequence) is found ("_" in Figure 9 signifies any character) and in the first sequence, an "A" is found in the neighborhood of the half conserved column. Then, Rule 1 is fired. The `modification routine` is called to produce a conserved column of "A" (computational optimal or semi-opitmal alignment is produced in the constraint that all "A" should be aligned in a column).

Figure 9 Example application of Rule 1

"_" means any character

Figure 10 shows an example of the application of Rule 2. Here, the sequences are decomposed into two groups, the first three sequences and the last three sequences according to similarities of the sequences (checked by `grouping routine`)

By `search routine 1`, it is found that a conserved column pattern "AP" is found in both groups. Then, Rule 2 is fired. The `modification routine` is called to produce a conserved column of "AP" that extends all sequences.

Figure 10 Example application of Rule 2

The sequences can be decomposed into two groups, the first three sequences and the last three sequences, according to the similarities between the sequences

The temporary alignment produced by Rule 1 or Rule 2 is sent to the `evaluation routine` with the current alignment (the most biologically optimal alignment at the time). In the routine, the current alignment and the temporary alignment are evaluated by consulting with the biological knowledge base. Then, the new current alignment is selected.

Rule 3 and Rule 4

An example application of Rule 3 and Rule 4 is explained with the use of Figure 11. The `motif_check routine` identifies that the conserved column of "CXXXC" is the latter part of zinc finger (reverse type) motif "H-X(3,5)-H-X(10,25)-C-X(3,5)-C" by consulting `the knowledge (1)` in the biological database. Then, rule 3 is fired. The `motif_finding routine` finds the half conserved pattern of "HXXXH" in which the the latter column of "H" is half conserved. Then, rule 4 is fired. The `modification routine` makes the alignment which has a conserved column pattern of "H-X(3)-H-X(15)-C-X(3)-C"

Figure 11 Example application of Rule 3 and Rule 4

Rule 5

The example application of Rule 5 is explained by using the sequences belonging to a protein called tyrosine kinase. When the rule is applied, `the knowledge (2)(3)(6)(7)` in the biological knowledge base are consulted.

If motif "[LIVMFYC]-X-[HY]-X-D-[LIVMFY]-K-X(2)-N-[LIVMFC](3)" is identified in the alignment, the `knowledge_consulting routine` finds that it is the motif of tyrosine kinase (`knowledge (3)` in the biological database). Then, other motifs belonging to tyrosine kinase are sought in the biological knowledge base. Here, corresponding motifs are not stored explicitly in the biological database. However, using the inference rule (`knowledge (7)`), and the knowledge that kinase is the upper concept of tyrosine kinase (`knowledge (6)`) and knowledge on motif of kinase (`knowledge(2)`), it is derived that tyrosine kinase also has a motif "[LIV]-G-X-G-[FY]-[SG]-X-[LIV]".

Then, Rule 5 is fired, and the `Motif_finding routine` is called to identify "[LIV]-G-X-G-[FY]-[SG]-X-[LIV]".

4 Result

Because of the limitation of space in this paper, we can't show examples. However, we applied the intelligent refiner to several examples and succeeded. For example, alignment in figure 3 can be refined to the alignment that catches the zinc finger. The reader who are interested in the examples of application should consult our recent paper[Hirosawa *et. al* 1992].

5 Discussion

1. [Barcon and Anderson ; Butler *et al.* 1990] demployed divide-and-conquer strategy to multiple alignment. Their algorithm interatively find conserved amino acids and pinning these amino acids in the alignment, then, find another conserved amino acids in both sides of the pinned region of the alignment.

   The program has two weak points. One is that, because the program is applied to the sequences that are not aligned, it often produces spurious conserved columns. The other is that once some spurious conserved column is produced, the spurious conserved column is fixed and is never revised.

   The alignment system with intelligent refiner solves the problem as follows to make the reliability of the alignment the system produces higher.

   - Our system, firstly, generates computationally near-optimal alignment and the system refines the computationally near-optimal alignment using knowledge. The computationally near-optimal alignment doesn't necessarily correspond to the biological optimal alignment. However, we can get information on where the possible conserved columns are. Then, we can gradually increase the reliability of the information by iteratively refining the alignment.
   - Because we have knowledge of the motif in the biological knowledge base, our system can identify the motif with high reliability. For example, with biological knowledge, the spurious column conserved of "L" in the 19th column in the alignment in Figure 3 is remedied to securely identify the motif.
   - Because we have alignment rules that break spurious conserved column (e.g. rule 4), the risk that the alignment is trapped by spurious conserved columns is reduced.

2. Conceptually, it is better to input computationally optimal or near-optimal alignments into the intelligent refiner. However, it is possible for biologists to input the alignment roughly made by the hand or the alignment produced by the tree base algorithm. Although, the quality of the alignment that the intelligent refiner produces when the biologists use these alignments as input is worse than the case when the computationally optimal alignment is used as input, the quality of the resultant alignment is tolerable for practical use.

3. The alignment system with intelligent refiner is not only for biologists but also for computer scientists. By analyzing the resultant alignment produced by the intelligent refiner which contains plenty of biological knowledge and refinement rules. the possibility of making biological discoveries will be open.

4. There are refinement rules that we extracted from biologists but we haven't incorporated into the intelligent refiner yet. One of the knowledge is on the alignments that biologists don't favor.

   One class of these unfavored alignments is the alignment with islands phenomenon. The phenomenon is shown in Figure 12 (left), every amino acid is expressed by "X" or "O". There is an island composed

of amino acids in an ocean composed of gap letters "-". The island is composed of amino acids from the sequence $1 \sim 4$. Alignment experts think that the ocean of gaps should be reduced to make compact alignment like alignment on the right of the figure. We are now investigating how to recognize the phenomenon and how to remedy it.

5. Since the intelligent refiner is still at a primitive level, we must continue research to improve the power of the system. The quality of the produced alignment is determined by the amount and quality of biological knowledge and refinement rules. We must increase the biological knowledge and extract effective rules more from experts on multiple alignment to improve the intelligent refiner.

```
Sequence1:   XXXXXXXXXX---O---XXXXXXXX            XXXXXXXXXXOXXXXXXXX
Sequence2:   XXXXXXXXXX--OOO-----XXXXX            XXXXXXXXXXOOO-XXXXX
Sequence3:   XXXXXXXXXX--OOOO----XXXXX     =>     XXXXXXXXXXOOOOXXXXX
Sequence4:   XXXXXXXXXX-----OOO--XXXXX            XXXXXXXXXX-OOOXXXXX
Sequence5:   XXXXXXXXXXXXX-------XXXXX            XXXXXXXXXXXXX-XXXXX
```

Figure 12 An example of unfavored alignment : Island phenomenon and its remedied alignment.

Acknowledgment

The authors acknowledge R.Tanaka and Y.Totoki of IMS for their programming and experimental effort. We would like to especially thank K.Kuma, N.Iwabe of Kyoto Univ. for their willingness to answer our insistent questions when they showed us their actual alignment process. We also acknowledge T.Miyata and H.Hayashida of Kyoto University, H.Toh of PERI and George Michaels of NIH for their discussion on multiple alignment.

References

[Bacon and Anderson 1986] D.J Bacon, W.F.Anderson. *J.Mol.Biol.*, 191, 153-161 (1986)

[Bairoch 1991] Bairoch,A. Prosite : A dictionary of protein site ans pattern : User manual Release 7.00, May 1991.

[Berger and Manson 1991] Berger,M. and Manson,P. A novel randomized iterative strategy for aligning multiple protein sequences. *Computer Application in the Biosciences*, 7, 1991. pp.479-484.

[Barton 1990] Barton,J.G. Protein Multiple Alignment and Flexible Pattern Matching. in *Methods in Enzymology Vol.183*, Academic Press, 626-645.

[Butler *et al.* 1990] Butler,R., ,Butler,T., Foster,I., Karonis,N., Olson,R., Overbeek,R., Pfluger,N., Price.M. and Tuecke,S. Aligning Genetic Sequences *in Foster,I. and Taylor,S. Strand – New concept in parallel programming.* Prentice Hall.

[Carrillo and Lipman 1988] H. Carrillo and D. Lipman. The multiple sequence alignment problem in biology. *SIAM J. Appl. Math.*, 48, 1988, pp.1073–1082.

[Dayhoff,O. *et al.* 1978] Dayhoff,M.O., Schwatz,R.M. and Orcutt,B.C. A model of evolutionary change in proteins. In Dayhoff,M.O.(ed), *Atlas of Protein Sequence and Structure Vol.5, Suppl.3*, Nat. Biomed. Res. Found., Washington, D. C., 363–373.

[Hirosawa *et al.* 1991] Hirosawa,M., Hoshida,M., Ishikawa,M. and Toya,T. Multiple Alignment System for Protein Sequences employing 3-dimensional Dynamic Programming. *Genome Informatics Workshop II*, (in Japanese).

[Hirosawa *et al.* 1992] Hirosawa,M., Ishikawa,M. Hoshida,M. Protein Multiple Sequence Alignment System using Knowledge *ICOT TR 793*, 1992.

[Ishikawa *et al.* 1991] Ishikawa,M., Toya,T., Hoshida,M., Nitta,K., Ogiwara,A. and Kanehisa,M. Multiple Alignment by Parallel Simulated Annealing. *Genome Informatics Workshop II*, (in Japanese).

[Ishikawa *et al.* 1992] Ishikawa,M., Hoshida,M., Hirosawa,M., Toya,T. and Nitta,K. (1992) Protein Sequence Analysis by Parallel Inferrence Machine. *Proc. Int. Conf. on Fifth Generation Computer Systems 1992.*

[Johnson and Dolittle 1986] M. S. Johnson and R. F. Doolittle. A method for the simultaneous alignment of three or more amino acids sequences. *J. of Mol. Evol.*, **23**, 1986, pp.267–278.

[Murata 1985] Murata,M. Simultencous comparison of three protein sequences *Proc. Natl. Acad. Sci. USA Vol. 32*, 1985, pp.3073-3077.

[Needleman and Wunsch 1970] Needleman,S.B. and Wunsch,C.D. (1970) A General Method Applicable to the Search for Similarities in the Amino Acid Sequences of Two Proteins. *J. of Mol. Biol.*, 48, 443-453.