유전자 알고리즘을 이용한 하플로타입 추론
(Haplotype Inference Using a Genetic Algorithm)

이 시 영 † 한 현 구 †† 김 희 철 ††
(See Young Lee) (Hyun Goo Han) (Hee Chul Kim)

요 약 인간과 같은 2배체의 각 염색체는 부모로부터 물려받은 2벌의 복제로 이루어져 있다. 이들 각 복제에서 SNP(single nucleotide polymorphism) 서열 정보를 하플로타입이라 부른다. 인간의 하플로타입 지도를 완전히 찾는 것은 인간 지놈의 중요한 작업 중의 하나인데, 실험적인 방법으로 하플로타입을 직접 얻는 것은 시간이 많이 걸리고 비용이 많이 든다. 따라서 두 하플로타입 정보가 혼합된 지노타입의 샘플들로부터 하플로타입을 추론하는 것에 대하여 연구되어왔다. 이 논문에서는 지노타입들을 설명하는 최소 개수의 하플로타입들을 찾는 모델(최소 하플로타입 추론문제)에 근거하여, 유전자 알고리즘을 사용하여 하플로타입을 추론하는 새로운 접근 방법을 제시한다. 좋은 결과를 주는 것으로 알려진 HAPAR[1]와 이 논문에 제시한 알고리즘을 컴퓨터 실험에 의한 비교를 통하여, 입력이 클 때 이 논문의 알고리즘이 수행시간은 적게 걸리면서 정확성이 비슷함을 보인다. 또한 이 실험 결과를 최근에 제시된 방법인 PTG[2]와 비교한다.

키워드 : 하플로타입 추론, 지노타입, SNP, 유전자 알고리즘

Abstract In diploid organisms like human, each chromosome consists of two copies. A haplotype is a SNP(single nucleotide polymorphism) sequence information from each copy. Finding the complete map of haplotypes in human population is one of the important issues in human genome. To obtain haplotypes via experimental methods is both time-consuming and expensive. Therefore, inference methods have been used to infer haplotyes from the genotype samples. In this paper, we propose a new approach using genetic algorithm to infer haplotypes, which is based on the model of finding the minimum number of haplotypes that explain the genotype samples. We show that by doing a computational experiment, our algorithm has the correctness similar to HAPAR[1] which is known to produce good results while the execution time of our algorithm is less than that of HAPAR as the input size is increased. The experimental result is also compared with the result by the recent method PTG[2].

Key words : Haplotype Inference, Genotype, SNP, Genetic Algorithm

1. 서 론

유전자 변이의 가장 흔한 형태인 SNP(Single Nucleotide Polymorphism)는 개인과 개인 간에 염기 서열에 존재하는 하나의 염기쌍의 차이로 나타나는 변이를 말한다. 이러한 SNP는 유전자 염기 서열에서 대략 1000개 당 한 개 나타난다. SNP는 생물이 진화해가는 동안 생기는 변이나 기타 다른 요인에 의해 발생하고, 이는 어떠한 표현형(phenotype)에 영향을 주게 된다. SNP는 사람의 눈동자 색이 틀리거나 유전질병이 발생하는 것과 같은 생물의 다형성을 가지게 하는 인자이다. 이배체 (diploid) 생물은 각 염색체가 두벌이 있는데, 이 두벌의 각 SNP 서열 정보를 하플로타입이라고 한다. 하플로타입 정보는 개인별, 가계별, 인종별 특성을 결정짓고, 나아가 각종 유전병의 원인 및 치료방법을 밝히는 데 중요한 단서가 된다.

생물학 실험에 의한 일상적인 서열 결정방법에서는 하플로타입 정보보다는 염색체의 두벌의 하플로타입 정보가 혼합된 지노타입 정보를 제공한다. 지노타입은 각 위치에 하플로타입 정보 쌍으로 이루어져 있다. 생물학 실험에 의한 방법으로 염색체의 두 벌의 각 하플로타입

· 이 연구는 2006학년도 한국외국어대학교 교내학술연구비의 지원에 의하여 이루어진 것임
† 비 회 원 : 한국외국어대학교 컴퓨터및정보통신공학부
synuri@paran.com
†† 정 회 원 : 한국외국어대학교 컴퓨터및정보통신공학부 교수
hghan@hufs.ac.kr
hckim@hufs.ac.kr
논문접수 : 2005년 5월 11일
심사완료 : 2006년 3월 28일

정보를 직접 찾는 것은 비용과 시간이 많이 든다. 따라서 다른 대안으로서 컴퓨터를 이용하여 하플로타입을 구하는 방법이 연구되어 왔다. 이 방법은, 주어진 여러 개의 지노타입 정보들의 샘플로부터 각 지노타입을 구성하는 두 벌의 하플로타입을 추론하는 것이다. 이를 하플로타입 추론문제라 한다.

여기서 고려하는 SNP는 biallelic SNP이다. 즉, SNP에 두 가지 종류의 대립유전자(allele)가 나타나는 경우를 고려한다. 지노타입에서의 어떤 위치에 동일한 대립유전자가 나타나면 이를 동질접합체(homozygous)라고 하며, 이는 야행성(wild type)과 변이성(mutant) 두 가지가 있다. 지노타입에서의 어떤 위치에 서로 다른 대립유전자가 나타나면 이를 이질접합체(heterozygous)라고 한다. 예를 들어, SNP site 수가 6이고 지노타입들이 표 1과 같다고 하자.

이 예의 모든 SNP에서 두 가지 대립유전자(염기) 중 하나가 나타난다. SNP 1에서 만약 A가 야행성이라면, T는 변이성이다. 이때, 지노타입 1의 SNP 1의 염기쌍 AA는 동질접합 야행성(homozygous wild type)이고, TT는 동질접합 변이성(homozygous mutant)이다. 그리고 지노타입 4의 SNP 1의 염기쌍 AT는 이질접합체이다. 지노타입 6의 하플로타입 쌍은 <AAATAT, AAATAT>이고, 또한 지노타입 1의 하플로타입 쌍은 <AAACGT, AGACGT>임을 알 수 있다. 지노타입 1과 지노타입 6을 제외한 지노타입들의 하플로타입 쌍을 바로 결정할 수 없다. 예를 들어, 지노타입 5의 하플로타입 쌍은 <TACCAG, TGCTGG>, <TACCGG, TGCTAG>, <TACTAG, TGCCGG>, <TACTGG, TGCCAG> 중 하나이다. 하플로타입 추론문제는 집단의 지노타입들(지노타입 샘플)로부터 각 지노타입을 구성하는 두 벌의 하플로타입을 구하는 것이다.

하플로타입 추론 문제: 길이(SNP site 수)가 m인 지노타입들이 n개 주어져 있을 때, 각 지노타입에 대한 하플로타입 쌍을 찾아라.

하플로타입 추론 문제에 대하여 Clark[3]이 하플로타입 추론문제를 해결하는 알고리즘을 처음으로 제시하였다. Clark는 집단의 지노타입들이 하플로타입을 재구성하는데 유용하다는 사실을 처음으로 발견하고 추론 방법을 제시하였다. 그 이후 하플로타입 문제를 해결하기 위하여 여러 모델들이 제시되고 알고리즘 및 프로그램들이 개발되어왔다[1-5].

본 논문에서 연구하는 모델은 하플로타입 추론문제의 해결방법으로 최소 개수의 하플로타입들을 사용하여 주어진 모든 지노타입들의 하플로타입 쌍을 구하는 것이다. 최소 개수의 하플로타입들을 찾는 문제는 NP-hard [1,4]인데, Gusfield[4]는 정수계획법에 의하여 이 문제에 대한 해를 찾는 방법을 제시하였다. 이 방법은 입력이를 경우 수행시간이 매우 많이 걸린다. 그리고 Wang과 Xu[1]는 분기와 한정(branch and bound) 방법을 이용하여 이 문제의 해를 구하는 알고리즘을 제시하고 이를 프로그램(HAPAR)으로 구현하여 다른 모델과 비교하였다. HAPAR[1]는 지노타입으로부터 하플로타입을 추론하는데 있어서, 좋은 결과를 주는 것으로 알려져 있는 확률과 통계기반의 하플로타입 추론 프로그램 PHASE [5]와 비교하여 유사한 성능을 가지고 있으며, 최소 개수의 하플로타입들에 의한 접근이 실제 하플로타입을 추론하는 문제와 밀접한 관련성을 가진다는 것을 실험을 통하여 보여 주었다. 그러나 이 알고리즘은 모집단의 지노타입들의 수가 많아지거나 지노타입의 길이가 길면 수행시간이 매우 많이 걸리는 단점이 있다. 본 논문에서는 최소 개수의 하플로타입들을 사용하여 모든 지노타입들의 하플로타입 쌍을 구하는 문제와 관련하여, 수행시간이 많이 걸리는 기존의 알고리즘을 개선한 새로운 접근 방법으로 유전자 알고리즘을 이용한 방법을 제안한다. 실험을 통하여, 이 논문에서 제안한 방법을 기존 방법인 HAPAR와 비교하여 우수성을 보인다. 또한, 이 실험결과를 최근에 제시된 방법인 PTG[2]와도 비교한다.

2장에서는 최소 개수의 하플로타입들을 사용하여 추론하는 문제에 대하여 설명하고, 3장에서는 유전자 알고리즘을 이용한 새로운 하플로타입 추론 알고리즘을 제시한다. 4장에서는 실험을 통하여 제시된 알고리즘을 HAPAR과 비교 분석한다. 마지막으로 5장에서 결론을 맺는다.

표 1 지노타입들의 예

	SNP 1	SNP 2	SNP 3	SNP 4	SNP 5	SNP 6
지노타입 1	AA	AG	AA	CC	GG	TT
지노타입 2	AT	AG	AC	CC	GG	GT
지노타입 3	AA	AA	AA	CT	AG	TT
지노타입 4	AT	AG	AC	CT	AG	GT
지노타입 5	TT	AG	CC	CT	AG	GG
지노타입 6	AA	AA	AA	TT	AA	TT
지노타입 7	AA	AA	AA	CT	AG	TT

2. 최소 하플로타입 추론 문제

입력으로 길이가 m인 지노타입들이 n개 있을 때, 각 지노타입은 다음과 같이 0, 1 혹은 2로 이루어진 문자열로 나타낸다: 지노타입의 각 위치에서 동질접합 야행성은 0으로 나타내고, 동질접합 변이성은 1로 나타낸다. 그리고 이질접합체는 2로 나타낸다. 지노타입 g와 하플로타입 h에 대하여, i번째 위치의 요소를 각각 $g[i]$, $h[i]$로 표기한다.

정의. g를 0, 1 혹은 2로 이루어진 길이가 m인 문자열의 지노타입이라 하고 h_1, h_2를 0, 1로 이루어진 길이가 m인 문자열의 하플로타입이라 하자. 모든 $1 \le i \le m$에 대하여, $g[i] \neq 2$이면 $h_1[i] = h_2[i] = g[i]$이고 $g[i] = 2$이면 $h_1[i] \neq h_2[i]$일 때, $\langle h_1, h_2 \rangle$를 g에 대한 하플로타입 쌍이라고 말한다.

하플로타입 추론문제의 해를 구하는 것은 각 지노타입 g에 대한 하플로타입 쌍, 즉 다음을 만족하는 $\langle h_1, h_2 \rangle$를 구하는 것이다: 각 위치에 대하여 g가 0 (혹은 1)을 가지면 h_1, h_2 모두 이 위치에서 0 (혹은 1)을 가지고, g가 2를 가지면 h_1, h_2 중 하나는 0을, 나머지 하나는 1을 가진다.

예를 들어, 지노타입 g가 0212라 하자. g에 대한 하플로타입 쌍으로 <0010, 0111>와 <0011, 0110> 두 가지가 있다. 일반적으로, 하나의 지노타입에서 2의 개수가 k일 때, 이 지노타입에 대한 가능한 하플로타입 쌍의 수는 최대 2^{k-1}개다. 주어진 지노타입들의 집합에 대하여 각 지노타입에 대한 하플로타입 쌍은 매우 많기 때문에 생물학적 정보가 없으면 어느 것이 정확한지를 알 수 없다. 이를 추론하기 위한 모델 중 하나로, 최소 개수의 하플로타입들을 사용하여 주어진 모든 지노타입들의 하플로타입 쌍을 추론하는 모델이 있다. 이를 Pure-Parsimony 접근방법이라고도 하며, 이 모델하에서 하플로타입을 추론하는 문제를 최소 하플로타입 추론문제라 부른다.

최소 하플로타입 추론문제: 길이(SNP site 수)가 m인 지노타입들이 n개 주어져 있을 때, 최소 개수의 하플로타입들을 사용하여 모든 지노타입에 대한 하플로타입 쌍을 구하라.

예를 들어, 길이가 5인 3개의 지노타입을 g_1 = 02120, g_2 = 22110, g_3 = 20120라 하자. g_1, g_2, g_3에 대한 가능한 하플로타입 쌍은 각각 2개 있다. g_1, g_2, g_3의 하플로타입 쌍이 각각 <01100, 00110>, <11110, 00110>, <00100, 10110>일 때 사용하는 전체 하플로타입들의 수는 5이다. 그러나 g_1, g_2, g_3의 하플로타입 쌍이 각각 <00100, 01110>, <01110, 10110>, <00100, 10110>일 때 사용하는 전체 하플로타입들의 수는 3이고, 이것이 최소이다. 최소 하플로타입 추론문제의 추론기준은 집단에서 실제로 관찰된 서로 다른 하플로타입 수가 모든 가능한 하플로타입 수에 비하여 아주 적다는 집단 유전이론의 사실을 반영한 것이다.

3. 유전자 알고리즘을 이용한 하플로타입 추론

이 장에서는 최소 하플로타입 추론문제를 유전자 알고리즘으로 해결하는 새로운 방법을 제시한다. 최소 하플로타입 추론문제는 주어진 n개의 지노타입 집합 $G = \{g_1, g_2, \cdots, g_n\}$에 대하여, 최소 개수의 하플로타입들 집합 $H = \{h_1, h_2, \cdots, h_k\}$를 사용하여 G의 각 지노타입의 하플로타입 쌍을 구하는 것이다.

서론에서 언급하였듯이 Wang과 Xu[1]는 분기와 한정 기법을 이용하여 최소 하플로타입 추론문제의 최적해를 구하여 추론하는 방법을 제시하였는데, 이 방법은 모든 가능한 탐색공간에서 해를 구하기 때문에 최소 개수의 하플로타입들을 구하는데 걸리는 시간이 많이 걸리는 단점이 있다. 수행시간을 개선하기 위하여, 본 논문에서는 유전자 알고리즘을 이용하여 지노타입들의 집합 G를 해결하는데 사용하는 하플로타입의 개수가 가능한 한 적은 하플로타입들의 집합 H(즉, 최소 하플로타입 개수에 근사한 휴리스틱 해)를 구하고, 이로부터 집합 G의 각 지노타입에 대한 하플로타입 쌍을 추론한다.

3.1 유전자 알고리즘

유전자 알고리즘[6]은 두 부모의 유전자로부터 그들 자손의 유전자를 형성하는 유성생식과 자연환경에서 일어나는 진화원리를 흉내낸 것이다. 먼저 초기 집단을 구성한다. 이 집단은 진화원리를 모방한 유전 연산자에 의해 점진적으로 개선되어 진다. 집단에서의 구성원인 각 개체의 우수성은 적합도(fitness) 함수에 의해 계량화된다. 유전자 알고리즘에서는 적합도가 큰 개체들을 선택하고 이들 개체들 간에 유전정보가 교환되어 다음 세대의 집단이 형성된다.

유전자 알고리즘은 최소화 및 최대화문제의 휴리스틱 해를 얻는데 효과적인 방법을 제공한다. 이 논문에서 이용하는 유전자 알고리즘의 기본적인 프로시저는 다음과 같다.

알고리즘 GA

1. 초기 모집단을 생성한다.
2. 모집단의 개체들을 평가한다.
3. while (정지조건을 만족하지 않으면)

{
4. for($i=1$; $i \leq$ 교배를 통해 생성할 개체 수 ; i++)
{
5. 룰렛 휠 선택을 이용하여 적합도가 높은 두 개체를 모집단에서 선택한다.
6. 선택한 두 개체를 교배하여 새로운 개체를 생성한다.
}
7. 교배를 통해 생성되어진 개체들에 대하여 어떤 확률로서 돌연변이 시킨다.
8. 생성되어진 새로운 개체들과 기존의 개체들을 평가하여, 적합도가 높은 개체들을 모집단의 크기만큼 선택하여 모집단을 새로 구성한다.
}

유전자 알고리즘에서 임의의 해들을 가지고 초기 모집단을 구성하고, 교배나 돌연변이와 같은 유전 연산을 수행한다. 유전 연산은 해들을 교배시켜 더 나은 해를 탐색하게 하고, 돌연변이를 통하여 탐색 공간이 지역적인 공간에 머무르지 않도록 해준다. 이를 통해 새로운 해를 얻고, 기존의 해와 새롭게 얻은 해들 중 우수한 해들로 다시 모집단을 구성하게 된다. 따라서, 초기 모집단으로부터 반복적인 수행을 통해 좋은 해로 수렴하도록 유도하는 것이 유전자 알고리즘의 기본적인 목적이다. 이러한 과정은 유전자 알고리즘이 최적 해와 근사한 해들을 얻을 수 있도록 해준다.

3.2 유전자 알고리즘을 이용한 하플로타입 추론 (GA-Haplotyping)

지노타입 g와 하플로타입 h에 대하여, $\langle h, h' \rangle$가 g에 대한 하플로타입 쌍이 되는 h'이 존재하면 h는 지노타입 g와 compatible하다고 말한다. 지노타입 g 또한 하플로타입 h와 compatible하다고 말한다. 그리고 h'를 g에 대한 h의 실현(realization)이라고 말한다. 앞으로 표기 $R_{(g,h)}$는 g에 대한 h의 실현을 나타낸다. 지노타입 g와 하플로타입 h가 주어질 때, g에 대한 h의 실현 $R_{(g,h)}$는 유일하고, 이는 다음과 같다: g의 모든 위치 i에 대하여, $g[i] \neq 2$이면 $R_{(g,h)}[i] = h[i]$이고, $g[i] = 2$이면 $R_{(g,h)}[i] = 1 - h[i]$이다. 이제 3.1절에서 기술한 유전자 알고리즘 GA를 이용하여 길이가 m인 n개의 지노타입들에 대한 하플로타입을 추론하는 방법을 설명한다. 유전자 알고리즘 GA의 각 단계에 대하여 하플로타입 추론과 관련된 부분을 구체적으로 기술한다.

(1) 단계 1의 모집단 초기화

주어진 n개의 지노타입들의 집합을 $G = \{g_1, g_2, \ldots, g_n\}$이라 하자. G의 각 지노타입 $g_i (1 \leq i \leq n)$에 대한 하플로타입 쌍이 $\langle h_{i_1}, h_{i_2} \rangle$일 때, $\bigcup_{i=1}^{n} \{h_{i_1}, h_{i_2}\}$이 하나의 개체가 된다. 다시 말하면, 하나의 개체는 G의 모든 지노타입들에 대한 하플로타입 쌍에 사용되는 $2n$개 이하의 하플로타입들의 어떤 집합이다. 모집단은 개체들의 집합으로서 모집단의 초기화는 다음과 같다. G의 각 지노타입에 대한 하플로타입 쌍을 임의로 생성한 후, 이들 하플로타입들의 집합이 하나의 개체가 된다. 이 개체들을 모집단의 크기만큼 생성한다. 모집단을 P라 하고, 그 크기를 N으로 표기한다(모집단의 크기 N은 유전자알고리즘의 파리미터이다). 다음은 초기 모집단 생성 알고리즘이다.

$P \leftarrow \varnothing$.
for ($i = 1$; $i \leq N$; i++)
{
$p_i \leftarrow \varnothing$ //p_i는 하나의 개체로서 모든 지노타입들을 설명하는 하플로타입들의 집합이다.
모든 지노타입 $g_i \in G$에 대하여
{
// g_i와 compatible한 임의의 하플로타입 h_i을 얻
// 는다.
g_i의 모든 위치 $j (1 \leq j \leq m)$에 대하여,
$g_i[j] = 0$ 혹은 1이면 $h_i[j] = g_i[j]$로 두고, 그렇지 않으면 $h_i[j]$는 무작위로 0과 1 중 하나로 선택한 값을 가지도록 한다.
g_i에 대한 h_i의 실현 h'_i을 구하여 g_i에 대한 하플로타입 쌍을 $\langle h_i, h_i' \rangle$로 둔다.
$p_i \leftarrow p_i \cup \{h_i, h'_i\}$.
}
$P \leftarrow P \cup \{p_i\}$.
}

(2) 단계 2의 개체의 적합도 평가

개체에 있는 하플로타입들의 수가 적을수록 개체의 적합도가 높도록 한다. 이를 위한 방법으로 개체 $p \in P$의 적합도, $f(p)$는 p에 있는 하플로타입의 개수에 반비례하도록 다음과 같이 정의한다.

$$f(p) = \frac{1}{|p|}$$

(3) 단계 3의 정지조건으로 세대 수를 이용하며 이는 유전자 알고리즘의 파라미터이다.

(4) 단계 4의 교배를 통하여 생성할 개체 수가 M일

때, 교배 비율을 M/N로 정의하며 이는 유전자 알고리즘의 파라미터이다.

(5) 단계 5의 선택 연산

단계 5의 개체 선택방법을 기술한다. 개체들의 집합인 모집단을 $P=\{p_1, p_2, \cdots, p_N\}$라 하자. 개체들의 적합도로부터 각 개체 $p_i \in P$가 선택되어질 확률 $\Pr(p_i)$을 다음과 같이 정의한다.

$$\Pr(p_i) = \frac{f(p_i)}{\sum_{k=1}^{N} f(p_k)}$$

개체들이 선택될 확률 $\Pr(P_1), \Pr(P_2), \ldots, \Pr(P_N)$에 대하여 다음의 룰렛 휠 선택을 사용하여 개체 p_i를 선택한다.

```
무작위로 0과 1사이의 실수 r을 생성한다.
sum = 0;
for (i = 1; i ≤ N; i++)
{
   sum = sum + Pr(p_i);
   if (sum >= r) return p_i;
}
```

(6) 단계 6의 교배 연산

교배 연산은 유전자 알고리즘의 성능을 좌우하는 가장 중요한 요인이다. 이 연산은 두 개체 p_1, p_2로부터 보다 우수한(하플로타입 개수가 작은) 새로운 개체 p를 생성하고 이로부터 모든 지노타입들에 대한 하플로타입 쌍을 구하기 위한 것으로 기본적인 개념은 다음과 같다. 먼저 선택연산 (5)에 의하여 모집단 P의 두 개체 p_1, p_2를 선택한다. 모든 지노타입들은 하플로타입 쌍이 정해지지 않은 상태로 초기화되고 p는 $\varnothing$으로 초기화되며 집합 A는 전체 하플로타입들의 집합 $p_1 \cup p_2$로 초기화된다. 다음과 같이 A에서 하플로타입을 선택한 후, 이를 A에서 삭제하여 개체 p에 넣고 이 하플로타입으로부터 해결가능한 지노타입들에 대한 하플로타입 쌍을 구한다: A에서 하플로타입을 선택할 때, 룰렛 휠을 사용하며 compatible한 미해결 지노타입들이 많을수록 하플로타입이 선택될 확률이 크도록 한다. 선택된 하플로타입 h'는 A에서 삭제하고 이를 p에 넣는다. 또한 하플로타입 쌍이 정해지지 않은 미해결 지노타입들 중 h'을 이용하여 해결할 수 있는 각 지노타입 g'에 대한 하플로타입 쌍 $<h', h''>$을 구하고, h''을 p에 넣는다. 모든 지노타입에 대한 하플로타입 쌍을 얻을 때까지 A에서 하플로타입을 선택하여 위의 과정을 계속한다. 다음은 교배 연산에 대한 알고리즘을 단계별로 기술한 것이다.

6.1 룰렛 휠 선택에 의해 P의 두 개체 p_1, p_2를 선택한다.

6.2. $G' \leftarrow G$, $A \leftarrow p_1 \cup p_2$, $p \leftarrow \varnothing$로 둔다.

// G'은 해결되지 않은 지노타입들의 집합으로 G로 초기화된다.

// A는 G'의 지노타입을 해결하기 위한 하플로타입들의 집합으로 $p_1 \cup p_2$로 초기화된다.

// G'와 A는 다음의 루프를 수행할 때마다 크기가 줄어든다.

// p는 교배연산에 의하여 얻어지는 새로운 개체이다.

6.3. A의 모든 원소 h에 대하여,

h와 compatible한 G'의 지노타입들의 수 $n(h)$를 구한다.

6.4. while (G'가 공집합이 아니면)

{

6.4.1. A의 모든 원소 h에 대하여, 룰렛 휠로 h가 선택될 확률 $\Pr(h)$를 다음과 같이 계산한다: $\Pr(h) \leftarrow \frac{n(h)}{\sum_{k \in A} n(k)}$

6.4.2. A의 하플로타입 중에서, 6.4.1에서 계산한 확률에 의한 룰렛 휠 선택으로 하플로타입 h'을 선택한다.

$A \leftarrow A - \{h'\}$, $p \leftarrow p \cup \{h'\}$로 둔다.

6.4.3. h'와 compatible한 G'의 각 지노타입 g'에 대하여

{

g'에 대한 하플로타입 쌍을 $<h', R_{(g',h')}>$로 둔다.

// $R_{(g',h')}$은 g'에 대한 h'의 실현임

$p \leftarrow p \cup \{R_{(g',h')}\}$, $G' \leftarrow G' - \{g'\}$로 둔다.

g'와 compatible한 A의 각 하플로타입 h에 대하여

{

// 남아있는 지노타입 집합 G'에서

// g'가 삭제되므로 h과 compatible한

// 지노타입 수를 1 감소한다.

$n(h) \leftarrow n(h) - 1$로 둔다.

$n(h) = 0$이면 h를 A에서 삭제한다.

}

}

}

교배를 수행하여 좋은 개체들은 다음 세대에 살아남

을 확률이 높아지게 되고, 이들로부터 파생되어지는 자손들은 계속해서 좋은 개체를 생성시키는데 유리하다. 따라서, 반복 수행을 통해 어느 정도까지 좋은 개체들의 집단으로 수렴할 수 있다. 하지만, 어느 정도까지는 좋은 해로 수렴되나 탐색 공간의 지역적인 한계를 벗어날 수 없다. 이러한 한계를 벗어날 수 있도록 교배를 통해 생성되어진 새로운 개체들에 대해 작은 확률로 돌연변이를 수행한다.

(7) 단계 7의 돌연변이

교배 연산에 의하여 생성된 개체의 각 지노타입의 하플로타입 쌍에 대하여 돌연변이가 일어날 확률 μ(유전자 알고리즘의 파라미터임)로 다음을 수행한다.

```
for ( i = 1 ; i ≤ n ; i++) // n은 지노타입의
개수
{
  무작위로 0과 1 사이의 수 r을 생성한다.
  if (r < μ)
    i번째 지노타입에 대한 임의의 하플로타입 쌍을
    얻어 기존의 것을 대체한다.
}
```

(8) 단계 8의 재평가 및 새로운 모집단 결정

교배와 돌연변이를 통해 새롭게 생성되어진 모든 개체들과 현재 세대의 개체들을 적합도에 의하여 내림차순으로 정렬하여, 모집단 크기(N)만큼의 상위 개체들을 선택하여 새로운 모집단 P를 재구성한다. 매 세대마다 위와 같은 작업을 반복적으로 수행하게 되면 최소 하플로타입 개수와 근사한 휴리스틱 해를 얻을 수 있다.

(9) 유전자 알고리즘의 파리미터들

하플로타입 추론을 위한 유전자 알고리즘의 파라미터들인 세대 수, 모집단의 크기, 교배 확률, 돌연변이 확률 등은 4장의 실험 및 분석에 주어져 있다.

유전자 알고리즘에 의한 하플로타입 추론과정을 예를 들어 설명한다. 10개의 지노타입들과 각 지노타입의 원래 하플로타입 쌍이 그림 1과 같다고 하자.

그림 1(c)의 지노타입 정보가 주어질 때, 이로부터 초기모집단의 구성, 개체의 선택, 교배연산을 통하여 지노타입에 대한 하플로타입쌍을 구하는 과정을 기술한다. 먼저 각 지노타입들에 대한 하플로타입 쌍을 임의로 생성한다. 그러면 유전자 알고리즘을 위한 하나의 개체가 만들어지게 되며, 모집단 크기만큼 개체들을 반복 생성하여 초기 모집단을 구성한다. 그림 2는 모집단 크기가 4인 초기 모집단의 예이다.

초기 모집단을 구성한 후, 표 2와 같이 모집단의 개체들에 대한 적합도와 선택확률을 구한다.

표 2에서, 개체 2가 가장 높은 0.071의 적합도를 가지고, 개체 3, 4가 가장 낮은 0.058의 적합도를 가진다.

교배를 위한 개체 선택은 다음과 같이 룰렛 휠 선택 방법을 이용한다. 먼저 각 개체들의 적합도를 이용하여

g_1: (10101010, 10101010)	g_1: (10101010, 10101010)
g_2: (11001010, 00100010)	g_2: (00001010, 11100010)
g_3: (10010010, 11100011)	g_3: (11110010, 10000011)
g_4: (11100010, 10011010)	g_4: (11000010, 10111010)
g_5: (10100011, 10111011)	g_5: (10101011, 10110011)
g_6: (01011010, 11010010)	g_6: (01011010, 11010010)
g_7: (10010100, 01000010)	g_7: (11010100, 00000010)
g_8: (11001010, 00111010)	g_8: (10101010, 01011010)
g_9: (11101011, 00010010)	g_9: (10101010, 01010011)
g_{10}: (10111011, 10101010)	g_{10}: (10101011, 10111010)
(a) 개체 1	(b) 개체 2

g_1: (10101010, 10101010)	g_1: (10101010, 10101010)
g_2: (11101010, 00000010)	g_2: (01001010, 10100010)
g_3: (11010011, 10100010)	g_3: (11110010, 10000011)
g_4: (11111010, 10000010)	g_4: (11001010, 10110010)
g_5: (10110011, 10101011)	g_5: (10101011, 10110011)
g_6: (11011010, 01010010)	g_6: (01011010, 11010010)
g_7: (10010010, 01000100)	g_7: (10010010, 01000100)
g_8: (10101010, 01011010)	g_8: (10001010, 01111010)
g_9: (00101011, 11010010)	g_9: (01110011, 10001010)
g_{10}: (10101011, 10111010)	g_{10}: (10111010, 10101011)
(c) 개체 3	(d) 개체 4

그림 2 크기가 4인 초기 모집단의 예

h_1: 10101010
h_2: 10100011
h_3: 11010010
h_4: 10010100
h_5: 10111011
h_6: 01000010
h_7: 01011010

(a) 사용된 하플로타입들

g_1: (h_1, h_1)
g_2: (h_1, h_6)
g_3: (h_2, h_3)
g_4: (h_1, h_3)
g_5: (h_2, h_5)
g_6: (h_3, h_7)
g_7: (h_4, h_6)
g_8: (h_1, h_7)
g_9: (h_2, h_7)
g_{10}: (h_1, h_5)

(b) 지노타입들과 원래의 하플로타입 쌍

g_1: 10101010
g_2: 22202010
g_3: 12220012
g_4: 12222010
g_5: 10122011
g_6: 21012010
g_7: 22020220
g_8: 22221010
g_9: 22222012
g_{10}: 10121012

(c) 지노타입들의 값

그림 1 지노타입들 예

표 2 적합도 평가

	개체 1	개체 2	개체 3	개체 4
사용된 하플로타입 수	16	14	17	17
적합도	1/16	1/14	1/17	1/17
하플로타입 종류	00010010 00100010 00111010 01000010 01011010 10010010 10010100 10011010 10100011 10101010 10111011 11001010 11010010 11100010 11100011 11101011	00000010 00001010 01010011 01011010 10000011 10101010 10101011 10110011 10111010 11000010 11010010 11010100 11100010 11110010	00000010 00101011 01000100 01010010 01011010 10000010 10010010 10100010 10101010 10101011 10110011 10111010 11010010 11010011 11011010 11101010 11111010	01000100 01001010 01011010 01110011 01111010 10000011 10001010 10010010 10100010 10101010 10110011 10110010 10110011 10111010 11001010 11010010 11110010
선택 확률	0.252 (0.063/0.25)	0.284 (0.071/0.25)	0.232 (0.058/0.25)	0.232 (0.058/0.25)
누적 확률	0~0.252	0.252~0.536	0.536~0.768	0.768~1

g_1:	10101010
g_2:	22202010
g_3:	12220012
g_4:	12222010
g_5:	10122011
g_6:	21012010
g_7:	22020220
g_8:	22221010
g_9:	22222012
g_{10}:	10121012

하플로타입들	하플로타입과 compatible한 지노타입들	n(h)
00000010	g_2, g_7, g_9	3
00001010	g_2, g_8, g_9	3
00010010	g_7, g_9	2
00100010	g_2, g_9	2
00111010	g_8, g_9	2
01000010	g_2, g_7, g_9	3
01010011	g_9	1
01011010	g_6, g_8, g_9	3
10000011	g_3, g_9	2
10010010	g_3, g_4, g_7, g_9	4
10010100	g_7	1
10011010	g_4, g_8, g_9	3
10100011	g_3, g_5, g_9	3
10101010	g_1, g_2, g_4, g_8, g_9, g_{10}	6
10101011	g_5, g_9, g_{10}	3
10110011	g_3, g_5, g_9	3
10111010	g_4, g_8, g_9, g_{10}	4
10111011	g_5, g_9, g_{10}	3
11000010	g_2, g_3, g_4, g_7, g_9	5
11001010	g_2, g_4, g_8, g_9	4
11010010	g_3, g_4, g_6, g_7, g_9	5
11010100	g_7	1
11100010	g_2, g_3, g_4, g_9	4
11100011	g_3, g_9	2
11101011	g_9	1
11110010	g_3, g_4, g_9	3

(a) 지노타입들 (b) 하플로타입들 및 이와 compatible한 지노타입들

그림 3 교배의 초기단계
(지노타입들 및 초기에 선택된 두 개체에 사용된 전체 하플로타입들)

개체에 대한 선택 확률을 구하고 이를 이용하여 누적

g_3:	12220012
g_5:	10122011
g_6:	21012010
g_7:	22020220

하플로타입들	하플로타입과 compatible한 지노타입들	n(h)
00000010	g_7	1
00010010	g_7	1
01000010	g_7	1
01011010	g_6	1
10000011	g_3	1
10010010	g_3, g_7	2
10010100	g_7	1
10100011	g_3, g_5	2
10101011	g_5	1
10110011	g_3, g_5	2
10111011	g_5	1
11000010	g_3, g_7	2
11010010	g_3, g_6, g_7	3
11010100	g_7	1
11100010	g_3	1
11100011	g_3	1
11110010	g_3	1

(a) 해결되지 않은 지노타입들 (b) 남아 있는 하플로타입들 및 이와 compatible한 지노타입들

그림 4 교배의 중간단계-1

확률을 구성한다(표 2 참조). 무작위로 0과 1사이의 수를 만든 후 누적확률이 이 수를 포함하는 개체를 선택한다. 이러한 룰렛 휠 선택 방법으로 두 개의 개체를 선택할 때, 개체 1과 개체 2가 선택되어졌다고 하자. 개체 1과 개체 2에서 나타난 하플로타입들을 병합한 후, 각 하플로타입과 compatible한 지노타입의 개수에 대한 테이블을 구성한다(그림 3 참조).

룰렛 휠 선택으로 가장 빈도수가 높은 하플로타입 10101010이 선택되어졌다고 하자. 이로부터 지노타입들 g_1, g_2, g_4, g_8, g_9, g_{10}은 각각 g_1(10101010+10101010), g_2(10101010+01000010), g_4(10101010+11010010), g_8(10101010+01011010), g_9(10101010+01010011), g_{10}(10101010+ 10111011)으로 결정되어진다. 하플로타입 쌍이 정해진 지노타입들의 집합 = {g_1, g_2, g_4, g_8, g_9, g_{10}}이 되고, 해결되지 않은 지노타입들의 집합 G′={g_3, g_5, g_6, g_7}이 된다(그림 4 참조).

남아 있는 지노타입에 대한 하플로타입 쌍을 구하기 위해서 룰렛 휠 선택으로 11010010이 선택되어졌다고 하자. 지노타입들 g_3, g_6, g_7은 각각 g_3(11010010+10100011), g_6(11010010+01011010), g_7(11010010+00000100)으로 결정되어 하플로타입 쌍이 정해진 지노타입들의 집합 = {g_1, g_2, g_3, g_4, g_6, g_7, g_8, g_9, g_{10}}이고, 해결되지 않은 지노타입들의 집합 G′={g_5}가 된다(그림 5 참조).

마지막으로 10100011이 선택되어졌다고 하면, g_5(10100011+10111011)이 되어 모든 지노타입에 대한 하플로타입 쌍이 구해지게 된다. G′ = ∅가 되어 교배 연산은 끝나고, 이를 통하여 지노타입들에 대한 하플로타입 쌍은 다음과 같이 정해진다:

g_5: 10122011

하플로타입들	하플로타입과 compatible한 지노타입들	n(h)
10100011	g_5	1
10101011	g_5	1
10110011	g_5	1
10111011	g_5	1

(a) 해결되지 않은 지노타입 (b) 남아 있는 하플로타입들 및 이와 compatible한 지노타입

그림 5 교배의 중간단계-2

g_1(10101010+10101010), g_2(10101010+01000010), g_3(11010010+10100011), g_4(10101010+11010010), g_5(10100011+10111011), g_6(11010010+01011010), g_7(11010010+00000100), g_8(10101010+01011010), g_9(10101010+01010011), g_{10}(10101010+10111011).

위에서 사용된 하플로타입들의 집합은 {00000100, 01000010, 01010011, 01011010, 10100011, 10101010, 10111011, 11010010}으로 이것이 교배를 통하여 생성된 개체가 된다. 하플로타입 수가 각각 14인 개체와 16인 개체를 교배하여 하플로타입 수가 8인 개체가 생성되었다. 이 결과를 지노타입들의 실제 하플로타입 쌍(그림 1 (b))과 비교해보면 g_7, g_9를 제외하고는 모두 정확하게 추론되어진 것을 알 수 있다.

4. 실험 및 분석

알고리즘의 성능을 분석하기 위하여 오차율을 사용하였다. 오차율은 n개의 입력 지노타입에 대하여 알고리즘으로 하플로타입 쌍을 결정하였을 때, 실제와 일치하는 않는 지노타입의 개수이다. 즉, n개의 지노타입들에 대하여 하플로타입 쌍을 정확하게 추론하지 못한 지노타입의 개수가 k라면 오차율은 k/n이다.

이 논문에서 제안한 유전자 알고리즘에 기반한 하플로타입 추론방법(GA-Haplotyping)의 성능을 분석하기 위해, 분기와 한정 기법으로 Wang과 Xu[1]가 제안한 HAPAR과 비교 분석하였다. 본 논문에서 제안한 GA-Haplotyping은 동일한 환경에서 성능을 비교하기 위해 모집단 크기 100, 세대 수 20, 교배 비율 0.5, 돌연변이 비율 0.01의 일률적인 수행환경을 유지하였다. 각 테스트에 대한 오차율 계산은 10회 수행한 후 평균한 값으로 하였다. 사용된 시스템은 CPU : Intel Pentium4 2.0, Memory : 512M이다.

4.1 임의의 데이타 집합에 대한 시뮬레이션

먼저 SNP site 수가 10인 하플로타입들을 무작위로 10개 생성하였다. 이들 10개의 하플로타입들의 집합 $H=\{h_1,h_2,...,h_{10}\}$로부터 n=4, 10, 20, 40에 대하여 다음과 같이 n개의 지노타입들의 집합 $\{g_1,g_2,\cdots,g_n\}$을 만들어 실험하였다. 각 $g_i(1\le i\le n)$를 구성하는 하플로타입 쌍 h_{i_1},h_{i_2}을 H로부터 무작위로 선택한 후, $1\le k\le 10$에 대하여 $h_{i_1}[k]=h_{i_2}[k]$이면 $g_i[k]=h_{i_1}[k]$로 하고, 그렇지 않으면 $g_i[k]=2$로 한다. 표 3은 이 데이타 집합에 대하여 성능을 비교한 결과이다. 지노타입 수가 매우 작을 경우는 두 프로그램이 모두 정확한 추론이 불가능했고, 10개 이상일 경우 GA-Haplotyping과 HAPAR 모두 오차율 0.05 이하로 상당히 좋은 결과를 얻었다.

표 3 임의 생성 데이타에 대한 오차율 비교
(SNP site 수: 10, 하플로타입 수: 10)

지노타입 수	GA-Haplotyping	HAPAR
4	0.625	0.725
10	0.04	0.03
20	0.005	0
40	0	0

4.2 MX1 데이타 집합에 대한 시뮬레이션

두 번째 실험에서는 MX1[7]에 나타난 SNP site 수가 12인 10개의 하플로타입들로부터 실제 출현되어지는 빈도수를 이용하였다. 임의의 지노타입 입력 데이타를 구성하는 하플로타입들의 비율은 실제 MX1에서 출현되어진 확률 값이 되도록 입력데이타를 생성하였다. 이 경우 지노타입 수가 8, 12일 때는 HAPAR가 좀 더 정확한 결과를 나타냈고, 지노타입 수가 12, 16, 20일 때는 GA-Haplotyping이 좀 더 나은 결과를 보였으며, 30 이상부터는 둘 다 거의 비슷하게 오차율 0.02 미만의 매우 높은 정확성을 보였다(표 4 참조). GA-Haplotyping이 좀 더 나은 결과를 보인 것은 최소 하플로타입 추론문제의 최적 해가 항상 원래 해가 되는 것이 아니기 때문이다.

표 4 MX1 데이타에 대한 오차율 비교
(SNP site 수: 12, 하플로타입 수: 10)

지노타입 수	GA-Haplotyping	HAPAR
4	0.225	0.222
8	0.1	0.09
12	0.075	0.064
16	0.019	0.028
20	0.025	0.05
30	0.016	0.012
40	0.012	0.015

4.3 MS프로그램으로 생성한 데이타 집합에 대한 시뮬레이션

세 번째 실험에서는 Hudson에 의해 제안되어진 MS 프로그램[8]을 이용하여 SNP site 수가 30개, 40개인

하플로타입 데이타를 얻은 뒤 지노타입 입력 데이타를 구했다(재조합(recombination) 파라미터는 0으로 조정하였다 - 재조합이 발생하지 않음). MS 프로그램은 조상이 되는 하나의 하플로타입을 임의로 생성한 뒤, 변이율과 재조합 등 다양한 유전자 파라미터를 이용하여 하플로타입들을 생성한다. 이것은 실세계를 모방하여 생산되어진 하플로타입들이기 때문에 실제 데이타와 매우 유사한 신빙성 있는 지노타입 입력 데이타를 제공한다. 이 실험에서 HAPAR는 수행시간이 너무 오래 걸려 결과를 도출하지 못했지만, GA-Haplotyping은 평균 오차율 0.1 정도의 정확성을 보였다(표 5 참조).

표 5 MS를 이용한 SNP site 수가 많은 데이타에 대한 GA-Haplotyping의 오차율

지노타입 수	SNP site 수: 30	SNP site 수: 40
30	0.16	0.26
40	0.12	0.155
60	0.136	0.08
80	0.057	0.139
100	0.042	0.056

(* HAPAR는 수행시간이 많이 걸려 결과를 얻지 못함)

4.4 Maize 데이타 집합에 대한 시뮬레이션

네 번째 실험에서는 정확도 계산을 위한 벤치마크에서 가장 널리 사용되어지는 것 중 하나인 Maize 데이타 집합[9]에 대한 실험 결과이다. 이 데이타는 SNP site 수가 17이고, 4개의 하플로타입들로 이루어져 있다. 임의의 지노타입 입력 데이타를 구성하는 하플로타입들의 비율은 실제 Maize에서 출현되어진 확률 값이 되도록 입력데이타를 생성하였다. 이 결과 GA-Haplotyping이 HAPAR보다 좀 더 나은 정확도를 보여주고 있으며, 지노타입 샘플개수가 4개 이상에서는 오차율 0.1 이하의 정확도를 보였다(표 6 참조).

그리고 최근에 발표된 PTG[2] 방법에 의해 추론되어진 결과와 비교하였는데, PTG 방법에 의한 결과가 조금 나았다.

표 6 Maize 데이타에 대한 오차율 비교
(SNP site 수: 17, 하플로타입 수: 4)

지노타입 수	Ga-Haplotyping	HAPAR	PTG
3	0.367	0.51	0.02
4	0.1	0.1	0
7	0.042	0.05	0
10	0	0	0

4.5 ACE 데이타 집합에 대한 실험

다섯 번째 실험에서는 Maize와 함께 가장 널리 사용되고 있는 ACE 데이타 집합[10]에 대한 실험 결과이다. ACE 데이타 집합은 52개의 SNP site와 11개의 지노타입으로 이루어져 있다. ACE 데이타 집합에 대하여 GA-Haplotyping이 HAPAR보다 나은 결과를 보였다. 그러나, PTG[2]의 경우 ACE 데이타 집합에서 가장 좋은 결과인 오차율 0.182을 보였다.

표 7 ACE 데이타에 대한 오차율 비교
(SNP site 수: 52, 하플로타입 수: 13, 지노타입 수: 11)

	GA-Haplotyping	HAPAR	PTG
오차율	0.236	0.273	0.182

4.6 β_2AR 데이타 집합에 대한 실험

여섯 번째 실험에서는 Wang과 Xu[1]에서 언급된 β_2AR[11] 데이타 집합에 대한 실험 결과이다. 이 데이타 집합에서는 하플로타입 쌍이 알려져 있는 18개의 지노타입들로 구성되어 있다. 이 데이타에 대하여 HAPAR의 추론결과는 제시되어 있지 않은데, GA-Haplotyping에 의한 10번의 실험에서는 18개의 지노타입을 모두 정확하게 추론하였다. PTG[13]에서는 이 데이타에 대하여 100번의 실험을 한 결과의 오차율이 0.056이었다. β_2AR 데이타 집합에 대하여 GA-Haplotyping이 PTG[2]보다 약간 좋은 결과를 얻었다.

4.7 HAPAR와 GA_Haplotyping의 수행 시간 비교

일반적으로 수행속도는 SNP site 수와 매우 밀접한 연관을 보였다. SNP site 수가 15 정도 일 때 HAPAR는 1-3초 정도 걸린 반면, GA_Haplotyping은 5초-10초 정도 걸렸다. 이는 GA_Haplotyping은 모집단의 크기나 교배 개체들의 수에 따라 수행 시간의 영향을 상대적으로 많이 받기 때문에 SNP site 수가 적더라도 일정한 시간이 걸리기 때문이다. SNP site 수가 20일 경우 HAPAR는 20-40초 정도, GA_Haplotyping은 30초-1분 정도 걸렸다. 하지만, SNP site 수가 20을 넘어서면서 HAPAR는 수행시간이 급격하게 증가해서 실험 4.3에서와 같이 SNP site 수가 30일 경우와 40인 경우 HAPAR는 시간이 너무 많이 걸려 결과를 얻을 수 없었지만, GA_Haplotyping은 5분 이내에 결과를 도출하였다.

4.8 모집단 크기와 정확성 상관관계 테스트

MX1[7] 데이타를 이용하여 모집단의 크기에 따른 오차율의 상관관계를 테스트하였다. 실험에서는 8개의 지노타입과 20개의 지노타입을 이용하였다. 이 실험에서 지노타입이 20개인 경우는 모집단의 크기와 크게 상관없이 비슷한 정확도를 보였다. 지노타입이 8개인 경우, 모집단 크기가 5, 40, 100, 200인 경우는 비슷한 정확도를 보였는데, 모집단 크기가 10인 특정한 크기에서 정확도가 약간 높았다.

표 8 모집단 크기 별 오차율과 상관관계

지노타입 수 \ 모집단 수	5	10	40	100	200
8	0.11	0.04	0.11	0.1	0.11
20	0.04	0.035	0.03	0.025	0.2

위의 모집단 크기의 정확도 관계 실험을 바탕으로, 이 논문에서 실험한 모든 데이타에 대한 모집단을 크기를 100으로 정하였다.

5. 결 론

본 논문에서는 지노타입 샘플로부터 하플로타입을 추론하기 위해 유전자 알고리즘을 이용한 새로운 알고리즘(GA-Haplotyping)을 제안하고, 실험을 통하여 최소 하플로타입 추론문제에 기반한 기존의 하플로타입 추론 프로그램인 HAPAR와 비교 분석하였다. HAPAR은 다른 하플로타입 추론 프로그램과 비교하여 정확도가 매우 높은 것으로 알려진 프로그램이다. 본 논문에서 제안한 방법을 이용하여 구현한 GA-Haplotyping은 SNP site 수가 20 미만인 대부분의 데이타 집합에 대하여, HAPAR과 GA_Haplotyping 모두 짧은 시간 내에서 정확성이 높은 결과를 도출하였다. SNP 길이가 20 이상인 경우 HAPAR는 수행 시간이 너무 많이 걸려 결과를 얻지 못한 반면, GA-Haplotyping은 SNP 수가 30~40일 때, 최대 5분 이내에 결과를 얻었으며, 지노타입 개수가 너무 작아 분석하기 매우 힘든 경우를 제외하고는 평균 0.1 이하의 오차율을 보였다. HAPAR는 분기와 한정 기법을 통해 최소 하플로타입 추론문제의 최적의 해를 구하기 때문에 SNP site 수가 클수록 그 결과를 얻는데 걸리는 시간이 급속하게 증가한다. 그러나 휴리스틱 해를 이용하는 GA-Haplotyping은 SNP 길이가 클 경우에도 해를 찾을 수 있으며, HAPAR과 비교하여 유사한 정확성을 얻었다.

최근의 결과인 PTG[2]와도 비교를 하였는데, PTG가 좋은 경우도 있었고, GA_Haplotyping이 좋은 경우도 있었다. PTG와는 체계적인 비교연구가 앞으로 계속 필요하다.

참 고 문 헌

[1] L. Wang and Y. Xu, "Haplotype inference by maximum parsimony," *Bioinfomatics* Vol. **19**(14), pp. 1773-1780, 2003.

[2] Zhenping Li, Wenfeng Zhou, Xiang-Sun Zhang, and Luonan Chen, "A parsimonious tree-grow method for haplotype inference," *Bioinformatics*, Vol. **21**(17), pp. 3475-3481, 2005.

[3] A. G. Clark, "Inference of haplotypes from PCR-amplified samples of diploid populations," *Mol. Biol. Evol.* **7**, pp. 111-122, 1990.

[4] D. Gusfield, "Haplotype inference by pure parsimony," *Lecture Notes in Computer Science* **2676**, Springer, pp. 144-155, 2003.

[5] M. Stephens, N. J. Smith, and P. Donnelly, "A new statistical method for haplotype reconstruction for population data," *Am. J. Hum. Genet.* **68**, pp. 978-989, 2001.

[6] D.E. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley Publishing Company, Inc. 1989.

[7] L. Jin et al., "Distribution of haplotypes from a chromosome 21 region distinguished multiple prehistoric human migrations," in *Proc. of Natl Acad. Sci. USA* **96**, pp. 3796-3800, 1999.

[8] R. Hudson, "Generating samples under a Wright-Fisher neutral model of genetic variation," *Bioinformatics* **18**, pp. 337-338, 2002.

[9] A. Ching et al., "SNP frequency, haplotype structure and linkage disequilibrium in elite maize inbred lines," *BMC Genet.* **3**, pp. 19, 2002.

[10] M. Rieder et al., "Sequence variation in the human angiotensin converting enzyme," *Nat. Gene.* **22**, pp. 59-62, 1999.

[11] C. Drysdale *et al.*, "Complex promoter and coding region β_2-adrenergic receptor haplotypes alter receptor expression and predict in vivo responsiveness," in *Proc. Natl Acad. Sci. USA* **97**, pp. 10483-10488, 2000.

이 시 영

2002년 한국외국어대학교 컴퓨터공학과 학사. 2004년 한국외국어대학교 컴퓨터공학과 석사. 2004년~현재 ㈜토필드 부설 연구소 연구원. 관심분야는 생물정보학, 임베디드 시스템, 유비쿼터스

한 현 구

1980년 서강대학교 수학과 학사. 1985년 University of South Florida 전산학석사 1990년 Auburn University 전산학과 박사. 1992년~현재 한국외국어대학교 컴퓨터및정보통신공학부 교수. 관심분야는 로봇 planning, Software Agent Systems

김 희 철

1980년 서울대학교 계산통계학과 학사 1982년 한국과학기술원 전산학과 석사 1987년 한국과학기술원 전산학과 박사 1987년~현재 한국외국어대학교 컴퓨터및정보통신공학부 교수. 1997년 8월~1998년 8월 미국 Michigan State University 방문교수. 관심분야는 그래프 이론, 상호연결망, 생물정보학