효율적인 복수서열정렬 최적화기법 및 서열 분석 소프트웨어 개발

황재준° 김동회 엄상용 김진
한림대학교 정보통신공학부, 한림대학교 컴퓨터공학과
director@hallym.ac.kr° kdh@hallym.ac.kr suhmn@ekus.ce.hallym.ac.kr jinkim@hallym.ac.kr

Development of an efficient sequence alignment algorithm and sequence analysis software

JaeJun Hwang° DongHoi Kim SaangYong Uhmn Jin Kim
Divison of Information and Communication Engineering, Department of Computer Engineering

요 약

단백질들의 복수서열정렬은 단백질 서열간의 관계를 유추할 수 있는 유용한 도구이다. 최적화된 복수서열정렬을 얻기 위해 사용되는 가장 유용한 방법인 dynamic programming은 특정한 비용함수를 사용할 수 없기 때문에 특별한 경우 최적의 복수서열정렬을 제공하지 못하는 문제점이 있어 이를 해결하기 위하여 이 논문에서는 부분정렬 개선 기법을 사용한 알고리즘을 제안하였으며, 서열정렬을 하는 사용자가 윈도우 시스템의 GUI환경을 사용하여 서열정렬을 보다 편하게 할 수 있도록 우리가 제안한 알고리즘과 다양한 서열정렬 알고리즘을 및 여러 개의 서열포맷형식을 하나의 프로그램으로 통합한 서열정렬 및 편집 프로그램을 Visual C++ 사용하여 개발하였다.

1. 서 론

복수서열정렬은 세 개 또는 그 이상의 서열에서 동일한 서열을 찾는 유용한 기술이다. 이것은 분자의 구조, 기능, 발현 연구에 널리 사용된다.

이 논문에서는 복수서열정렬을 얻기 위한 최적화 알고리즘 가운데 가장 표준적인 알고리즘인 Needleman과 Wunch[1]에 의해 소개된 dynamic programming 기법의 단점인 특정 비용함수를 사용하지 못하는 문제점을 해결하기 위한 방법을 제시하였다.

서열정렬은 다양한 알고리즘으로 구현되어 있고 각각의 알고리즘 마다 프로그램이 개발 되어 있다. 개발된 프로그램들은 고유한 서열포맷형식을 가지고 있으며 대부분 텍스트 기반의 프로그램이다. 프로그램이 통합되어 있지 않아 서열정렬을 함에 있어 많은 수의 프로그램을 사용해야 되는 불편함이 있고, 프로그램마다 서열포맷형식이 각각 틀려 서로 간에 같은 파일을 사용할 수 없다. 또한 텍스트 기반의 프로그램은 실험 결과를 비교 평가하는데 많은 어려움과 불편함을 가지고 있다. 그러므로 우리는 서열정렬을 하는 대부분의 사용자들이 사용하는 윈도우 시스템에서 편하게 서열정렬을 할 수 있도록 윈도우 GUI 환경의 소프트웨어를 개발하였다. 개발된 소프트웨어는 Visual C++로 제작되었다.

본 논문의 구성은 2장에서는 복수서열정렬에 사용되는 비용함수를 소개하였고. 3장에서는 우리가 제안하는 알고리즘을 기술하였고. 4장에서는 Visual C++로 개발한 윈도우 기반의 서열 분석 소프트웨어 대한 소개를 하였으며 5장에서 결론을 내렸다.

2. 복수서열정렬 비용함수

작은 비용을 가지는 복수서열정렬은 큰 비용을 가지는 복수서열정렬보다 생물학적현상을 잘 표현한다. Altschul[16]은 복수서열정렬을 위한 몇 가지 비용함수를 분류하였는데, 이들 비용함수는 교체비용(substitution cost)와 갭비용(gap cost)으로 구성된다.

2.1 교체비용

본 논문에서 사용된 교체 비용은 SP(Sum of Pairs) 교체비용이다. SP 교체비용은 n개의 서열이 정렬되었을 때, $n(n-1)/2$개의 쌍의 교체비용의 합이다. 즉 복수서열정렬 A의 교체비용은 아래와 같이 계산될 수 있다.

$$\text{교체비용}(A)=\sum_{i=1}^{n-1}\sum_{j=i}^{n}\text{교체비용}(S_i', S_j')*weight(i,j)$$

이때 $weight(i,j)$는 서열 i와 j를 비교하여 얻은 비용에 대한 가중치이며, 단백질 서열간의 교체비용과 관련하여 주로 사용되는 비용 매트릭스는 Dayhoff matrix이다[17]. MSA에서는 PAM250 매트릭스를 사용한다.

2.2 갭비용

갭비용은 연속된 널('-')의 개수에 부과되는 비용이다. Altschul은 생물학적으로 자연스러운 갭 비용함수로써 natural gap cost를 제안하였다[16]. 특정복수서열 A의 점수는 교체비용과 갭비용을 계산하여 더하면 된다.

비용(A)=교체비용(A)+갭의 개수(A)*gap penalty

그러므로 최적의 정렬은 작은 비용을 가지는 정렬(A_{min})이다.

우리는 MSA에 의해 제공되는 정렬이 가진 점수를 $Opt_{quasi-natural}$로 정의한다. 또한 natural-gap cost를 적용하여 얻은 최적의 점수를 $Opt_{natural}$로 정의한다.

3. Dynamic Programming의 서열정렬을 refine하는 알고리즘

이 장에서는 MSA가 제공하는 $Opt_{quasi-natural}$을 가지는 서열정렬 중 완전히 포함되는 형태의 갭을 가질 수 있는 부분정렬만을 선택하고, natural gap cost의 갭 비용함수를 적용하여 $Opt_{natural}$표을 가지는 서열 정렬을 만드는 Sub-alignment Refinement Algorithm(SRA)방법에 대하여 논의한다.

3.1 Realignment

MSA에 의해 획득되는 서열정렬은 최적의 서열정렬과 매우 유사하다. 따라서 MSA가 만들어 낸 서열정렬에서 널이 위치한 부분정렬에 대해 최적의 정렬을 만들 수 있다면 쉽게 전체 서열을 최적 혹은 최적에 가깝게 정렬할 수 있다. 이 때

정제를 시도할 부분정렬은 다음 조건 (1)과 (2)를 만족하여야 한다.
(1) 갭이 존재하는 서열의 수가 2이상이어야 한다.
(2) 가장 많은 널의 개수와 가장 적은 널의 개수의 차가 2이상이어야 한다.

조건 (1)과 (2)는 갭들이 완전히 포함되기 위한 최소한의 조건이다.

먼저 부분 서열의 개수를 n, 길이를 l, i번째 서열에 포함되는 갭의 길이를 G_i 라 하자. 또한 부분 서열 내의 가장 긴 갭의 길이를 G_{max}라 하면, 만들어질 수 있는 가능한 부분 서열의 개수 K는 다음 식과 같다.

$$K = \Pi_{i=1}^{n} (l - G_i + 1)$$

새로이 만들어낸 부분정렬들에 대하여 natural gap cost를 적용하여 다시 계산한다. 이때 기존의 $Opt_{quasi-natural}$보다 작은 값을 가지는 부분정렬이 발견되면 기존의 부분정렬과 교체한다. 만일 MSA에 제공되는 정렬에 완전히 포함된 갭들이 k_2만큼 포함되어 있다면, 해당되는 갭의 개수만큼이 더 부가되어 있는 셈이 된다. 따라서 이러한 경우 natural gap cost 갭비용을 적용하기 위해 해당 갭의 개수만큼의 패널티를 감하여야 한다.

3.2 알고리즘의 분석

[과정1] MSA에서 획득된 서열정렬을 잘 보존된 지역을 이용하여 몇 개의 부분정렬로 나눈다.
[과정2] 나누어진 부분에 대하여 개선될 수 있는 지역을 찾는다.
[과정3] 발견된 지역에 대해 realignment, recalculation 방법을 적용한다.
[과정4] 더 낮은 비용을 가지는 부분정렬을 얻게 되면 기존의 부분정렬과 교체한다.

그림 1 요약한 알고리즘

본 논문에서 제안된 알고리즘을 요약하면 그림 1과 같다. 위의 알고리즘의 시간 복잡도는 [과정 3]에 의존하게 된다. 하나의 부분정렬의 비용을 계산하는데 $O(rl^2)$이 필요하므로, K개의 서열정렬에 대해서는 $O(K*rl^2)$의 시간이 필요하다. 이때 공간복잡도는 서열을 저장하기 위한 공간이외에 필요한 공간이 없으므로 [과정 3]의 공간복잡도는 $O(l*n)$이다.

그림 1을 보면 $Opt_{quasi-natural}$과 $Opt_{natural}$의 관계와 우리가 제시한 알고리즘의 이론적인 근거를 찾을 수 있을 것이다.

4. 서열 분석 소프트웨어 개발

우리는 서열정렬을 위한 프로그램을 개발하였다. 이 프로그램은 논문에서 제안한 알고리즘과 서열정렬을 위해 잘 알려진 MSA와 ClustalW 알고리즘이 구현되어 있다. 사용자들은 서열에 다양한 알고리즘을 적용할 수 있으며 각 알고리즘의 파라미터 값을 사용자가 설정 할 수 있게 되어있다. 이 프로그램은 알고리즘의 결과를 사용자가 분석하기 편한 화면으로 구성하여 볼 수 있게 되어있다. 앞으로 우리는 복수서열정렬을 위해 널리 사용되어지는 다른 알고리즘들을 이 프로그램에 구현할 것이다. 앞으로 프로그램의 구현 화면과 기능에 대한 간단한 설명을 하겠다.

이 프로그램에는 File의 입출력 기능, 서열 문자의 삽입, 삭제와 같은 편집 기능, 여러 개의 서열정렬 알고리즘을 구현한 서열정렬 기능, 서열정렬 결과 분석에 편의성을 제공하기 위한 다양한 화면 출력 기능이 있다.

프로그램은 ClustalW와 MSA가 지원하는 다양한 서열포맷 형식을 지원하고 있다.

프로그램은 다중윈도우로 구성되어 있고 윈도우를 분할하여 좌측은 서열의 정보를 보여주고 우측에는 서열의 내용을 보여 주고 있다.

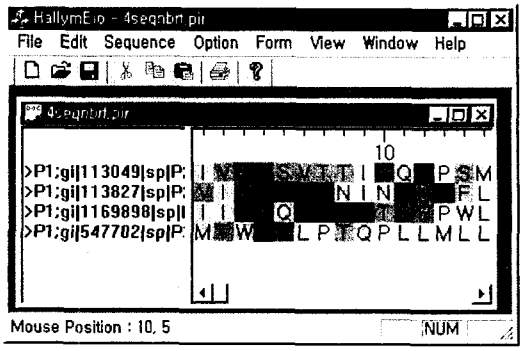

그림 2 서열 입력 윈도우

그림 2에서 보이는 거와 같이 입력된 서열의 문자는 색깔별로 구분하여 보여주고 있고 화면 상단에는 서열의 길이를 나타내는 룰러를 좌측 하단에는 서열의 위치를 보여주어 사용자가 서열을 비교 분석 하는데 편리함을 제공해 준다. 문자의 색 및 화면의 구성은 사용자가 보기 편하게 구성 하여 볼 수 있다.

이 프로그램은 입력된 서열문자의 삽입, 삭제, 잘라내기, 붙여넣기와 같은 편집기능을 제공하는데 일반 문서 편집기의 기능이 아닌 특정 영역의 편집과 같은 서열 전용 편집기의 기능을 제공한다.

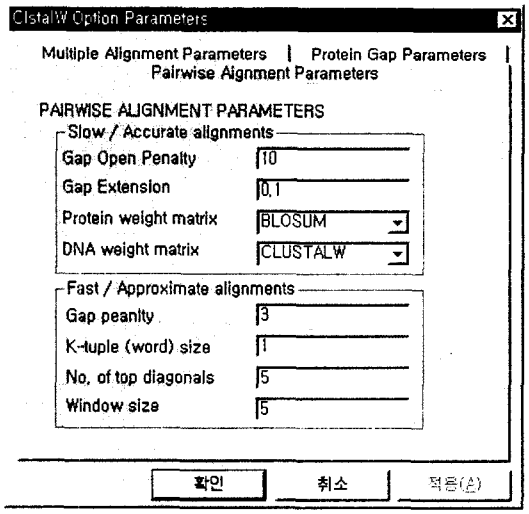

그림 3 ClustalW 파라미터 윈도우

우리는 다양한 복수서열정렬 알고리즘의 파라미터 값을 설정할 수 있게 하였다. 그림 3은 ClustalW 파라미터 값 설정 윈도우 화면을 보여주고 있다.

그림 4, 그림 5는 우리가 개선한 MSA 알고리즘과 ClustalW의 결과 화면이다. 그림에서 보여주는 것 같이 우리는 플랫폼의 변경 없이 같은 입력 서열의 집합에 다양한 알고리즘을 적용 할 수 있다.

5. 결론 및 향후과제

본 논문에서는 복수서열정렬의 최적화 기법에 대하여 논의하였다. Dynamic programming 알고리즘은 최적의 복수서열

정렬을 제공하는 가장 널리 사용되는 알고리즘이나, 속성상 natural gap cost를 적용할 수 없는 것이 가장 결정적인 문제점이었다. 이러한 문제점을 해결하기 위한 다양한 방법이 제안되었지만, 어느 논문에서도 $Opt_{natural}$과 $Opt_{natural}$ 의 관계에 대한 이론적인 근거를 제시하지 못하였다. 본 논문에서 제안하는 개선 알고리즘을 사용하여 natural gap cost를 사용한 최적의 복수서열정렬을 얻을 수 있었으며, $Opt_{natural}$의 하한값에 대한 이론적인 근거를 제시하였다. 본 논문에서 제공한 알고리즘을 사용하면 손쉽게 MSA의 서열정렬을 개선한 최적의 서열정렬을 얻을 수 있었다.

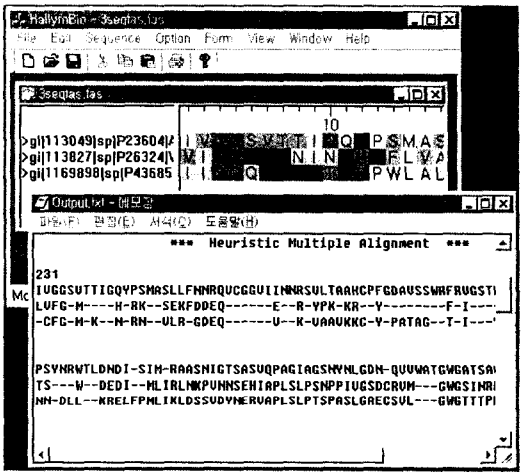

그림 4 개선된 MSA 결과

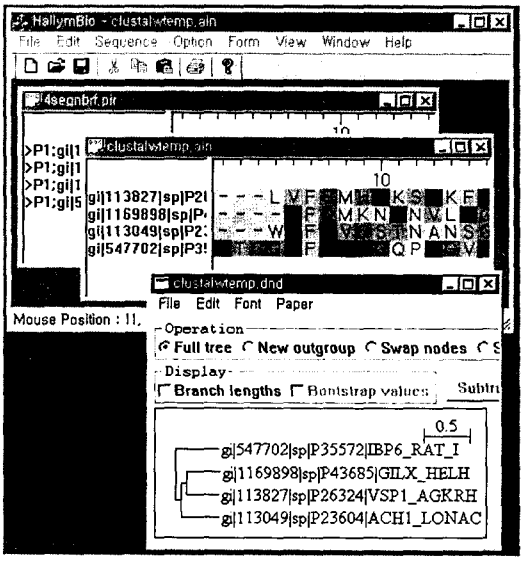

그림 5 ClustalW 결과

구현에서 우리는 다른 복수서열정렬 알고리즘들을 우리의 소프트웨어에 통합하였다. 이 패킷지 안에 널리 사용되어지는 여러 개의 복수서열정렬 알고리즘의 통합함으로써 사용자는 정렬 소프트웨어 또는 웹 사이트의 변경 없이 쉽게 사용할 수 있고, 윈도우의 GUI 환경을 이용하여 보다 편리하게 복수서열정렬의 결과를 분석 비교 평가 할 수 있으며 바로 서열문자를 편집할 수 있다는 이점을 가질 수 있다.

참고문헌

[1] Needleman. S. B and Wunch, C. D. "A general method applicable to the search for similarities in the amino acid sequence of two proteins" *J. Molec. Biol.*, Vol. 48, pp. 443-453, 1970.

[2] Feng, D. F and Johnson, M. S. and Doolittle, R. F. "Aligning amino acid sequences: comparison of commonly used methods," *J. Molec. Evol.*, Vol.21, pp. 112-125, 1982.

[3] Ficket, J. W. "Fast optimal alignment," *Nucl. Acids Res.*, Vol.12 pp.175-180, 1984.

[4] Sankoff, D. and Kruskal, J. B. "Time Warps, String Edits and Macromolecules: The theory and practice of Sequence Comparison," *Addison-Wesley, Reading, MA*, 1983.

[5] Taylor, W. R. "Multiple sequence alignment by a pairwise algorithm," *CABIOS*, Vol.3, pp.81-87, 1987.

[6] Martinez, H. M. "A flexible multiple sequence alignment program," *Nucl. Acids. Res.*, Vol16, pp.1683-1691, 1988.

[7] Sankoff, D. "Simultaneous comparison of three or more sequences related by a tree," *Addison-Wesley, Reading, MA*, 1983.

[8] Notredame, C. and Higgins, D. "SAGA:sequence alignment by genetic algorithm," *Nucl. Acids. Res.*, Vol.24, No.8, pp.1515-1524, 1996

[9] Chan, S. C. C. and Wong, A. K. and Chiu, D. K. Y. "A survey of multiple sequence comparison methods," *Bull. Math. Bio.*, Vol.43, pp.563-598, 1992

[10] Feng, D. F. and Doolittle, R. F. "Progressive sequence alignment as a prerequisite to correct phylogenetic trees," *J. Molec. Evol.*, 25:351-360, 1987.

[11] Murata, M. and Richardson, J. S and Sussman. J. L. "Simultaneous comparison of three protein sequences," In Proc. *Natl. Acad. Sci.* USA., Vol.82, pp.3073-3077, 1985.

[12] Altschul, S. F. and Lipman, D. J. "Threes, stars, and multiple biological sequence alignment," *SIAM J. appl. Math.*, Vol.49 pp.197-209, 1989.

[13] Kim, J. and Pramanik, S. "An efficient method for multiple sequence alignment," *In Second International Conference on Intelligent Systems for Molecular Biology*, 1994.

[14] Kim, J. and Pramanik, S. and M. J. Chung. "Multiple sequence alignment using simulated annealing," *CABIOS*, Vol.10, pp.419-426, 1994.

[15] Lipman, D. J. and Altschul, S. F. and Kececioglu, J. D. "A tool for multiple sequence alignment," *Proc. Natl. Acad. Sci. USA.*, Vol.86, pp. 4412-4415, 1989.

[16]Altschul, S. F. "Gap costs for multiple sequence alignment," *J Theor. Biol.*, Vol.138 pp.297-309, 1989.

[17] Dayhoff. M. O. "A model of evolutionary change in proteins. matrices for detecting distance relationships," *In Atlas of Protein sequence an Structure*, Vol. 5 suppl.3, pp.354-352. Dayhoff. M. O.(ed) Washington. DC: National Biomedical Research Foundation, 1978.

감사의 글

본 연구는 정보통신부 정보통신선도기반기술개발사업의 지원에 의하여 이루어진 것임.