Direct Comparison DNA and Amino Acid Sequences Based on a Dynamic Programming Method

Naoko Kasahara
kasahara@crl.hitachi.co.jp

Susumu Hiraoka
hiraoka@crl.hitachi.co.jp

Keiichi Nagai
k-nagai@crl.hitachi.co.jp

Central Research Laboratory, Hitachi, Ltd., 1-280 Higashi-koigakubo, Kokubunji-shi, Tokyo 185, Japan

Abstract

We have developed a method based on a dynamic programming method, that enables us to directly compare DNA and amino acid sequences. This method makes it possible to find homologies between translated DNA sequences and amino acid sequences by recognizing gaps in both types of sequences. This method allows higher sensitivity and specificity than is possible with BLASTX, which has a similar function. To reduce the computation time, we performed a parallel computation on a workstation cluster using a PVM (Parallel Virtual Machine) programming.

1 Introduction

The DNA database size is increasing exponentially, and amino acid sequence comparison plays an important role in protein function analysis. In this respect, it is very important to find homologous sequences of a translated DNA sequence in the amino acid sequence database. BLASTX is often used for such purposes. BLASTX translates a DNA sequence into amino acid sequences in six frames, including its complementary sequences, and compares these sequences to amino acid sequences in the database. However, BLASTX doesn't recognize any gaps in either the DNA or amino acid sequences, and insertions and deletions in DNA sequences cause frame shift errors, especially in EST sequences. Therefore, with BLASTX, the sensitivity and specificity when searching for homologous sequences is degraded in some cases. We have developed a highly sensitive method to compare DNA sequences and amino acid sequences directly that recognizes the gaps in both sequences by using a dynamic programming calcu! lation[1][2].

2 Method

The algorithm we use to directly compare a DNA sequence with an amino acid sequence, has three steps : 1) translating the DNA sequence into an amino acid sequence nucleotide - by - nucleotide, 2) comparing the translated amino acid sequence with amino acid sequences in the database, allowing gaps to exist in either sequence, 3) displaying the alignments of these sequences.

First, we will explain how we translate a DNA sequence into an amino acid sequence. From the 5' end of the DNA sequence, we got a codon and translate it into an amino acid. Then we move on to the next codon by shifting one nucleotide in the 3' direction and translating it. By continuing this process until we reach the 3' end of the sequence, we can get the translated amino acid sequence. The same procedure is then performed with the complementary sequence.

Next, we compare the translated amino acid sequence with an amino acid sequence that is already stored in the protein database. Since our method is based on a dynamic programming method, we consider a matrix that shows the translated amino acid sequence vertically and the other amino acid sequence horizontally. Then we calculate the homology score by a dynamic programming calculation. To identify any possibility of error in each sequence due to substitution, insertion or deletion, we take seven paths in the calculation. Those paths are taken to consider nucleotide base deletion and insertion, amino acid residue insertion and deletion, or the combination of these. After calculating homology scores for all pairs of the amino acid sequence, we choose the highest homology score in the matrix. Finally, we calculate an optimum alignment by retracing the calculation path. Then we can display alignments with the homology scores.

3 Result

We evaluated the sensitivity and the specificity in terms of how many amino acid sequences can be selected from among the sequences that from the same superfamily. As query sequences, we chose DNA sequences from GenBank (rel. 95), that could be identified to code the amino acid sequences that belong to certain superfamilies. Then we compared the DNA sequences and all the amino acid sequences in the PIR database (rel. 47). In some cases, we got much better results than with BLASTX.

Since this method is based on dynamic programming calculation, it requires considerable computational time. To reduce the computation time, we performed a parallel computation on a workstation cluster using a PVM programming. When using up to 11 CPUs, the computational time was reduced in inverse proportion to the CPU number.

References

[1] S.B.Needleman, and C.D.Bunsch; A general method applicable to the search for similarities in the amino acid sequences of two proteins, *J.Mol.Biol*, Vol. 48, pp. 444-453, 1970.

[2] T.F.Smith, and M.S.Waterman; Identification of common molecular subsequences, *J.Mol.Biol*,Vol.147, pp.195-197, 1981.