

유전자 정렬 및 재배열 알고리즘

서울대학교 한보형* · 박근수**

부산대학교 김동규***

1. 서론

생물정보학(bioinformatics)은 생명과학분야의 문제들에 대하여 정보, 컴퓨터공학에서 개발된 신기술을 적용하는 새로운 분야이다. 이 가운데 계산 분자 생물학(computational molecular biology)은 생물정보학의 여러 분야 중 가장 먼저 활발히 연구되기 시작한 분야이다. 인간의 유전정보를 가지고 있는 DNA는 4종류의 염기들로 구성된 이중나선형 구조로서, 이는 문자 집합이 $\{A, C, G, T\}$ 이고 길이가 30억 정도의 하나의 스트링(string)으로 간주할 수 있다. 그러므로 계산 이론(computation theory)의 한 분야인 스트링 처리 분야에서 개발된 매칭알고리즘, 근사 매칭알고리즘, 유사도 분석, 집합 매칭알고리즘 등의 여러 기법들은 DNA를 분석할 때 발생하는 문제들의 해결에 직접적으로 사용될 수 있다. 이처럼 유전자 분석 시 발생한 실제적인 문제 해결을 위하여 효율적인 스트링 처리 알고리즘을 개발하는 분야를 계산 분자 생물학이라고 한다.

현재 계산 분자 생물학에서 유전자 서열 분석을 위하여 가장 활발히 연구되고 있는 문제들 중에는 유전자 정렬(alignment) 문제와 유전자 재배열(rearrangement) 문제 등이 있다. 생명체의 유전정보는 세포의 염색체내에 있는 DNA에 집약되어 있고, 이 DNA는 유전정보를 가지는 유전자(gene)들로 구성되며, 이들 유전자의 집합을 유전체(genome)라고 한다. 총 30억 정도의 길이를

가지는 인간의 DNA에는 대략 10만개 정도의 유전체가 존재한다. 최근 들어 DNA 칩(chip) 개발 등의 연구성으로 방대한 양의 유전자 서열을 처리하고 분류, 분석하는 효율적인 알고리즘의 개발이 요구되고 있다.

유전자 정렬 문제는 주어진 유전자들의 서열을 비교하여 유전자 서열간의 유사성 정도를 판단하는 문제이다. 이 문제의 효율적인 알고리즘은 생물체간 유전자들의 연관성을 계산하여, 새로 밝혀진 생물체의 유전자의 구성과 진화관계를 밝혀내거나 전체 염기서열에서 특정한 유전자의 위치를 찾는 데에 기여하고 있다. 또한, 유전자 재배열 문제는 한 생물체가 가지는 유전체의 재배열을 통하여 다른 생물체의 유전체로 재구성이 가능한지를 다루는 문제로, 여러 생물체간의 유전체 내에서 유전자 구성의 연관성을 찾아내고 비교함으로써 유전체의 변형과 진화 여부를 결정하는데 많은 도움을 주고 있다.

본 논문에서는 2장에서 유전자 정렬문제를, 3장에서 유전자 재배열 문제에 대해 설명하고, 4장에서 결론을 맺고자 한다.

2. 유전자 정렬

생물들의 DNA, RNA, 아미노산 등에 있는 유전자의 서열상에 유사성이 발견된다면, 그 생물들간에 기능적, 구조적 유사성이 있다고 말할 수 있다. 현재 생물이 가지고 있는 유전정보는 이전부터 존재했던 생물로부터 물려받은 정보에 새로운 요소가 일부 추가된 것으로 볼 수 있으므로, 같은 종에 있는 생물들의 유전자 서열은 매우 유사한 경우가 많으며, 그 유사성의 정도에 따라

*학생회원

**중신회원

***정회원

그 생물들이 어떻게 진화해 왔는지 밝힐 수 있다. 유전자 정렬 알고리즘은 하나의 유전자 내에 존재하는 유전자 서열을 다른 생물의 유전자 서열과 비교하여 유사성을 발견함으로써, 두 생물의 생물학적 상관관계를 알아보는 것을 주목적으로 하고 있다.

생물의 유전정보는 4 종류의 염기들의 배열순서에 의하여 결정되므로, 본 논문에서는 문자집합 $\Sigma = \{A, C, G, T\}$ 상의 문자들로 이루어진 임의의 스트링을 시퀀스(sequence)라고 정의한다. 이 장에서는 유전자 서열을 비교하기 위하여 두 개의 유전자 시퀀스를 비교하는 시퀀스 정렬(sequence alignment) 문제와 여러 개의 유전자 시퀀스를 비교하는 다중 시퀀스 정렬(multiple sequence alignment) 문제를 정의하고 연구결과를 설명하고자 한다.

2.1 시퀀스 정렬

시퀀스 정렬은 두 개의 시퀀스 S_1 과 S_2 가 있을 때, 두 개의 시퀀스의 중간 혹은 양끝에 공백(space)이 들어갈 위치를 선택하고 삽입하여 두 스트링이 같은 길이가 되도록 정렬하는 것이다. 그림 1은 두 개의 시퀀스 **AAGTCA**와 **ACTGC**가 있을 때 두 시퀀스를 정렬한 결과를 보여준다. 여기서 - 로 표시된 부분은 공백을 나타낸다. 그림 1의 시퀀스 정렬 중, 1열에서와 같이 동일한 문자가 한 열에 위치하는 경우를 일치(match)라 하고, 2열에서와 같이 서로 다른 문자가 한 열에 위치하면 불일치(mismatch)라고 한다.

	1	2	3	4	5	6	7
S1:	A	A	G	T	-	C	A
S2:	A	-	C	T	G	C	-

그림 1 시퀀스 정렬의 예

시퀀스 정렬 문제는 주어진 두 개의 시퀀스 S_1 과 S_2 가 있을 때 모든 열에서 불일치를 최소화하는 정렬 방법을 찾는 문제이다. 이 문제는 스트링 처리분야에서 오랫동안 연구되어온 편집거리(edit distance)를 찾는 알고리즘으로 해결될

수 있는데, 편집 거리란 두 시퀀스 S_1 과 S_2 에 대하여 S_1 을 S_2 로 바꾸는데 필요한 최소의 편집연산(edit operation)의 총수로 정의되며, 여기서 편집연산이란 다음의 아래의 세 연산들 중 하나에 해당된다.

- (1) 삽입(insertion) : S_2 에 하나의 문자를 삽입
- (2) 삭제(deletion) : S_1 의 하나의 문자를 삭제
- (3) 변경(change) : S_1 의 하나의 문자를 S_2 의 다른 문자로 변경

그림 1의 예에서는 2열과 7열에서 **A**가 삭제되고, 3열에서 **G**가 **C**로 변경이 발생하고, 5열에서 **G**가 삽입되고 있으므로 두 시퀀스 S_1 과 S_2 의 편집거리는 4이다. 시퀀스 정렬 문제는 편집 거리를 구하는 문제와 동일하므로 그림 2와 같이 동적 프로그래밍(dynamic programming) 기법을 이용하여 해결한다.

두 개의 시퀀스 S_1 과 S_2 에 대하여, $D(i, j)$ 값은 $S_1[1...i]$ 와 $S_2[1...j]$ 사이의 편집 거리로서 S_1 의 i 번째 문자까지를 S_2 의 j 번째 문자까지로 바꿔주기 위한 삽입, 삭제, 및 변경 연산회수의 최소값을 의미한다. 따라서, 길이가 n 인 시퀀스 S_1 과 길이가 m 인 시퀀스 S_2 의 편집 거리는 $D(n, m)$ 이다. 이와 같이 편집 거리를 구하고 삽입, 삭제, 변경 연산이 어떻게 사용되었는지는 그림 2와 같은 테이블을 이용하여 역추적함으로써 그림 1과 같이 시퀀스를 정렬할 수 있다.

$D(i, j)$	S_1	A	A	G	T	C	A	
	0	1	2	3	4	5	6	
S_2	0	0	1	2	3	4	5	6
A	1	1	0	1	2	3	4	5
C	2	2	1	1	2	3	3	4
T	3	3	2	2	2	2	3	4
G	4	4	3	3	2	3	3	4
C	5	5	4	4	3	3	3	4

그림 2 두 시퀀스의 편집거리

2.2 다중 시퀀스 정렬

분자 생물학에서 중요 분야 중의 하나는

DNA, RNA, 또는 단백질(protein) 서열을 비교하고, 그 서열간의 관계를 추론하는 것이다. 이를 위해 최근 분자생물학에서 시퀀스에 대한 연구 가운데 활발히 진행되고 있는 것 중의 하나가 다중 시퀀스 비교(multiple sequence comparison) 문제이다. 다중 시퀀스 비교를 통해 나타나는 유전자 서열의 공통점을 찾아냄으로써 생물의 진화역사, DNA나 단백질의 분자구조를 밝힐 수도 있다. 또한, 서로 다른 생물들의 부분적인 유전자 서열의 유사성은 특정 유전자 서열의 생물학적 기능을 밝히는데 매우 유용하다.

다중 정렬(multiple alignment)의 정의는 다음과 같다: $k > 2$ 개의 시퀀스들의 집합 $S = \{S_1, S_1, \dots, S_k\}$ 의 다중 시퀀스 정렬은 각 k 개의 스트링에서 공백(space)을 적절한 위치에 삽입함으로써, 모든 시퀀스의 길이가 l 로 같게 하여 k 개의 시퀀스 모두를 정렬하는 것이다. 즉, k 개의 시퀀스를 어떤 기준에 따라 $k \times l$ 의 배열의 각 행에 정렬하여 넣는 문제이다. 그림 3은 4개의 스트링이 다중 시퀀스 정렬되어 있는 경우의 예를 보여 준다.

1:	A	A	G	A	A	-	A
2:	A	T	-	A	A	T	G
3:	C	T	G	-	G	-	G
4:	A	T	G	A	A	-	G

그림 3 4개의 DNA 시퀀스에 대한 다중 시퀀스 정렬 예

시퀀스들을 정렬하기 위해서 사용하는 기준을 목적 함수(objective function)라고 부른다. 이러한 목적 함수에는 SP(sum-of-pairs) 함수, 일치(consensus) 함수, 트리(tree) 함수 등이 있는데, 비교적 간단하면서도 많이 사용되고 있는 SP 함수에 대해서 설명하면 다음과 같다. 다중 시퀀스 정렬 M 에 대한 SP 점수(score)는 M 에 의하여 계산되는 모든 두 시퀀스의 쌍에 대한 점수(score)의 총합으로 정의되며, SP 정렬 문제(SP alignment problem)란 SP 점수를 최소화하는 다중 시퀀스 정렬을 찾는 문제이다. 최근 Wang과 Jiang[16]은 SP 정렬 문제는 NP-complete임을

증명하였다. 그러나, SP 점수를 기준으로 사용하지 않는 일반적인 다중 정렬 문제가 NP-hard인지는 아직 증명되지 않았다.

SP 정렬 문제를 해결하는 방법 역시 동적 프로그래밍 기법을 사용한다. 그러나, 동적 프로그래밍을 사용하여 길이가 n 인 k 개의 시퀀스를 다중 시퀀스 정렬하기 위해서는 $O(n^k)$ 시간이 걸린다. 따라서, 이 분야의 연구는 다항시간 내에 문제를 해결할 수 있는 근사 알고리즘을 구하는 것이 매우 중요하다. Gusfield[9]가 최초로 다항시간 근사 알고리즘을 제안하였는데, 이 알고리즘의 근사비는 입력 시퀀스의 개수가 k 일 때 $2 - \frac{2}{k}$ 이다. Pevzner[15]는 Gusfield의 알고리즘을 개선하여 $2 - \frac{3}{k}$ 근사 알고리즘을 발표하였다. 최근에는 Bafna와 Pevzner[3]가 고정된 값 l 에 대해 $2 - \frac{l}{k}$ 근사 알고리즘을 발표하였다.

3. 유전자 재배열

유전자 정렬 알고리즘은 주로 유전자 단위의 시퀀스 비교에 중점을 두었지만, 최근에는 염색체나 유전체와 같이 유전자보다 더 큰 단위의 변형에도 많은 관심을 기울이고 있다. 1980년대 후반에 Palmer와 Herbon은 Brassica oleracea (cabbage)와 Brassica campestris(turnip)의 미토콘드리아 유전체를 비교하여 본 결과 유전자 구성은 상당히 비슷하지만 그 배열 상태가 매우 다르다는 것을 밝혀내었다. 이 사실은 한 생물체의 유전자 시퀀스를 재배열하여 다른 생물의 유전자 시퀀스를 얻어내는 과정이 두 생물체간의 진화관계를 밝히는데 매우 중요하다는 것을 의미한다.

유전자 재배열 문제란 두 시퀀스에 대하여, 전도(reversal), 전치(transposition), 전위(translocation) 등의 연산을 사용하여 유전자를 재배열함으로써 주어진 유전체를 다른 유전체로 변환하는 문제이다. 여기서 전도 연산과 전치 연산은 하나의 염색체 내에서만 고려하는 연산인 반면, 전위 연산은 여러 개의 염색체 간에 사용될 수 있는 연산이다. 이 장에서는 이 연산들을 수학적으로 정의하고 현재까지의 연구결과와 앞으로의 과제 등을 설명하고자 한다.

3.1 유전자 재배열 문제의 유형별 정의

어떤 유전자 시퀀스 $\pi = (\pi_1 \pi_2 \dots \pi_n)$ 이 $\{1, 2, \dots, n\}$ 의 순열이라고 할 때, 전도연산 $r(i, j)$ ($1 \leq i < j \leq n+1$)는 시퀀스의 i 번째부터 $j-1$ 번째까지 원소들의 순서를 역으로 바꿔주는 연산이다.

$$\begin{pmatrix} 1 & \dots & i-1 & i & i+1 & \dots & j-1 & j & \dots & n \\ 1 & \dots & i-1 & j-1 & \dots & i+1 & i & j & \dots & n \end{pmatrix}$$

임의의 두 개의 순열 $\pi = (\pi_1 \pi_2 \dots \pi_n)$, $\sigma = (\sigma_1 \sigma_2 \dots \sigma_n)$ 과 연산 ρ_1, \dots, ρ_t 에 대하여 $\pi \cdot \rho_1 \cdot \rho_2 \dots \rho_t = \sigma$ 의 관계가 성립할 때, 두 순열사이의 재배열 거리는 연산의 개수 t 의 최소값을 의미한다. 이 때, $\sigma^{-1}\pi$ 는 정렬된 순열 $I = (12 \dots n)$ 과 같으므로 π 와 σ 사이의 거리는 $\sigma^{-1}\pi$ 와 I 와의 거리와 같다. 따라서, 임의의 두 순열사이의 재배열 거리를 구하는 문제는 하나의 순열 π 와 I 사이의 재배열 거리를 구하는 문제로 바꾸어 생각할 수 있다. 따라서, 전도 연산을 이용한 유전자 재배열 문제는 임의의 주어진 시퀀스 π 를 전도 연산을 이용하여 단위 시퀀스 I 로 재배열하는 문제이다.

이 문제는 부호가 있는 시퀀스에 대해서도 적용될 수 있다. 부호가 있는 시퀀스란 시퀀스내의 각각의 원소, 즉 유전자가 방향성을 가지고 있는 것으로서 그 방향성을 부호로 나타낸다. 부호가 있는 시퀀스의 전도 연산에 의한 재배열 문제는 임의의 부호가 있는 시퀀스 π 를 전도 연산을 이용하여 부호가 있는 단위 시퀀스 $I = (+1+2+\dots+n)$ 으로 바꾸는 문제이다. 부호가 있는 시퀀스에 전도 연산을 수행하면 그림 4와 같이 원소들의 순서뿐만 아니라 부호도 모두 바뀐다.

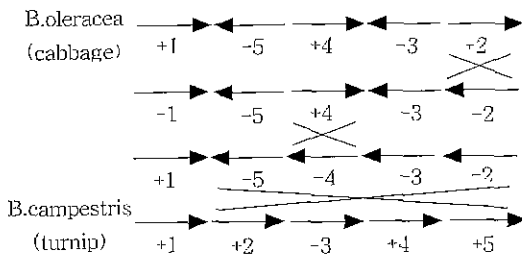


그림 4 부호가 있는 순열의 정렬의 예

전치 연산 $t(i, j, k)$ 는 다음과 같이 $1 \leq i < j \leq n+1$ 이고 $1 \leq k \leq n+1$ ($k \notin [i, j]$) 일 때, 유전자 시퀀스 π 내의 구간 $[i, j-1]$ 을 $k-1$ 번째 원소와 k 번째 원소 사이로 위치를 바꿔주는 연산이다.

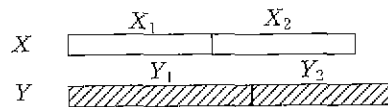
$$\begin{pmatrix} 1 & \dots & i-1 & i & \dots & j-1 & j & \dots & k-1 & k & \dots & n \\ 1 & \dots & i-1 & j & \dots & k-1 & i & \dots & j-1 & k & \dots & n \end{pmatrix}$$

전치 연산에 대한 유전체 재배열 문제는 임의의 주어진 시퀀스 π 를 전치연산을 이용하여 단위 시퀀스 I 로 재배열하는 문제이다.

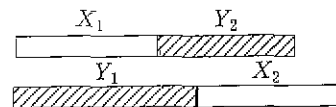
전도 및 전치 연산 $r(i, j, k)$ 는 $1 \leq i < j \leq n+1$ 이고 $1 \leq k \leq n+1$ ($k \notin [i, j]$) 일 때, 유전자 시퀀스 π 내의 구간 $[i, j-1]$ 을 $k-1$ 번째 원소와 k 번째 원소 사이로 위치를 바꾸고 그 순서도 역으로 만들어주는 연산이다. 전도 및 전치 연산을 이용한 유전자 재배열 문제는 임의의 주어진 시퀀스 π 를 전도, 전치, 전도 및 전치 연산을 이용하여 단위 시퀀스 I 로 재배열하는 것이다.

전도 및 전치연산에 대한 유전자 재배열 문제는 주로 부호가 있는 순열에 대해서 연구되고 있다. 따라서, 전도연산 또는 전도 및 전치연산이 사용되는 경우에는 연산이 적용되는 부분의 부호가 바뀐다. 그러나, 전치 연산만 사용될 때에는 부호가 바뀌지 않는다.

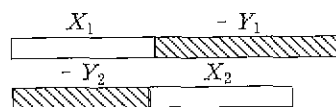
전위 연산은 두 개 이상의 염색체가 존재할 때, 염색체 내의 유전자 시퀀스 일부분을 서로 교환하는 연산이다. 전위 연산에는 두 가지가



(a) 기존의 염색체 X, Y



(b) 접두부-접두부 전위 연산



(c) 접두부-접미부 전위 연산

그림 5 두 염색체 간의 전위 연산

있는데, 접두부-접두부(prefix-prefix) 전위 연산은 두 염색체의 접두부를 서로 맞교환하는 연산이고, 접두부-접미부(prefix-suffix) 전위 연산은 한 염색체의 접두부와 다른 염색체의 접미부를 교환하는 연산이다. 그림 5에서는 전위연산의 예를 나타내고 그림에서 - 부호는 유전자 방향성을 나타낸다.

따라서, 전위 연산에 의한 유전자 정렬 문제는 두 개 이상의 임의의 주어진 시퀀스 $X = (x_1, x_2, \dots, x_l)$, $Y = (y_1, y_2, \dots, y_m)$, $Z = (z_1, z_2, \dots, z_n)$ 등이 있을 때, 이 모든 시퀀스 조합에 전위 연산을 수행하여 모든 시퀀스를 단위시퀀스 I 로 재배열하는 문제이다.

3.2 유전자 재배열 문제의 알고리즘

유전자 재배열 문제에서는 재배열 거리, 즉 필요한 연산 회수의 최소값을 구하는 것이 가장 중요한 요소이다. 따라서, 앞에서 설명한 각각의 연산에 대해 재배열 거리를 찾는 효율적인 알고리즘의 개발이 계속 진행 중이다.

전도 연산에 대한 유전자 재배열 문제에 대한 연구 결과는 다음과 같다. 먼저 Kececioglu와 Sankoff[14]는 처음으로 전도 연산에 의한 유전자 재배열 문제에 대한 2 근사 알고리즘을 제시하였고, Caprara[5]는 전도 연산에 의한 유전자 재배열 문제가 NP-hard라는 것을 증명하였다. 그리고, Bafna와 Pevzner[2]는 7/4 근사 알고리즘을 발표하였고, Christie[6]는 근사비를 더 향상시킨 3/2 근사 알고리즘을 제시하였다.

부호가 있는 전도 연산에 의한 유전자 재배열 문제는 이와는 별도로 연구되었다. Hannenhalli와 Pevzner[12]는 부호가 있는 시퀀스의 전도연산에 대한 재배열 문제의 $O(n^4)$ 알고리즘을 발표하였고, Berman과 Hannenhalli[4]는 $O(n^2 \alpha(n))$ 알고리즘을 발표하였는데, 여기서 $\alpha(n)$ 은 Ackerman 함수의 역함수이다. 그리고, Kaplan, Shamir와 Tarjan[10]은 $O(n^2)$ 알고리즘을 발표하였다.

전도연산에 의한 시퀀스 재배열 문제에 비해 다른 문제들에 대한 연구성과는 그리 많이 발표되지는 않았다. Bafna와 Pevzner[1]는 전치연산에 의해 재배열하는 1.5 근사 알고리즘을 발표하

였다. 그리고, Gu, Peng, Sudborough[7]는 부호가 있는 시퀀스에 대하여 전도 및 전치연산에 의해 재배열하는 2 근사 알고리즘을 발표하였다. 한편, 이 문제에 대해 휴리스틱을 사용함으로써 실험적으로 근사비가 거의 1에 가깝게 나온다는 연구결과가 발표되었다[17]. 이 결과는 이 문제가 이론적으로 더 좋은 근사비를 가지고 있을 가능성이 크다는 것을 시사하고 있다.

전위 연산에 대한 시퀀스 재배열 문제에 대해서는 Kececioglu와 Ravi[13]가 2 근사 알고리즘을 발표하였고, Hannenhalli[11]는 전위 연산을 이용한 재배열 문제의 $O(n^3)$ 알고리즘을 제시하였다. 한편 Kececioglu와 Ravi[13]는 전위 연산과 전도 연산을 모두 사용하는 1.5 근사 알고리즘을 발표하였다.

전도연산에 의한 재배열 문제에 대한 연구는 이미 많은 성과를 거두고 있고, 전위 연산에 대해서도 다항시간 알고리즘이 이미 알려졌다. 그러나, 전치연산과 전도 및 전치연산에 대한 재배열 문제에 대해서는 아직 다항시간 알고리즘이 존재하지 않으며 NP-hard 인지도 증명되지 않았다.

4. 결론

본 논문에서는 유전자 정렬 문제와 재배열 문제를 정의하고 해결 방법을 기술하였다. 유전자 정렬 문제로는 편집 거리를 계산하는 알고리즘으로 해결이 가능한 시퀀스 정렬 문제를 설명하였고, SP 함수를 이용한 다중 시퀀스 정렬 문제를 정의하고 연구 결과를 기술하였다. 또한, 전도, 전치, 전위 등의 연산을 사용한 유전자 재배열 문제를 정의하였으며 연구 동향과 향후 연구 방향에 대하여 기술하였다.

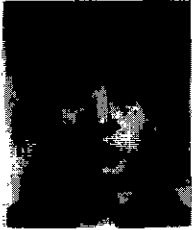
본 논문에서는 유전자 정렬과 재배열 문제 등과 같이, 분자생물학 분야에서 유전자의 분석을 수행 중에 발생하는 실제적인 문제가 계산이론분야의 스트링처리 알고리즘을 이용하여 어떻게 효율적으로 해결될 수 있는 가를 보이고 있다. 여기에서 설명한 대표적인 두 문제 이외에도 유전자 분석에서 발생하는 많은 문제들이 스트링 처리를 위하여 연구된 다양한 알고리즘과 접미사 트리(suffix tree), 접미사 배열(suffix array) 등의 자료구조를 이용하여 효율적으로 해결되어진다. 그

리므로, 유전정보에 숨겨진 생명의 신비를 밝혀 내려면, 기초 분야인 분자생물학 분야와 응용분야인 의학 및 약학 분야와 더불어 신기술이 개발되고 있는 정보, 컴퓨터 공학분야와의 협조적인 공동 연구가 활성화되어야 할 것이다.

참고문헌

- [1] V. Bafna, P. Pevzner, Sorting by transpositions, *SIAM J. Discrete Math.*, pp. 224-240, 1998.
- [2] V. Bafna, P. Pevzner, Genome rearrangements and sorting by reversals, *SIAM J. Comput.* 25 (2), pp. 272-289, 1996.
- [3] V. Bafna and P. Pevzner, Approximate methods for multiple sequence alignment, Manuscript, 1993.
- [4] P. Berman and S. Hannenhalli, Fast sorting by reversals, In *Proc. Combinatorial Pattern Matching(CPM'96)*, pp. 168-185, 1996.
- [5] A. Caprara, Sorting by reversals is difficult, In *Proc. of the First International Conference on Computational Molecular Biology(RECOMB'97)*, pp. 75-83, 1997.
- [6] D. Christie, A $3/2$ -Approximation Algorithm for Sorting by Reversal, In *Proc. ACM-SIAM Symp. on Discrete Algorithms (SODA'98)*, pp. 244-252, 1998
- [7] Q. Gu, S. Peng, H. Sudborough, A 2-approximation algorithm for genome rearrangements by reversals and transpositions, *Theoretical Computer Science*, pp. 327-339, 1999.
- [8] D. Gusfield, *Algorithms on Strings, Trees, and Sequences*, Cambridge University Press, 1997.
- [9] D. Gusfield, Efficient methods for multiple sequence alignment with guaranteed error bounds, *Bulletin of Mathematical Biology* 55, pp. 141-154, 1993.
- [10] H. Kaplan, R. Shamir, R. Tarjan, Faster and simpler algorithm for sorting signed permutation by reversals, In *Proc. ACM-SIAM Symp. on Discrete Algorithms (SODA'97)*, pp. 344-351, 1997. also: *Proc. RECOMB'97*.
- [11] S. Hannenhalli, Polynomial algorithm for computing translocation distance between genomes, *Discrete Applied Mathematics* 71, pp. 137-151, 1996.
- [12] S. Hannenhalli, P. Pevzner, Transforming cabbage into turnip (polynomial algorithm for sorting signed permutation by reversals), *Proc. 27th ACM symposium on Theory of Computing(STOC'95)*, pp. 178-189, 1995.
- [13] J. Kececioglu, R. Ravi, Of mice and men: Evolutionary distances between genomes under translocation, In *Proc. of 6th Annual ACM-SIAM Symp. on Discrete Mathematics(SODA'95)*, pp. 604-613, 1995.
- [14] J. Kececioglu, D. Sankoff, Exact and approximation algorithms for the inversion distance between two permutations. *Algorithmica* 13, pp. 180-210, 1995.
- [15] P. Pevzner, Multiple alignment, communication cost, and graph matching, *SIAM J. Applied Math* 56(6), pp. 1763-1779, 1992.
- [16] L. Wang, T. Jiang, On the Complexity of Multiple Sequence Alignment, *Journal of Computational Biology* 1, pp. 337-348, 1994.
- [17] 한보형, 박근수, 전도와 전치 연산을 이용하여 정렬하는 빠르고 간단한 알고리즘, 정보과학회 춘계 학술대회, pp. 674-676, 2000.

한 보 형



1997 서울대학교 컴퓨터공학과 학사
1997~현재 서울대학교 컴퓨터공학부 석사과정
관심분야: 컴퓨터 이론, 생물정보학
E-mail: bhhan@theory.snu.ac.kr

박 근 수



1983 서울대학교 컴퓨터공학과 학사
1985 서울대학교 컴퓨터공학과 석사
1991 미국 Columbia 대학교 전산학 박사
1991.11~1993.8 영국 런던대학교 King's College 조교수
1995.7~1995.8 호주 Curtin 대학교 방문연구원
1993.8~현재 서울대학교 컴퓨터공학과 부교수

관심분야: 컴퓨터 이론, 병렬 계산, 암호학
E-mail: kpark@theory.snu.ac.kr

김 동 규



1992 서울대학교 컴퓨터공학과 학사
1994 서울대학교 컴퓨터공학과 석사
1999 서울대학교 컴퓨터공학과 박사
1999 부산대학교 정보통신연구소 연구진담교수
2000~현재 부산대학교 컴퓨터공학과 기금 조교수
관심분야: 스트림처리 알고리즘, 계산

분자생물학, 인터넷 보안, 압축 알고리즘
E-mail: dkkim1@pusan.ac.kr

• 제27회 정기총회 및 추계학술발표회 •

- 일 자 : 2000년 10월 27 ~ 28일
- 장 소 : 숙명여자대학교
- 발표논문 접수마감 : 2000년 9월 2일(토)
- 문 의 처 : 한국정보과학회 사무국
Tel. 02-588-9246/7, 4001/2
<http://www.kiss.or.kr>