DNA 컴퓨팅 기술의 개요 및 응용

한남대학교 이상구*
한국과학기술원 이광형†
대전산업대학교 임기영

DNA 컴퓨팅(DNA computing)은 생체 분자들이 갖는 막대한 정보처리 능력을 디지털 컴퓨터의 기본 스위칭 소자로 대체할 수 있다는 발상에서 출발한다. 또한 현재의 반도체 기술도 머지않아 한계에 도달하고, 컴퓨터의 기본적인 소자를 더욱 더 마이크로화해야 된다는 추세 속에 양자계산(Quantum computing)과 DNA 컴퓨팅 분야가 최근 들어 지대한 관심을 끌고 있다. 본고에서는 DNA 컴퓨팅 기술의 개요와 응용분야에 대해서 살펴보도록 한다.

1. 서 론

생명체의 세포의 핵 내부에는 중요한 정보를 갖는 도서관이 존재하며, 생명현상에 요구되는 단백질의 생성 등에 필요한 방법론적인 정보가 들어 있어 필요시에 정보를 복사할 수 있고, 그 복사본을 핵의 외부로 반출할 수는 있지만 원본은 항상 핵 내부에 남아있어야 한다. 생물체의 발생, 성장 등에 필요한 모든 정보들이 세포내의 핵과 같이 작은 장소에 보관된다는 것이 신기할 따름이지만, 사실 이러한 정보를 기록하는데 사용되는 언어가 바로 DNA(Deoxyribo nucleic acid)이다. DNA는 유전 물질로서 다음과 같은 특성을 갖고 있다.

(1) 자기복제(replication)를 할 수 있고,
(2) 돌연변이(mutation)를 유발하여 다음 세대에 전달할 수 있으며
(3) 세포와 생명체의 특성을 결정짓는 정보를 저장하며
(4) 이러한 정보를 이용하여 세포나 생명체의 활동에 필수적인 여러 가지 단백질의 합성을 지시할 수 있다.

핵산(nucleic acid)은 아단위체(subunit)로 구성된다. 핵산의 아단위체는 뉴클레오티드(nucleotide)이다 각각의 뉴클레오티드는 한 분자씩의 5탄당(S), 인산기(phosphate group, P), 질소염기(nitrogenous base, B)로 구성된다. 뉴클레오티드를 이루는 염기는 크기에 따라 두 가지로 구별된다. 크기가 큰 종류(purine 계열)에는 아데닌(adenine, A)과 구아닌(guanine, G)이 해당되며, Primidine 계열에는 시토신(cytosine, C)과 티민(thymine, T)이 있다. DNA는 폴리펩티드 형성시에 원본 역할을 하는 핵산으로서, 디옥시리보스형태의 당, 인산 그리고 질소염기인 아데닌, 티민, 구아신 및 시토신을 함유하고 있다. 이러한 DNA는 두 개의 사슬로 구성되어 있다. 각각의 사슬은 전체적으로 머리빗 모양으로 구성되어 있고, 빗살에 해당되는 염기들은 다른 사슬의 염기들과 수소결합(hydrogen bond)에 의해 연결된다. 이러한 형태로 결합된 두 개의 사슬은 서로 뒤틀려서 이중나선구조를 이루고 있는데, 이것을 이중 DNA(duplex DNA)라 한다. 앞에서 설명한 4가지의 염기들은 언제나 일정한 방식으로 결합하여 당을 이루는데, 결합은 A와 T, C와 G만이 결합할 수 있다. 이와 같이 서로 쌍을 이루고 결합하는 염기들을 상보염기(complementary base)라고 한다. DNA를 구성하는 4개의 뉴클레오티드(A, T, G, C)를 특정의 서열

†종신회원

로 조합하면 DNA의 유전정보를 기록, 해독할 수 있다. 이러한 4개의 뉴클레오티드는 일종의 알파벳으로 사용되어 단위개의 철자로 구성되는 단어를 형성할 수 있다.

DNA 컴퓨팅 기술이 앞으로 크게 기대 할 수 있는 배경으로 다음과 같은 2가지의 기본적인 특징을 꼽을 수 있다.

(i) DNA 분자가 갖는 초병렬성(massive parallelism)

(ii) Watson, Crick의 상보성(complementarity)

DNA 분자를 이용하여 계산량적으로 어렵다고 하는 문제를 푸는 것 외에, DNA 컴퓨팅을 연구하는 이유는 작지 않다. 예를 들면 자연은 어떻게 "계산"을 수행하는 것인가에 대해 이해하는 것은 중요하다. 즉, 이 부분을 깊게 연구하면 DNA를 정교하게 조작함으로써 생명의 놀라울 정도의 정교함과 효율성에의 지식을 얻을 수 있다. 또한 현재의 컴퓨터 과학에 대한 기존의 개념과는 다른 새로운 계산 패러다임에 도달할 수 있다. 여기서는 새로운 데이터구조, 새로운 데이터 구조상에서의 새로운 형태의 연산, 새로운 계산가능성의 모델 등이 제안된다.

고전적인 이론계산 과학은 오토마타, 언어이론에 대한 모델에 기초하고 있지만, 자연계에서는 cut, paste, adjoining, insert, delete 등의 다른 형태의 연산을 이용하여 DNA 분자의 계산을 실행하고 있다.

2. 본 론

Feynman은 "Miniaturization"에서 초미세한 컴퓨터를 만들 수 있는 가능성을 제시하였다. 이후 컴퓨터의 소형화에 대한 경이적인 발전이 이루어 졌지만, 초미세한 컴퓨터를 만들려고 하는 목표는 아직 달성되어 있지 않다. 이에 DNA 컴퓨팅에 대한 연구가 최근에 각광을 받고 있지만 DNA 컴퓨팅의 궁극적인 영향이 어디까지 미칠까에 대한 예견은 현재 로서는 어렵다고 하겠다.

DNA 컴퓨팅에 관한 아주 좋은 예제가 여기에 소개하는 Adleman의 실험이다.

Adleman의 실험은 그림 1에 나타난 것처럼 유향 그래프에서의 해밀토니언 경로문제(Hamiltonian pathproblem: HPP)를 DNA 컴퓨팅 방법으로 푼 것이다. 입력정점 V_{in}에서 출력정점 V_{out}에 이르는 경로가 해밀토니언하다는 것은 모든 정점을 반드시 한번만 포함하는 것이다. 그림 1의 유향그래프에서 0 -> 1 -> 2 -> 3 -> 4 -> 5 -> 6 의 경로는 헤밀토니언 경로이다. 실제로 HPP는 NP-complete 문제이고, 다항식 시간에 계산할 수 있는 효율이 좋은 알고리즘은 존재하지 않는다고 알려져 있다. 그러나 DNA 컴퓨팅의 초병렬성과 상보성의 특징을 이용하면 매우 큰 size의 HPP에도 적용가능하다.

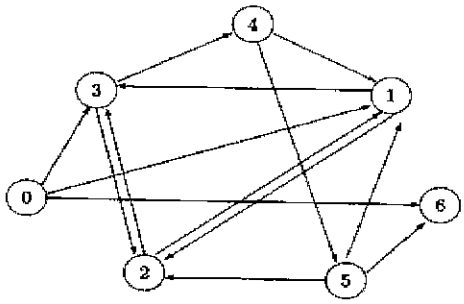


그림 1 해밀토니언 경로 문제

Adleman의 해법은 다음과 같은 HPP를 풀기 위한 non-deterministic한 알고리즘에 기반을 두고 있다.

Adleman의 Non-deterministic algorithm

Input:vertex의 수 n의 directed graph G로 지정된 V_{in} 과 V_{out}을 갖는다.

Step 1:대량의 자원을 이용하여 G에 대한 path를 random으로 생성한다.

Step 2:V_{in}으로 시작하지 않거나, V_{out}으로 끝나지 않는 path를 삭제한다.

Step 3:n개의 vertex를 포함하지 않는 것을 삭제한다.

Step 4:n개의 vertex v에 대해 v를 포함하지 않는 path를 삭제한다.

Output:path가 남아 있으면 "Yes", 아니면 "No"

Adleman의 실험에서의 자료구조는 다음과 같다. 각 정점 $S_i(0 \leq i \leq 6)$에 20-mer의 DNA 분자를 배열한다.

S_2 = TATCGGATCGGTATATCCGA

S_3 = GCTATTCGAGCTTAAAGCTA

s_1 = GGCTAGGTACCAGCATGCTT

여기서 상보성 매핑함수 h(Watson-Crick morphism)를 정의하면

h(A)=T, h(T)=A, h(C)=G, h(G)=C이다.
따라서

h(s_2)=ATAGCCTAGCCATATAGGCT
h(s_3)=CGATAAGCTCGAATTTCGAT

여기서 $S_i = S_i' S_i''$로 정의하면

정점 i와 정점 j사이의 edge는 $h(S_i'' S_j')$로 부호화 할 수 있다.

즉 $e_{2\to3}$ = CATATAGGCTCGATAAGCTC

$e_{3\to2}$ = GAATTTCGATATAGCCTAGC

Adleman의 실험에서는 각 정점 i와 각 변 i→j에 대해, 대량의 뉴클레오티드 S_i 와 $e_{i\to j}$가 한 개의 시험관내에서 ligase와 같이 혼합된다. 원하는 부분의 DNA를 복사하는 것은 중합효소연쇄반응(Polymerase Chain Reaction: PCR)에 의한다. 증폭하고자하는 표적 DNA가 결정되면, 우선 시료 DNA를 가열하여 두 가닥 DNA를 한 가닥씩으로 분리시킨다. 합성된 한 가닥 사슬 DNA들을 시료에 첨가하면 표적 DNA의 양쪽 끝 부분에 결합해 염기쌍을 이룬다.

DNA 중합효소를 첨가하면, 이 효소는 단일가닥 사슬 DNA로부터 시작하여 시료 DNA를 따라가면서 뉴클레오티드를 첨가하여 상보적인 DNA 사슬을 형성하는데, 이렇게 하여 표적 DNA의 새로운 사본이 완성된다. 이 반응은 반복적으로 일어나기 때문에 단 시간에 표적 DNA의 사본을 급격히 증가시킬 수 있다.

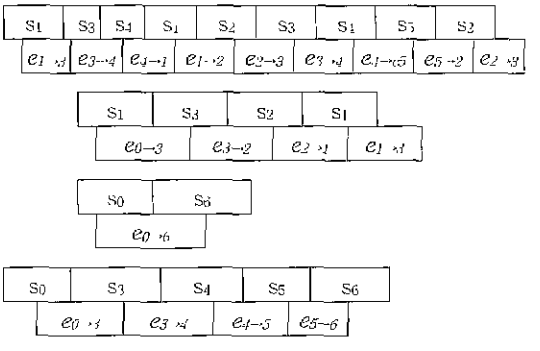


그림 2 Adleman의 그래프에 대한 경로의 예

Adleman의 실험에 있어서 중요한 점을 추상적으로 정형화하면 시험관은 알파벳 {A,C,G,T} 상의 유한길이의 기호열로 된 다중집합이라 할 수 있다. 다음의 기본 연산은 DNA 컴퓨팅에서의 주요한 연산들이다.

• merge:주어진 시험관 N_1과 N_2에 대한 $N_1 \cup N_2$

• amplify:주어진 시험관 N 에 대해 2개의 copy를 만듦

• detect:N이 적어도 1개의 DNA 분자를 포함하면 "true", 아니면 "false"

• separate 또는 extract:N과 {A,C,G,T}상의 기호열 w에 대해서 2개의 시험관+(N,W), -(N,W)를 만듦, 여기서 +(N,W)는 w를 부분기호열로서 포함하는 N의 모든 기호열(분자)이고, -(N,W)는 w를 부분기호열로서 포함하지 않는 N의 모든 기호열이다.

• length-separate : 주어진 시험관 N과 기호열 w에 대해서 N에대한 길이 n이하의 모든 배열문자를 포함하는 시험관(N,≤n)을 생성

• position-separate : N에 관한 모든 배열분자에서 w로 시작하는 것을 모두 포함하는 시험관 B(N,w)를 생성하고, w로 끝나는 것을 모두 포함하는 시험관 E(N,w)를 생성

Adleman의 실험에 있어서 필터링 또는 스크리닝 조작은 다음과 같은 프로그램에 의해 기술할 수 있다. 각 뉴클레오티드 $S_i(0\le i\le 6)$는 길이 20으로 한다.

(1) *input*(N)
(2) $N \leftarrow B(N, s_0)$
(3) $N \leftarrow E(N, s_6)$
(4) $N \leftarrow (N, \le 140)$
(5) for i = 1 to 5 do begin $N \leftarrow +(N, s_i)$ end
(6) *detect*(N)

이러한 연산들은 DNA 계산에 있어 대부분의 알고리즘에 대해서 기본적이며, 또한 신뢰성, 효율, 오차율에 관한 연구도 필수적이다. 이 외에도 DNA 연산을 실행하기 위하여 다음과 같은 연산이 사용된다.

• melting:어떤 특정한 온도에서 용액을 가열하여 이중가닥(double strand) DNA가 단일가닥(single strand)으로 분리된다. 이렇게 하여, 상보적 분자간의 수소 결합이 파괴된다.

• annealing:이것은 melting의 역 연산이다. 단일가닥 분자의 용액을 서서히 식히면 Watson, Crick 상보적인 분자들이 상호간에 결합한다.

또한, Boheh 등은 DES(data encryption standard) 암호 알고리즘을 DNA 컴퓨팅 방법으로 해독하였다. 현재, 암호의 해독작업은 DNA 계산에 있어서 적합하다고 생각되어진다. DES의 해독에는 5일간의 시간에 계산되어진다. 이것은 각각의 연산이 아마도 보조적인 로봇머신을 이용하여, 1분 이내에 실행할 수 있다는 가정에 기초하고 있다. DES 암호시스템의 해독이라는 대단히 특수한 작업에 의해 DNA 계산에 관한 실행가능성의 한계가 명확히 보여지고 있다. DNA 계산의 초병렬성을 이용하면 잘 알려진 여러 가지의 NP-Complete 문제를 푸는 것과 같이 축차적인 처리에 의한 계산에서는 지수함수를 이용하는 문제를 선형시간으로 바꿀 수 있을 것이다

DNA 계산의 이론에 관한 중심적 역할인 Watson, Crick 상보성과 트윈셔플 언어(Twin-Shuffle language)와의 상관관계에 대하여 간단히 살펴본다. 트윈셔플 언어에 관하여는 기본적인 변형판이 4개의 기호 {0, 1, $\overline{0}$, $\overline{1}$}로부터 알파벳 상의 언어로서 부여하여, TS로 표현된다. 알파벳 {0, 1} 상의 언어 w(즉, w는 0과 1로 만들어진 기호열)를 생각해본다. $\overline{w}$는 $\overline{0}$와 $\overline{1}$로 된 w의 상보적 기호열이다. 예를 들면, w = 00101로 하면 $\overline{w}$= $\overline{0}\,\overline{0}\,\overline{1}\,\overline{0}\,\overline{1}$이다. shuffle(w, $\overline{w}$)에 의해, w와 $\overline{w}$를 shuffle한 결과의 집합, 즉 w와 $\overline{w}$에 대한 순서는 유지하면서 기호를 임으로 shuffle한 것의 전체를 표시한다. 예를 들면, $0\overline{0}0\overline{0}1\overline{1}0\overline{0}\overline{1}1$, $\overline{0}\,\overline{0}\,1\overline{1}\,0\,1$ 00101, $00\overline{0}10\overline{0}\,1\,\overline{0}\,\overline{1}1$은 각각의 집합 shuffle(w, $\overline{w}$)에 속하지만 $0\overline{0}0\overline{0}1\overline{1}\,0\,1\,\overline{0}\,1$ 는 속하지 않는다. TS는 w가 (0,1)상의 임의에 언어를 취할 때의 shuffle(w, $\overline{w}$)에 속한 언어 전체로부터 된다. 4개의 기호 0, 1, $\overline{0}$,$\overline{1}$ 에서 주어진 언어 x가 실제 TS에 속하는가 아닌가를 판정하는 간단한 방법을 제시한다. 먼저, x 로부터 $\overline{0}$와 $\overline{1}$를 제거하고 그 결과를 $\overline{x}$로 한다. 다음에 x로부터 0과 1을 제거하고, 남아 있는 기호열로부터 모든 $\overline{\ }$를 제거한 언어를 $\vec{\overline{x}}$ 로 한다. 이때 원래의 x가 TS에 속하는 것은 $\vec{\overline{x}} = x$ 일 때뿐이다. DNA 알파벳과 앞에서 논한 4개의 기호로 된 알파벳과 관련 A=0, G=1, T=$\overline{0}$, C=$\overline{1}$를 생각해본다. 혹시 각 쌍(0, $\overline{0}$)과 (1, $\overline{1}$)에 대한 기호를 각각 상보성이라고 보면 이 상보성은 Watson, Crick 의 상보성과 같은 것으로 간주할 수 있다. TS 언어와 이중가닥 DNA와의 상호관계는 다음과 같다. 예를 들어, 이중가닥 TAGCATCAT, ATCGTAGTA는 아래와 같이 바꾸어 쓸 수 있다.

$\overline{0}01\overline{1}0\overline{0}\,\overline{1}\,0\overline{0}$, $0\overline{0}\,1\,1\overline{0}01\overline{0}\,0$

3. 결 론

앞으로의 컴퓨터는 "Silicon에서 Carbon"으로, "Microchip에서 DNA 분자"로 발전해 나갈 것이다. 현시점에서 중요한 점은 DNA 컴퓨터는 종래의 컴퓨터에 비해 새로운 계산 패러다임을 갖고 있을 뿐만 아니라, 어떤 종류의 문제의 클래스는 특히 DNA 컴퓨팅 방법이 매우 효율적이라고 생각된다.

DNA 컴퓨팅의 수학적인 이론으로는 형식언어 이론의 틀을 이용하여 전개된다. DNA 분자는 두 개의 사슬의 기호열에 대한 연산으로서 자연적으로 표현할 수 있다. 이러한 분자계산을 위한 수학적인 이론과 계산 모델에 관한 연구 성과로는 sticker 시스템, Watson · Crick 오토마타, 삽입 · 삭제 시스템, splicing 시스템, 유한 H 시스템, 환상기호열의 splicing, 분산형 H 시스템 등을 들 수 있다.

DNA 컴퓨터는 0과 1을 사용하여 연산 및 정보의 기억을 하지 않고, DNA 분자상의 A, C, G, T 의 염기배열의 패턴에 의해 표현되는 데이터를 취급한다. DNA 컴퓨터의 속도는 현재의 슈퍼컴퓨터보다도 훨씬 빠르고 극소형의 크기로 만들 수 있다. 또한 저장할 수 있는 정보량도 엄청나서 현재의 CD 1조개의 양에 해당하는 정보를 단지 물방울 크기의 DNA 컴퓨터에 저장할 수 있다. 이러한 DNA 컴퓨터는 복잡한 비밀 암호를 해독한다든지, 일기예보, 비행기 설계 등과 같이 복잡한 병렬환경에 적합한 문제를 푸는데 유용하게 쓰일 수 있다.

장기적인 관점에서 DNA 컴퓨팅의 미래에 대해서 예측할 수 있다. DNA의 단일가닥 분자를 튜어링 머신의 계산상황을 나타내는 시점표시를 부호화 하는 것에 이용할 수 있고, 현재 이용 가

능한 프로토콜 또는 효소를 이용하여 튜어링 머신의 실행에 대응할 수 있도록 분자의 연속적인 연결 표현을 이루어 낼 수 있다. 장래에는 분자생물학에 관한 연구의 발전에 의해 마크로 분자를 조작하는 기술이 개량될 것이다. 화학에 관한 연구의 성과에 의해 인공적으로 설계 합성한 효소가 개발될 지도 모른다. 최종적으로는 단일의 마크로 분자에 리보솜과 같은 효소가 여러개 접합하여 작업하고 있는 형태의 범용 컴퓨터를 상상할 수 있을 것이다.

참고문헌

[1] G. Paun, G. Rozenberg and A. Salomaa. *DNA computing*. Springer-verlag, 1998.

[2] L. M. Adleman, "Molecular computation of solutions to combinatorial problems." *Science*, 1021-1024, Nov. 1994.

[3] M. Arita, M. Hagiya, and A. Suyama "Joining and rotating data with molecules," *IEEE Intern. Conf. on Evolutionary Computing*, Indianapolis, 1997, 243-248.

[4] D. Bonch, C. Dunworth, R. J. Lipton and J. Sgall "On the computational power of DNA," *Discrete Appl. Math*, 71. 1996, pp. 79-94.

[5] R. P Feynman, In D. H. Gilbert (ed.). *Miniaturization*. Reinhold, New York, 1961, pp. 282-296.

[6] R. J. Lipton "Using DNA to solve NP-complete problems," *Science*, 268, pp. 542-545, Apr. 1995.

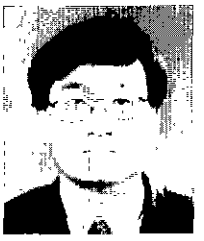
[7] M. Amos, "DNA Computation," *PhD thesis*, Department of Computer Science, University of Warwick, UK, 1997.

[8] Boneh, D., Dunworth, C., and Lipton, R. J., "Breaking DES using a molecular computer," DIMACS workshop, pp. 37-66, American Mathematical Society, 1996.

[9] Zhang, B. T. and Shin, S. Y., "Molecular Algorithms for Efficient and Reliable DNA Computing", Proc. of the 3rd *Annual Genetic Programming Conference*, Jul. pp. 735-742, 1998.

[10] Zhang, B. T. and Shin, S. Y., "Code Optimization for DNA Computing of Maximal Cliques," *Advances in Soft Computing Engineering Design and manufacturing*, Springer-Verlag, 1998.

이 상 구

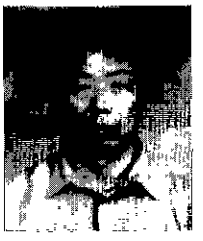


1978 서울대학교 전자공학과 졸업 (학사)
1981 한국과학기술원 전산학과 졸업 (석사), 와세다대학 전기전자컴퓨터공학과 졸업 (Ph D)
1983~현재. 한남대학교 컴퓨터공학과 교수
관심분야 병렬처리, 컴퓨터구조, Fuzzy-Neuro 시스템, DNA 컴퓨팅
E mail:sglee@eve.hannam ac kr

이 광 형

1978 서울공대 산업공학 학사
1980 한국과학원 산업공학 석사
1982 프랑스 INSA 전산학과 석사 (DEA)
1985 프랑스 INSA 전산학과 공학박사
1988 1 프랑스 국가박사(전산학 INSA- LYON대)
1985~1995 한국과학기술원 전산학과 조교수 및 부교수
1995~현재 한국과학기술원 전자전산학과 교수
1985 프랑스 INSA
1995 미국 Stranford Research Institute
관심분야 퍼지 이론 및 응용, 인공지능, 전문가 시스템 등
E-mail khlee@monami kaist.ac.kr

임 기 영



1979 2 건국대학교 전자공학과
1980 2 건국대학교 전기전자공학과 (공학박사)
1989 9 국립대만대학 전자계산학과 박사과정 무료
1991. 9 건국대학원 전자공학과(디지탈및컴퓨터시스템전공)박사
1980~1992 경원전문대학 전자계산과 부교수
1996~1998 시드니대학 전자공학과, 뉴시우스웨일즈대학 컴퓨터공학과 교환교수
관심분야 인공지능, 지능제어, 생체제어
E-mail:hmgy@hyunam.fnut ac.kr