A Fast Method for Searching Similar Protein Sequences[*] (extended abstract)

Yuan-Hsiu Tsai[a], Kuo-Si Huang[b] and Chang-Biau Yang[a†]
[a]Department of Computer Science and Engineering
National Sun Yat-sen University, Kaohsiung, Taiwan
[b]Department of Business Computing
National Kaohsiung University of Science and Technology, Kaohsiung, Taiwan

Abstract

For a given query protein sequence, this paper proposes the FastSearch algorithm to quickly pick up feasible sequences with better alignment scores in a database. The experimental datasets are retrieved from the NCBI database, including the COVID-19 proteins, the EUMAT proteins and the human proteins. As the experimental results show, our FastSearch algorithm gets better accuracy and time efficiency than the GGSEARCH. The FastSearch is implemented as a tool, and available at the following website. `https://par.cse.nsysu.edu.tw/~fastsearch`

Keywords: sequence alignment, protein sequence, longest common subsequence (LCS), heuristic algorithm

1 Introduction

For finding similar protein sequences in a database with respect to a given query sequence, we can use the global and local sequence alignment algorithms. The sequence alignment problem has been proposed for more than 50 years, and various algorithms have been developed and evolved. For example, the algorithm of Needleman and Wunsch [5] and the algorithm of Smith and Waterman [9] are well-known tools for optimal pairwise alignment [2, 8]. Also, the FASTA [3, 6] and BLAST [1] are well-known heuristic options.

In this paper, we adopt the filtering strategy to reduce the solution space. For a given query sequence, our FastSearch algorithm is heuristic to find the sequences with better alignment scores in the database. The FastSearch is composed of three stages. The first two stages predict the alignment quality according to predefined parameters to eliminate possibly dissimilar sequences. The final stage uses the optimal sequence alignment algorithm to find the sequences with the highest alignment score in the remaining sequences.

As experimental results show, our FastSearch gets better results with low time consumption in searching the COVID-19 protein dataset, the EUMAT dataset and the human dataset from the NCBI [7] database by comparing with the GGSEARCH, a global alignment tool in the FASTA package [4].

2 Our Fast Search Algorithm

Given a query sequence Q, the goal of our algorithm is to quickly get sequences with better alignment scores in the database D. Our algorithm uses the next-match table to locate possible match positions. In addition, we design some heuristic strategies to prune away some sequences from D for reducing the computation time.

Our FastSearch algorithm is composed of three stages, including the FastLCS, the BestJump and the NW stages. For two given sequences, in the first two stages, we design two evaluation functions for estimating the alignment score from the current position to some next candidate positions. Next, one of the candidate positions is selected with the best estimated score. Then, we jump to the selected position with ignoring the other positions, and view it as the new current position. With the jumping concept, the estimated score of two sequences can be obtained in linear time. After the estimated scores of Q and

[*] This research work was partially supported by the Ministry of Science and Technology of Taiwan under contract MOST 109-2221-E-110-040-MY2.

[†] Corresponding author. E-mail: cbyang@cse.nsysu.edu.tw (Chang-Biau Yang).

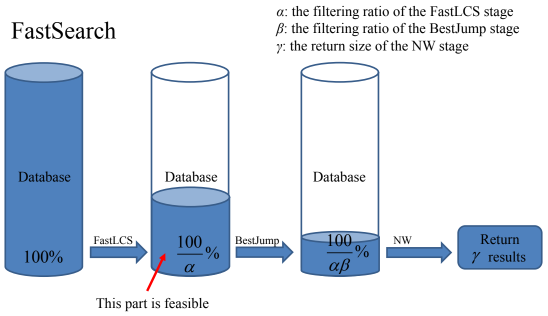
FastSearch
α: the filtering ratio of the FastLCS stage
β: the filtering ratio of the BestJump stage
γ: the return size of the NW stage
Database
100%
FastLCS
Database
$\frac{100}{\alpha}\%$
BestJump
Database
$\frac{100}{\alpha\beta}\%$
NW
Return
γ results
This part is feasible

Figure 1: The concept of our FastSearch algorithm.

all sequences in the database have been determined, some sequences with bad estimated scores are pruned away.

The last stage uses the NW algorithm to calculate the optimal alignment score of Q and each of the remaining sequences. The concept of the FastSearch algorithm is illustrated in Figure 1. With the pruning concept, the number of the remaining sequences for performing NW is small so that the totally computational time can be reduced, since the NW algorithm needs quadratic time.

For two sequences A and B over alphabet Σ, it requires $O(|\Sigma| \times (|A|+|B|))$ space to store the next-match tables, and $O(|A| \times |B|)$ space to compute candidate scores in the BestJump stage. The time complexities are $O(|\Sigma|^2 \times (|A| + |B|))$, $O((|A| \times |B|) \times (|A|+|B|))$, and $O(|A| \times |B|)$ in the FastLCS, the BestJump, and the NW stages, respectively. Note that $O((|A| \times |B|) \times (|A|+|B|))$ for BestJump is estimated with an extreme case. Practically, the real execution time required for BestJump is much less than $O((|A| \times |B|) \times (|A| + |B|))$. For performing the sequence searching in a database D with filtering ratios α and β in the FastLCS and BestJump stages, respectively, the time complexity of the proposed FastSearch algorithm is $O((|\Sigma|^2 \times (|A| + |B|)) \times |D| + (|A| \times |B|) \times (|A| + |B|)) \times \frac{|D|}{\alpha} + (|A| \times |B|) \times \frac{|D|}{\alpha \times \beta})$.

3 Experimental Results

The experimental datasets are obtained from the National Center for Biotechnology Information (NCBI) site [7]. The ratios α and β are varied according to the sizes of datasets. The experiments are performed on a computer of Windows 7 64-bit Ultimate operating system with an Intel(R) Core(TM) i5-4570 CPU and RAM with 16 GB.

In the experiment, our FastSearch algorithm, the GGSEARCH algorithm [6] and the Needleman-Wunsch (NW) algorithms [5] are performed to find some of the best alignment sequences in the database for every query sequence. The GGSEARCH algorithm is a tool in the FASTA package [6] to compute the global alignment with protein sequences and to search similar sequences in the database. The normalized mean error to the NW algorithm in Equation 1 normalizes the score difference by the sequence lengths, so this normalized mean errors are not identical to the original alignment scores.

$$\frac{\sum(|\hat{y} - y| / \min\{m, n\})}{N}, \tag{1}$$

where $\hat{y}$ is the alignment score of an algorithm, y is the optimal global alignment score obtained by the NW algorithm; m and n are lengths of the best aligned sequence and the query sequence, respectively; and N is the number of queries.

In addition, the COVID-19 protein sequences are used to evaluate the performance for high identity sequences, as shown in Table 1(a). The EUMAT dataset is used to examine the performance for sequences of various lengths, as shown in Table 1(b). The NCBI human protein dataset is for examining the general distribution of sequences as shown in Table 1(c).

In Table 1(a), the FastSearch has better mean errors and it is faster than the GGSEARCH, so the FastSearch is more efficient and more accurate than the GGSEARCH with high identity sequences in the COVID-19 protein dataset. In Table 1(b), some filtering ratios of the FastSearch requires more time than the GGSEARCH because the average length in the EUMAT dataset is short. The BestJump stage in the FastSearch is more suitable for long sequences. The longer the average length is, the less time of the FastSearch than the GGSEARCH requires. In Table 1(c), the FastSearch is at least 25 times faster than the GGSEARCH, and the mean errors of the FastSearch are lower than the GGSEARCH. Therefore, the FastSearch is more efficient and more accurate than the GGSEARCH.

4 Conclusion

This paper proposes the FastSearch algorithm to perform fast search for global pairwise alignment in a protein sequence database. The FastSearch filters feasible sequences by using the

Table 1: Mean error comparisons of our FastSearch and the GGSEARCH algorithm [6]. (α, β) is the pair of filtering ratios for (FastLCS, BestJump) stages. The time (in seconds) is the average of all queries. (a) Proteins of COVID-19 (100 queries); (b) EUMAT (100 queries); (c) NCBI human proteins (50 queries).

Proteins of COVID-19		
(α, β)	Mean error	Time
(2,50)	0.00	27
(4,25)	0.00	18
(5,20)	0.00	16
(10,10)	1.81 E-6	14
(20,5)	1.77 E-5	13
(50,2)	3.76 E-5	12
GGSEARCH	2.14 E-4	1898

(a)

EUMAT		
(α, β)	Mean error	Time
(2,1500)	7.75 E-3	170
(5,600)	1.12 E-2	79
(10,300)	1.32 E-2	45
(15,200)	1.94 E-2	34
(30,100)	2.24 E-2	22
(100,30)	4.78 E-2	13
GGSEARCH	1.00	35

(b)

	Mean error (NCBI human protein)					
(α, β)	Human1	Human2	Human3	Human4	Human5	Time
(2,50)	0.00	0.00	0.00	0.00	0.00	53
(5,20)	7.14 E-5	1.12 E-6	0.00	1.42 E-5	2.25 E-6	22
(10,10)	1.02 E-4	5.69 E-5	0.00	1.42 E-5	2.25 E-6	16
(50,2)	1.72 E-3	3.02 E-3	5.78 E-6	4.41 E-4	5.33 E-5	11
(2,300)	2.33 E-4	4.05 E-5	0.00	0.00	0.00	62
(3,200)	2.33 E-4	1.12 E-6	0.00	5.63 E-4	0.00	33
(5,120)	2.35 E-4	1.12 E-6	0.00	1.42 E-5	2.25 E-6	22
(10,60)	2.66 E-4	5.69 E-5	0.00	1.42 E-5	2.25 E-6	14
GGSEARCH	4.31 E-2	1.52 E-3	5.65 E-2	4.72 E-2	6.61 E-2	1548

(c)

FastLCS and BestJump stages with heuristic strategies. Finally, the Needleman-Wunsch (NW) algorithm is used to find the best alignment score in the filtered sequences.

As the experimental results show, for longer protein sequences, the FastSearch get better accuracy and time efficiency than the GGSEARCH tool in the FASTA package. Hence the FastSearch can quickly find a good aligned sequence in a protein sequence database. In the future, for improving the proposed algorithm, we may use adaptive filtering ratios to improve the database search, modify the estimated way of pairwise alignment, or consider other parameters to compute scores of candidate positions.

References

[1] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman, "Basic local alignment search tool," *Journal of Molecular Biology*, vol. 215, no. 3, pp. 403–410, 1990.

[2] O. Gotoh, "An improved algorithm for matching biological sequences," *Journal of Molecular Biology*, vol. 162, no. 3, pp. 705–708, 1982.

[3] D. J. Lipman and W. R. Pearson, "Rapid and sensitive protein similarity searches," *Science*, vol. 227, no. 4693, pp. 1435–1441, 1985.

[4] F. Madeira, Y. M. Park, J. Lee *et al.*, "The EMBL-EBI search and sequence analysis tools APIs in 2019," *Nucleic Acids Research*, vol. 47, pp. W636–W641, 2019.

[5] S. B. Needleman and C. D. Wunsch, "A general method applicable to the search for similarities in the amino acid sequence of two proteins," *Journal of Molecular Biology*, vol. 48, no. 3, pp. 443–453, 1970.

[6] W. Pearson, "Finding protein and nucleotide similarities with FASTA," *Current Protocols in Bioinformatics*, vol. 53, pp. 3.9.1–3.9.25, 2016.

[7] E. W. Sayers, R. Agarwala, E. E. Bolton *et al.*, "Database resources of the National Center for Biotechnology Information," *Nucleic Acids Research*, vol. 46, pp. D8–D13, 2018.

[8] T. F. Smith and M. S. Waterman, "Identification of common molecular subsequences," *Journal of Molecular Biology*, vol. 147, no. 1, pp. 195–197, 1981.

[9] M. S. Waterman, T. F. Smith, and W. A. Beyer, "Some biological sequence metrics," *Advances in Mathematics*, vol. 20, no. 3, pp. 367–387, 1976.